

UNIVERSIDADE FEDERAL DO PARANÁ

ALLAN LERNER GOLDENBERG

CLAUDINEI VARELA

DIEGO LEMOS PADILHA

JEAN DE JESUS FREIRE

EASY WASH: LAVAR SEU CARRO NUNCA FOI TÃO FÁCIL

CURITIBA

2020

ALLAN LERNER GOLDENBERG

CLAUDINEI VARELA

DIEGO LEMOS PADILHA

JEAN DE JESUS FREIRE

EASY WASH: LAVAR SEU CARRO NUNCA FOI TÃO FÁCIL

Monografia apresentada como requisito parcial à obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas, do Setor de Educação Profissional e Tecnológica da Universidade Federal do Paraná.

Orientador: Prof. Dr. Razer Anthom Nizer Rojas
Montaño

CURITIBA

2020

TERMO DE APROVAÇÃO

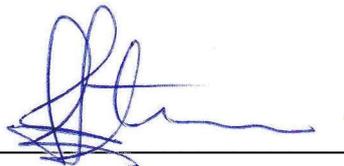
Allan Lerner
Claudinei Varela
Diego Lemos
Jean Freire

Easywash: Lavar Seu Carro Nunca Foi Tão Fácil

Monografia aprovada como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, do Setor de Educação Profissional e Tecnológica da Universidade Federal do Paraná.



Prof. Razer A. N. R. Montañó
Orientador – SEPT/UFPR



Profa. Rafaela Mantovani Fontana
SEPT/UFPR



Prof.^a Jaime Wojciechowski
SEPT/UFPR

Curitiba, 03 de Setembro de 2020.

AGRADECIMENTOS

Agradecemos primeiramente a Deus por nos conceder a oportunidade de concluir mais uma etapa da nossa vida e por todo o aprendizado adquirido nessa caminhada. Também por nos conceder saúde, paciência, humildade e por colocar as pessoas certas no momento oportuno em nossas vidas.

Nosso registro de agradecimento também a todas as pessoas que de algum modo colaboraram conosco nessa etapa.

Ao nosso professor orientador Razer Anthom Nizer Rojas Montañó pela amizade, paciência, disposição e todo conhecimento transmitido ao longo desse projeto.

A todos os demais professores do Curso de Análise e Desenvolvimento de Sistemas, pelo empenho, didática e por todo o conteúdo transmitido. Contribuíram muito para nosso amadurecimento e crescimento profissional, fato crucial para concluir este ciclo.

Aos nossos familiares, de modo especial aos nossos pais por todo apoio e compreensão e também aos nossos amigos pela paciência, principalmente nos momentos de ausência.

RESUMO

O seguinte artigo propõe a construção de uma plataforma para centralizar o fluxo de atendimentos relacionados a atividade de lavagem de veículos. A partir da realização de pesquisas bibliográficas e análises dos próprios serviços semelhantes disponíveis no mercado, foi possível constatar a necessidade e espaço para construção de um aplicativo que proporcione uma nova experiência entre clientes e lavadores de automóveis. O sistema é composto por um aplicativo móvel, o qual é utilizado tanto pelos clientes quanto pelos prestadores de serviço e que contemplam desde a consulta de lava-cars até a solicitação e execução de um atendimento. Além disso, foi desenvolvido um sistema web projetado para ser um instrumento de gerenciamento do produto. As tecnologias empregadas foram as seguintes: A biblioteca React Native para a construção do aplicativo móvel e React.js para o desenvolvimento do sistema Web. A equipe optou por elaborar uma API Restful utilizando o framework Spring Boot para controlar a camada back-end da aplicação, a qual é consumida tanto pelo sistema Web, quanto pelo móvel. O resultado obtido com esta implementação foi um sistema robusto e escalável, com possibilidades de adição de novas funcionalidades e melhorias que foram identificadas ao longo da sua construção. Este projeto proporcionou um grande aprendizado para a equipe que vai desde a análise e levantamento de requisitos até a publicação de um produto.

Palavras-chave: Automóveis, Lavagem, Serviços.

ABSTRACT

The following article proposes the construction of a platform to centralize the flow of assistance related to vehicle washing activities. From bibliographic research and analysis of similar services available on the market, it was possible to see the need and space for building an application that provides a new experience for customers and car washers. The system consists of a mobile application, which is used by both customers and service providers, which range from the consultation of car washes to the request and execution of a service. In addition, a web system designed to be a product management tool was developed. The technologies employed were as follows: The React Native library for the construction of the mobile application and React.js for the development of the Web system. The team chose to develop a Restful API using the Spring Boot framework to control the back-end application layer , which is consumed by both the web and mobile systems. The result obtained with this implementation was a robust and scalable system, with possibilities for adding new features and improvements that were identified throughout its construction. This project provided a great learning experience for the team, ranging from the analysis and survey of requirements to the publication of a product.

Key-words: Vehicles. Washing. Services.

LISTA DE FIGURAS

FIGURA 1 - EXPANSÃO DE APPS	24
FIGURA 2 - REPRESENTAÇÃO DA TEORIA DAS FILAS	25
FIGURA 3 - DIAGRAMA EM BLOCOS DE UM SISTEMA DE FILAS	26
FIGURA 4 - BANCO DE DADOS RELACIONAIS MAIS UTILIZADOS AO REDOR DO MUNDO	32
FIGURA 5 - EMPRESAS AO REDOR DO MUNDO QUE UTILIZAM O MYSQL.....	33
FIGURA 6 - BIBLIOTECAS, FRAMEWORKS E FERRAMENTAS MAIS POPULARES NO STACK OVERFLOW.....	34
FIGURA 7 - FRAMEWORKS WEB MAIS POPULARES NO STACK OVERFLOW EM 2019	36
FIGURA 8 - PROCESSO DE GESTÃO DE PROJETOS TRADICIONAL	40
FIGURA 9 - PROCESSO DE GESTÃO DE PROJETOS TRADICIONAL	41
FIGURA 10 - INTERFACES APLICATIVO APPWASH	43
FIGURA 11 - INTERFACES APLICATIVO LAVÔ	44
FIGURA 12 - UTILIZAÇÃO DA FERRAMENTA TRELLO PARA ATRIBUIÇÃO E ACOMPANHAMENTO DAS ATIVIDADES.....	48
FIGURA 13 - INTERFACE GRÁFICA DO ASTAH PROFESSIONAL 8.1	49
FIGURA 14 - AMBIENTES DE DESENVOLVIMENTO MAIS POPULARES, SEGUNDO A PLATAFORMA STACK OVERFLOW	51
FIGURA 15 - CRONOGRAMA DE ATIVIDADES SPRINT 0 A SPRINT 2	56
FIGURA 16 - CRONOGRAMA DE ATIVIDADES SPRINT 3 A SPRINT 8	57
FIGURA 17 - CRONOGRAMA DE ATIVIDADES SPRINT 6 A SPRINT 8	57
FIGURA 18 - CRONOGRAMA DE ATIVIDADES SPRINT 9 A SPRINT 10.....	58
FIGURA 19 - CRONOGRAMA DE ATIVIDADES SPRINT 10 A SPRINT 11.....	59
FIGURA 20 - CRONOGRAMA DE ATIVIDADES SPRINT 13 A SPRINT 14.....	59
FIGURA 21 - CRONOGRAMA DE ATIVIDADES SPRINT 15 A SPRINT 16.....	60
FIGURA 22 - CRONOGRAMA DE ATIVIDADES SPRINT 17	60
FIGURA 23 - DIAGRAMA GERAL DA APLICAÇÃO	72
FIGURA 24 - ARQUITETURA MVC DO SISTEMA BACK-END.....	74
FIGURA 25 - PAGINA DE LOGIN	75
FIGURA 26 - ESQUECI A SENHA.....	76
FIGURA 27 - PAGINA HOME (RELATÓRIOS).....	76

FIGURA 28 - PAGINA HOME (NOTIFICAÇÕES).....	77
FIGURA 29 - PÁGINA MODERADORES.....	77
FIGURA 30 - PÁGINA NOVO MODERADOR.....	78
FIGURA 31 - PÁGINA MODERADORES MODO NOTURNO.....	78
FIGURA 32 - PÁGINA CLIENTES.....	79
FIGURA 33 - PÁGINA CLIENTE DETALHES.....	79
FIGURA 34 - PÁGINA CLIENTES ENVIAR NOTIFICAÇÃO.....	80
FIGURA 35 - PÁGINA PRESTADORES.....	80
FIGURA 36 - PÁGINA PRESTADORES 1.....	81
FIGURA 37 - PÁGINA PRESTADORES 2.....	81
FIGURA 38 – PÁGINA APROVAÇÕES PENDENTES.....	82
FIGURA 39 - PÁGINA APROVAÇÕES PENDENTES DETALHES P1.....	82
FIGURA 40 - PÁGINA APROVAÇÕES PENDENTES DETALHES P2.....	83
FIGURA 41 - PÁGINA NOTIFICAÇÕES.....	83
FIGURA 42 - PÁGINA NOTIFICAÇÕES DETALHES.....	84
FIGURA 43 - PÁGINA NOTIFICAÇÕES NOVA NOTIFICAÇÃO.....	84
FIGURA 44 - ICONE DO APLICATIVO.....	85
FIGURA 45 - SPLASH SCREEN.....	85
FIGURA 46 - TELA DE INICIAL.....	86
FIGURA 47 – TELA DE LOGIN.....	86
FIGURA 48 - TELAS DE CADASTRO CLIENTE.....	87
FIGURA 49 – TELAS DE CADASTRO PRESTADOR.....	87
FIGURA 50 - TELA ESQUECI A SENHA.....	88
FIGURA 51 - TELA PRINCIPAL DO SISTEMA.....	88
FIGURA 52 – TELA CADASTRO FOTO.....	89
FIGURA 53 – TELA MEUS CARTÕES.....	89
FIGURA 54 – TELA MEUS AUTOMÓVEIS.....	90
FIGURA 55 – TELA PAGAMENTOS.....	90
FIGURA 56 – TELA DE SERVIÇOS.....	91
FIGURA 57 – TELA DE NOTIFICAÇÕES.....	91
FIGURA 58 – TELAS DE ATENDIMENTO 1.....	92
FIGURA 59 – TELAS DE ATENDIMENTO 2.....	93
FIGURA 60 – TELA DE PEDIDOS.....	93
FIGURA 61 – INFORMAÇÕES DE ENDEREÇO DE CLIENTE/PRESTADOR.....	94

FIGURA 62 – TELAS DE ATENDIMENTO	95
FIGURA 63 - DIAGRAMA DE CASOS DE USO - CLIENTE	117
FIGURA 64 - DIAGRAMA DE CASOS DE USO - PRESTADOR.....	118
FIGURA 65 - DIAGRAMA DE CASOS DE USO - MODERADOR.....	119
FIGURA 66 - DV01 - TELA INICIAL APLICATIVO CLIENTE	120
FIGURA 67 - DV02 – TELA DE LOGIN.....	121
FIGURA 68 - DV03 – TELA DE SOLICITAÇÃO PARA RECUPERAÇÃO DE SENHA	121
FIGURA 69 - DV01 - TELA DE CADASTRO DE DADOS PESSOAIS DO CLIENTE	123
FIGURA 70 - DV02 - TELA DE CADASTRO DE DADOS DO ENDEREÇO DO CLIENTE	124
FIGURA 71 - DV03 - TELA DE UPLOAD DE FOTO DE PERFIL.....	124
FIGURA 72 - DV01 - TELA DE CADASTRO E EDIÇÃO DE AUTOMÓVEL	127
FIGURA 73 - DV01 - TELA INICIAL DE GERENCIAMENTO DE INFORMAÇÕES	128
FIGURA 74 - DV02 - TELA DE GERENCIAMENTO DE AUTOMÓVEIS	129
FIGURA 75 - DV03 - TELA DE GERENCIAMENTO DE CARTÕES.....	129
FIGURA 76 - DV01 - TELA DE PESQUISA DE SERVIÇOS.....	131
FIGURA 77 - DV02 - TELA DE PESQUISA POR FILTRO.....	132
FIGURA 78 - DV03 - TELA LISTAGEM DE EMPRESAS POR FILTRO	132
FIGURA 79 - DV01 - TELA DE NOTIFICAÇÕES DO CLIENTE	134
FIGURA 80 - DV01 - TELA DE CONTRATAR SERVIÇO RESERVA HORÁRIO....	135
FIGURA 81 - DV02 - TELA DE SELEÇÃO DE SERVIÇOS DA EMPRESA.....	136
FIGURA 82 - DV01 - TELA DE SELEÇÃO DE FORMA DE PAGAMENTO	137
FIGURA 83 - DV02 – TELA DE SELEÇÃO DE CARTÃO DE CRÉDITO	138
FIGURA 84 - DV03 - TELA DE CADASTRO DE CARTÃO DE CRÉDITO.....	138
FIGURA 85 - DV04 - TELA DE AGUARDANDO CONFIRMAÇÃO DE SERVIÇO....	139
FIGURA 86 - DV01 - TELA DE HISTÓRICO DE SERVIÇOS CONTRATADOS.....	141
FIGURA 87 - DV02 - TELA HISTÓRICO DETALHADO DE SERVIÇO CONTRATADO	142
FIGURA 88 - DV01 - TELA DE PERFIL DO PRESTADOR DE SERVIÇOS	143
FIGURA 89 - DV01 - TELA DE AVALIAÇÃO DE SERVIÇO	144
FIGURA 90 - DV01 - FORMULÁRIO DE ENVIO DE MENSAGEM AO SUPORTE	146
FIGURA 91 - DV01 - FORMULÁRIO DE CADASTRAR EMPRESA	148

FIGURA 92 - DV02 - FORMULÁRIO DE CADASTRAR ENDEREÇO DA EMPRESA	149
FIGURA 93 - DV03 - FORMULÁRIO DE CADASTRAR SERVIÇO PRINCIPAL.....	149
FIGURA 94 - DV04 - FORMULÁRIO DE CADASTRAR DADOS FINANCEIROS...	150
FIGURA 95 - DV05 - FORMULÁRIO DE CADASTRAR FOTO.....	150
FIGURA 96 - DV06 - TELA DE FINALIZAÇÃO	151
FIGURA 97 - DV01 - TELA DE LOGIN	152
FIGURA 98 - DV01 - TELA RECUPERAR SENHA.....	153
FIGURA 99 - DV01 - TELA DE MINHAS INFORMAÇÕES – DADOS DA EMPRESA	154
FIGURA 100 - DV02 - TELA DE MINHAS INFORMAÇÕES – DADOS DO ENDEREÇO	155
FIGURA 101 - DV03 - TELA DE MINHAS INFORMAÇÕES - FOTO	155
FIGURA 102 - DV01 - TELA DE LISTAGEM DE SERVIÇOS	157
FIGURA 103 - DV01 - TELA DE INCLUSÃO/EDIÇÃO DE SERVIÇOS	157
FIGURA 104 - DV01 - TELA DE DADOS FINANCEIROS.....	159
FIGURA 105 - DV01 - TELA DE HISTÓRICO DE SERVIÇOS PRESTADOS	160
FIGURA 106 - DV02 - TELA DE HISTÓRICO DE SERVIÇO DETALHADO	161
FIGURA 107 - DV01 - TELA DE NOTIFICAÇÕES.....	162
FIGURA 108 - DV01 - TELA DE ATENDIMENTO.....	163
FIGURA 109 - DV02 - TELA DE INICIO DE ATENDIMENTO.....	164
FIGURA 110 - DV03 - TELA DE ATENDIMENTO EM ANDAMENTO	164
FIGURA 111 - DV04 - TELA DE ATENDIMENTO CONCLUÍDO	165
FIGURA 112 - DV01 - TELA DE CONFIRMAÇÃO DE ATENDIMENTO	166
FIGURA 113 - DV01 - FORMULÁRIO DE LOGIN DO SISTEMA.....	168
FIGURA 114 - DV02 - FORMULÁRIO DE RECUPERAÇÃO DE SENHA.....	169
FIGURA 115 - DV01 - TELA DE GERENCIAMENTO DE MODERADORES.....	171
FIGURA 116 - DV02 - TELA DE CADASTRO/EDIÇÃO DE UM MODERADOR.....	171
FIGURA 117 - DV01 - TELA DE GERENCIAMENTO DE CLIENTES.....	174
FIGURA 118 – DV02 - TELA DE VISUALIZAÇÃO DE INFORMAÇÕES DE UM CLIENTE	174
FIGURA 119 - DV01 - TELA DE GERENCIAMENTO DE PRESTADORES DE SERVIÇO	176
FIGURA 120 - DV02 - TELA DE VISUALIZAÇÃO DE INFORMAÇÕES DE UM	

PRESTADOR DE SERVIÇO.....	177
FIGURA 121 - DV01 - TELA DE LISTAGEM DE SOLICITAÇÕES DE PEDIDO PARA ACESSO A PLATAFORMA.....	179
FIGURA 122 - DV02 - TELA DE VISUALIZAÇÃO DE SOLICITAÇÃO DE PEDIDO PARA ACESSO A PLATAFORMA.....	179
FIGURA 123 - DV01 - TELA DE LISTAGEM DE MENSAGENS ENVIADAS PELOS USUÁRIOS A PLATAFORMA.....	181
FIGURA 124 - DV02 - TELA DE VISUALIZAÇÃO DE MENSAGEM ENVIADA A MODERAÇÃO À PLATAFORMA.....	182
FIGURA 125 - TELA DE RELATÓRIOS.....	183
FIGURA 126 - DIAGRAMA DE CLASSES DE DOMÍNIO.....	185
FIGURA 127 - REALIZAR LOGIN.....	186
FIGURA 128 - CADASTRAR-SE NO SISTEMA.....	186
FIGURA 129 - CADASTRAR AUTOMÓVEL.....	187
FIGURA 130 - GERENCIAR DADOS CADASTRAIS.....	187
FIGURA 131 - PESQUISAR SERVIÇOS.....	187
FIGURA 132 - VISUALIZAR NOTIFICAÇÕES.....	188
FIGURA 133 - CADASTRAR ATENDIMENTO.....	188
FIGURA 134 - REALIZAR PAGAMENTO.....	188
FIGURA 135 - CONSULTAR HISTÓRICO DE SERVIÇOS COTRATADOS.....	189
FIGURA 136 - VISUALIZAR INFORMAÇÕES E SERVIÇOS DA EMPRESA.....	189
FIGURA 137 - AVALIAR ATENDIMENTO.....	190
FIGURA 138 - ENVIAR MENSAGEM AO SUPORTE.....	190
FIGURA 139 - CADASTRAR PRESTADOR.....	191
FIGURA 140 - REALIZAR LOGIN.....	191
FIGURA 141 - REALIZAR LOGIN/ RECUPERAR SENHA.....	191
FIGURA 142 - GERENCIAR DADOS CADASTRAIS.....	192
FIGURA 143 - GERENCIAR SERVIÇOS.....	192
FIGURA 144 - GERENCIAR FORMAS DE PAGAMENTO.....	193
FIGURA 145 - CONSULTAR HISTÓRICO DE ATENDIMENTOS REALIZADOS... ..	193
FIGURA 146 - VISUALIZAR NOTIFICAÇÕES.....	193
FIGURA 147 - REALIZAR ATENDIMENTO.....	194
FIGURA 148 - ACEITAR SOLICITAÇÃO DE ATENDIMENTO.....	194
FIGURA 149 - REALIZAR LOGIN.....	195

FIGURA 150 - REALIZAR LOGIN - ESQUECI MINHA SENHA.....	195
FIGURA 151 - GERENCIAR MODERADORES	196
FIGURA 152 - GERENCIAR CLIENTES.....	196
FIGURA 153 - GERENCIAR PRESTADORES DE SERVIÇO	197
FIGURA 154 - GERENCIAR NOVAS SOLICITAÇÕES DE PRESTADORES À PLATAFORMA.....	197
FIGURA 155 - GERENCIAR MENSAGENS ENVIADAS	198
FIGURA 156 - DIAGRAMA DE PACOTES.....	199
FIGURA 157 - DIAGRAMA DE IMPLEMENTAÇÃO DE PACOTES DAO	199
FIGURA 158 - DIAGRAMA DE IMPLEMENTAÇÃO DE PACOTES SERVICE.....	200
FIGURA 159 - DIAGRAMA DE IMPLEMENTAÇÃO DE PACOTES CONTROLLER	200
FIGURA 160 - DIAGRAMA DE IMPLEMENTAÇÃO DE PACOTES VIEW	201
FIGURA 161 - MODELO FÍSICO DO BANCO DE DADOS.....	202

LISTA DE QUADROS

QUADRO 1 - COMPARATIVO DE APLICATIVOS.....	45
QUADRO 2 - PLANO DE RISCOS.....	54
QUADRO 3 - PLANO DE RESPONSABILIDADES.....	55

LISTA DE ABREVIATURAS E SIGLAS

API	- Application Programming Interface, Interface de Programação de Aplicativos
AWS	- Amazon Web Services
CSS	- Cascading Style Sheets, Folha de Estilo em Cascatas
DAO	- Data Access Object, objeto de acesso à dados
DBA	- Database Administrator, administrador de banco de dados
DV	- Data View
EAP	- Estrutura Analítica de Projeto
EE	- Enterprise Edition
GB	- Gigabyte
GHz	- Giga Hertz
HD	- Hard Disk, disco rígido
HTML	- Hyper Text Markup Language
IBM	- International Business Machines Corporation
IBGE	- Instituto Brasileiro de Geografia e Estatística
IBOPE	- Instituto Brasileiro de Opinião Pública
IDE	- Integrated Development Environment, ambiente integrado de Desenvolvimento
ISMA	- International Stress Management Association
IOS	- iPhone Operation System
JPA	- Java Persistence API
LTS	- Long Time Support, suporte à longo prazo
MAC	- Macintosh Operating System
MEI	- Microempreendedor Individual
PO	- Product Owner
POC	- Proof of Concept, prova de conceito
RAM	- Random Process Memory, memória de acesso aleatório
SEBRAE	- Serviço Brasileiro de Apoio às Micro e Pequenas Empresas
SEPT	- Setor de Educação Profissional Tecnológica
SGBD	- Sistema de Gerenciamento de Banco de Dados
SM	- Scrum Master
SQL	- Structured Query Language, linguagem de consulta estruturada

SSD	- Solid State Drive
UC	- Use Case, caso de uso
UFPR	- Universidade Federal do Paraná
UML	- Unified Modeling Language, linguagem de modelagem unificada
UNICEF	- United Nations Children's Fund, Fundo das Nações Unidas para Infância

SUMÁRIO

1	INTRODUÇÃO	19
1.1	PROBLEMA	20
1.2	OBJETIVOS	20
1.2.1	Objetivo geral	21
1.2.2	Objetivos específicos	21
1.3	JUSTIFICATIVA	21
2	FUNDAMENTAÇÃO TEÓRICA	23
2.1	FUNDAMENTAÇÃO DO NEGÓCIO	23
2.1.1	Uso do celular	23
2.1.2	Teoria das filas	25
2.1.3	Perda de tempo para o ser humano	27
2.2	FUNDAMENTAÇÃO TEÓRICA DAS TECNOLOGIAS	28
2.2.1	Unified Modeling Language – UML	28
2.2.2	Banco de dados	30
2.2.3	React	33
2.2.4	React Native	34
2.2.5	Linguagem de programação Java	34
2.2.6	<i>Spring</i>	35
2.2.7	Metodologia Ágil	38
2.3	COMPARATIVO COM APLICAÇÕES SEMELHANTES	42
2.3.1	App Wash	42
2.3.2	LAVÔ	43
3	MATERIAIS E MÉTODOS	46
3.1	SCRUM	46
3.2	TRELLO	47
3.3	ASTAH	48
3.4	VISUAL STUDIO CODE	50
3.5	ECLIPSE IDE	51
3.7	INFRAESTRUTURA DE DESENVOLVIMENTO	53
3.8	PLANO DE RISCOS	54
3.9	RESPONSABILIDADES	55
3.10	PLANO DE ATIVIDADES	56

3.10.1	Planejamento.....	60
3.10.2	Sprint 1	61
3.10.3	Sprint 2	62
3.10.4	Sprint 3	62
3.10.6	Sprint 5	63
3.10.7	Sprint 6	64
3.10.8	Sprint 7	65
3.10.9	Sprint 8	65
3.10.10	Sprint 9	66
3.10.11	Sprint 10	66
3.10.12	Sprint 11	67
3.10.13	Sprint 12	67
3.10.14	Sprint 13	68
3.10.15	Sprint 14	69
3.10.16	Sprint 15	69
3.10.17	Sprint 16	70
3.10.18	Sprint 17	70
4	APRESENTAÇÃO DO SISTEMA	72
4.1	ARQUITETURA DO SISTEMA.....	73
4.2	FUNCIONALIDADES.....	74
4.3	SOBRE O SISTEMA WEB	75
4.4	SOBRE O SISTEMA MOBILE	85
5	CONSIDERAÇÕES FINAIS	96
5.1	RECOMENDAÇÕES PARA TRABALHOS FUTUROS.....	97
	REFERÊNCIAS.....	99
	APÊNDICE A - LISTA DE REQUISITOS APLICATIVO MÓVEL CLIENTE	103
	APÊNDICE B - LISTA DE REQUISITOS APLICATIVO MÓVEL PRESTADOR	109
	APÊNDICE C - LISTA DE REQUISITOS SISTEMA WEB MODERADOR.....	114
	APÊNDICE D - DIAGRAMAS DE CASO USO CLIENTE.....	117
	APÊNDICE E - DIAGRAMAS DE CASO USO PRESTADOR.....	118
	APÊNDICE F - DIAGRAMAS DE CASO USO MODERADOR.....	119
	APÊNDICE G - ESPECIFICAÇÃO APLICATIVO CLIENTE.....	120
	APÊNDICE H - ESPECIFICAÇÃO APLICATIVO PRESTADOR.....	148
	APÊNDICE I - ESPECIFICAÇÃO SISTEMA WEB MODERADOR	168

APÊNDICE J - DIAGRAMA DE CLASSES DE DOMÍNIO.....	185
APÊNDICE K - DIAGRAMAS DE SEQUÊNCIA CLIENTE	186
APÊNDICE L - DIAGRAMAS DE SEQUÊNCIA PRESTADOR.....	191
APÊNDICE M – DIAGRAMAS DE SEQUÊNCIA MODERADOR.....	195
APÊNDICE N - DIAGRAMAS DE PACOTES.....	199
APÊNDICE O – MODELO FÍSICO DO BANCO DE DADOS	202

1 INTRODUÇÃO

A falta de tempo no mundo moderno é um mal que afeta cada vez mais pessoas, de acordo com uma pesquisa realizada pela ISMA-BR (International Stress Management Association Brasil), 62% dos brasileiros sofrem com a falta de tempo devido ao grande número de tarefas no dia a dia (EQUIPE ADMINISTRADORES, 2015).

A ideia de produzir mais em menos tempo, tornando as atividades mais produtivas e rápidas, surgiu na revolução industrial. Diversas máquinas começaram a fazer o trabalho dos humanos em um tempo menor e de maneira mais efetiva, essa forma de agir e pensar saiu das fábricas e passou a fazer parte do cotidiano das pessoas. Atividades como o deslocamento de um lugar ao outro, tarefas domésticas como lavar o carro e até mesmo o lazer, tudo se tornou mais rápido e ágil com o foco na redução do tempo gasto (BLOG FREESIDER, 2016).

Visando a produtividade e um grande potencial de mercado, começaram a surgir terceiros dispostos a realizar as tarefas que os outros não podiam: motoristas particulares (Uber), entregadores de comida (Rappi e iFood) e muitos outros serviços passaram a coexistir com o intuito de reduzir filas e agilizar o dia, já tão corrido, das pessoas.

Em aplicativos que oferecem algum tipo de serviço, a satisfação é algo essencial para que o cliente volte a consumir a atividade oferecida. Segundo Filho (2016):

“As filas enfrentadas pelos consumidores reduzem a satisfação dos clientes e, conseqüentemente, reduzem a lealdade dos mesmos em relação a empresa, ou ocasiona perda de receita. Portanto, a redução das filas é fundamental para aumentar a competitividade das mesmas.” (FILHO, 2016, p. 10)

Percebeu-se que as filas e a satisfação dos clientes estão ligados em uma razão inversamente proporcional ao tempo esperado, quando menos tempo, maior a felicidade do cliente.

“A satisfação do cliente consiste na sensação de prazer ou de decepção resultante da comparação do desempenho (ou resultado) percebido de um

produto em relação expectativas do comprador. Os clientes estarão satisfeitos quando suas expectativas forem atendidas e estarão encantados quando suas expectativas forem superadas. Clientes satisfeitos permanecem fiéis por mais tempo, compram mais, são menos sensíveis a preços e fazem propaganda boca-a-boca positiva sobre a empresa.” (CURRITO, 2008, p. 1)

Com a ideia de diminuir ou eliminar filas, foi desenvolvido o aplicativo Easy Wash para ajudar na solução de alguns problemas da atualidade que ainda não possuem uma ferramenta ou resolução eficiente, como: manter o seu carro limpo evitando perda de tempo e dores de cabeça.

O aplicativo vem com o objetivo de facilitar a pesquisa de preço e agendamento de lavagens para o cliente, tornando uma atividade simples, rápida e segura. Trazendo uma opção inovadora de lavagem “pronta entrega” onde o prestador de serviço se deslocará para o local do cliente a fim de realizar a limpeza do automóvel, trazendo comodidade e agilidade ao usuário do sistema.

1.1 PROBLEMA

O problema que deu origem a este trabalho se baseia na seguinte premissa: Segundo Novo Negócio (2020) e Sebrae/ES (2008) o número de automóveis aumenta mais rápido que o número de lava-carros existentes, gerando demanda e longos períodos de espera para o atendimento, falta de variedade na escolha do serviço e insatisfação do cliente. O desentendimento de valores no momento de realização de pagamento é também um ponto muito importante.

O sistema Easy Wash vem para auxiliar nas atividades de lavação de carro e realização de polimentos, que serão atendidas de maneira rápida e segura para ambas as partes, gerando lucro para o prestador e suprimindo as necessidades do cliente.

1.2 OBJETIVOS

Nesta seção serão abordados os objetivos gerais e específicos estabelecidos para realização deste projeto.

1.2.1 Objetivo geral

O objetivo deste trabalho é desenvolver o software Easy Wash, visando a conectividade de diversos prestadores de serviço à uma vasta gama de clientes, facilitando a busca por melhores preços e evitando o tempo de espera para atendimento.

1.2.2 Objetivos específicos

Os objetivos específicos deste trabalho são:

- Disponibilizar formulários de cadastro simplificados, com informações pertinentes e sem burocracias desnecessárias;
- Fornecer uma ferramenta de centralização de serviços de limpeza de automóveis, acelerando o processo agendamento e de pesquisa por filtros;
- Disponibilizar gráficos para uma melhor visualização das informações geradas pelo sistema;
- Apresentar uma interface intuitiva e fácil de ser utilizada;
- Efetuar pagamentos com agilidade, evitando problemas de valores conflitantes.

1.3 JUSTIFICATIVA

Visto a necessidade de otimizar as filas e aumentar a comodidade do usuário, o aplicativo Easy Wash propõe soluções para todos os problemas mencionados anteriormente, resolvendo as questões de longos períodos de espera, oferecendo mais variedades de escolha de serviço e proporcionando maior satisfação ao cliente. Com consultas rápidas a preços, localização e disponibilidade dos prestadores de serviço, serão evitadas as filas de espera e o atendimento será mais rápido do que nunca.

Outro diferencial será a possibilidade dos prestadores de serviço, realizarem seus atendimentos no próprio domicílio do cliente, o que aumentará a comodidade e praticidade do serviço.

A organização do restante deste trabalho está da maneira a seguir:

O Capítulo 2 descreve a fundamentação teórica do sistema, detalhando de maneira mais específica os assuntos essenciais para a construção do mesmo. Entre os assuntos relevantes estão temas como o crescimento na utilização de smartphones e aplicativos para realização de diversos serviços, a apresentação teórica das tecnologias utilizadas e comparativo de aplicativos de lava carros semelhantes e as vantagens do Aplicativo Easy Wash.

O Capítulo 3, descreve os materiais e métodos, definições técnicas necessárias para execução do projeto e como a equipe se organizou para melhor aproveitar o tempo e os recursos tecnológicos.

O Capítulo 4 é a apresentação do sistema desenvolvido. As imagens geradas refletem as telas das aplicações com a descrição de seus recursos e funcionamento, proporcionando um melhor entendimento do mesmo.

O Capítulo 5 expõe as considerações finais do projeto. São descritas as possibilidades de melhorias e de trabalhos futuros, assim como a análise dos objetivos apresentados em relação aos que foram alcançados.

Ao final deste documento estão anexados os artefatos que foram gerados durante a execução do projeto.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo será apresentada a fundamentação teórica realizada para o embasamento do desenvolvimento do software Easy Wash, a qual foi dividida em duas partes: a primeira sendo a fundamentação do negócio, em que será abordado os conceitos literários sobre o assunto, e a segunda a fundamentação das tecnologias utilizadas para o desenvolvimento do aplicativo e estudo de softwares semelhantes.

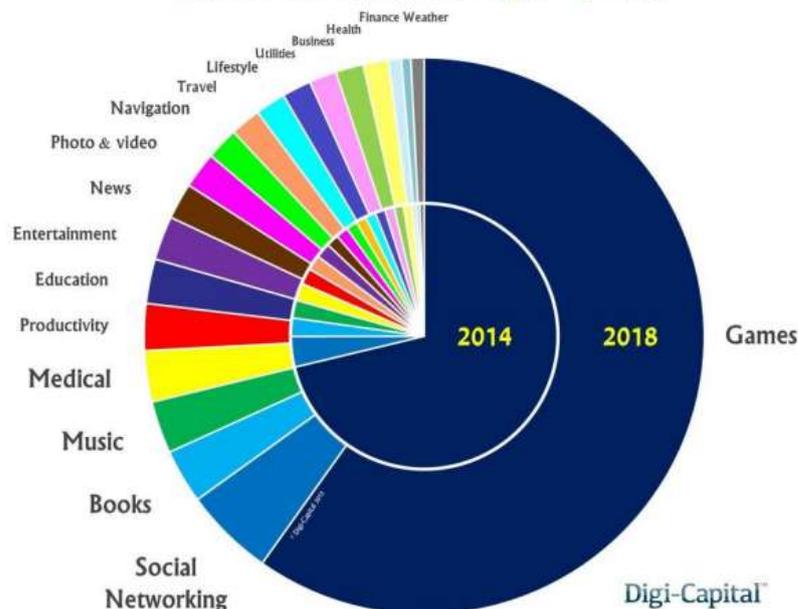
2.1 FUNDAMENTAÇÃO DO NEGÓCIO

Será abordado neste tópico a fundamentação teórica do negócio, que serviu como base para a ideia do projeto e para a construção deste trabalho.

2.1.1 Uso do celular

“O desenvolvimento da tecnologia foi mais rápido do que pudemos imaginar” (MOSAVI, 2016, p.23). De fato, diariamente milhares de pessoas usam seus telefones para diferentes finalidades básicas, como realizar compras, olhar a previsão do tempo ou o extrato bancário, fazendo download de outros aplicativos em seus smartphones quando não estão satisfeitos com os que utilizam. Quem imaginaria o que o futuro iria trazer? Por meio de poucos cliques, não é mais necessário enfrentar filas gigantescas. Os aplicativos vieram para auxiliar a vida das pessoas, e não apenas em um ramo específico (MOSAVI, 2016, p.23). A FIGURA 1 demonstra uma comparação de espaço tempo, de 2014 a 2018 (período de 4 anos), e mostra como a utilização de aplicativos expandiu para diferentes categorias, a dominância da utilização dos aparelhos móveis apenas para jogos está cada vez menor. É possível identificar as ramificações disponíveis no mercado de aplicativos, e ver o crescente avanço para ofertas de diversos serviços. (MOSAVI, 2016).

FIGURA 1 - EXPANSÃO DE APPS

\$75 billion consumer apps by 2018

FONTE: MOSAVI, 2016

Diferente de décadas atrás, smartphones têm preços acessíveis, já que são classificados como algo indispensável. Além de facilitar o contato entre as pessoas, ele também comporta muitos aplicativos, arquivos e funciona rapidamente. Mais do que tudo que foi apontado, as pessoas utilizam smartphones por causa das necessidades de socialização, conveniência e moda (XU, 2014, p. 2-4).

De acordo com o estudo de Jane Xu (2014) feito na Austrália, foram questionados 20 estudantes, sendo 4 homens e 16 mulheres, na Universidade de Taylors e na Embaixada CES. Questionados se utilizavam celulares ou smartphones, 95% responderam utilizar smartphones. No Brasil, os dados não são tão diferentes. Recentemente, em uma pesquisa relatada pelo Brasil Econômico (2019), realizada pelo Instituto Brasileiro de Opinião Pública (IBOPE) no final do ano anterior, dentre as 2.000 pessoas entrevistadas das classes A, B e C, 52% delas afirmaram que não conseguem ficar sem usar o celular por um (1) dia. A respeito da outra metade, apenas 7% informaram conseguir ficar longe do aparelho por mais de 12 horas.

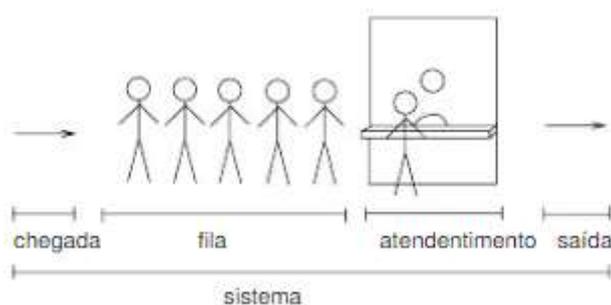
Uma coisa é certa: tanto a pesquisa australiana quanto a brasileira comprovam a necessidade das pessoas de terem um aparelho celular por perto para diferentes finalidades, sendo elas desde de otimização do tempo até o lazer para fugir da correria do dia-a-dia (BRASIL ECONÔMICO, 2019).

2.1.2 Teoria das filas

“Esperar por um serviço faz parte da vida cotidiana da maioria das pessoas. (...) Mesmo que a espera não possa ser completamente eliminada seu estudo pode proporcionar a redução dos impactos negativos da mesma”. (CAMPELO e CAVALCANTI, 2015, p.1)

Segundo Filho (2016, p.2), citado por Barbosa (2009), “Define-se fila como uma sequência de pessoas que se colocam umas atrás das outras, pela ordem cronológica de chegada a um ponto de embarque”. (FILHO, 2016, p.2)

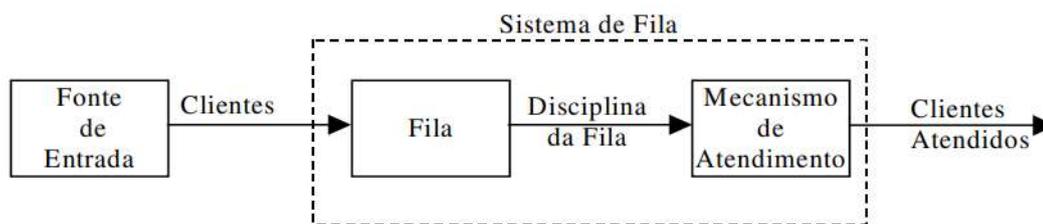
FIGURA 2 - REPRESENTAÇÃO DA TEORIA DAS FILAS



FONTE: Adaptado de CAMARGO, 2008

Normalmente as filas são resultado da falta de planejamento, mesmo não deliberado. Se fosse possível planejar quantos clientes e quais serviços seriam realizados no dia, também seria possível evitar completamente a espera do cliente, dispensando as filas (CAMPELO e CAVALCANTI, 2015 apud TORRES,1966). A FIGURA 3 mostra, em forma diagramada, como os clientes de um serviço são administrados ao longo do tempo numa fonte de entrada. Ao entrar no sistema os clientes se unem a uma fila, em determinado momento um membro da mesma é selecionado para receber o serviço, mediante uma regra conhecida como disciplina da fila. Então, o cliente é atendido em um mecanismo de recepção, que após concluído o libera e o mesmo é retirado do sistema (CAMPELO e CAVALCANTI, 2015 apud HILLIER e LIEBERMAN, 2006).

FIGURA 3 - DIAGRAMA EM BLOCOS DE UM SISTEMA DE FILAS



FONTE: Adaptado de CAMPELO e CAVALCANTI, 2015.

Sobre a fonte de entrada, de acordo com Campelo e Cavalcanti (2015):

“Uma característica da fonte de entrada é seu tamanho. O tamanho é o número total de clientes potenciais que podem requerer serviço em algum momento. A capacidade máxima de clientes no sistema poderá ser limitada por questões de espaço, custo ou para evitar um tempo de espera muito longo.” (CAMPELO E CAVALCANTI, 2015, p.1)

A fila é onde os clientes aguardam antes de receber o serviço desejado, e a ordem em que os clientes são selecionados para receber este benefício é denominada disciplina de fila. A mais comumente utilizada é a disciplina FIFO (*first in, first out*) em que o primeiro a entrar é o primeiro a sair, porém além destas, ainda existem outras como o último a entrar e primeiro a sair (LIFO - *last in, first out*) e em ordem aleatória (SIRO – *service in random order*). Outro fator de seleção de atendimento do cliente é o de prioridade, que pode ser em função de tempo de espera para o atendimento, tamanho do cliente (pacote de mensagem, exemplo: maior primeiro, menor primeiro), maior sensibilidade a atrasos e finalmente, de qualidade de serviço (CAMPELO e CAVALCANTI, 2015).

Por último, existem os mecanismos de atendimento que, segundo Campelo e Cavalcanti (2015), referem-se ao tempo gasto por cada cliente em canais de serviços, que são denominados de tempo de serviço. Com este tempo obtido, os resultados podem ser utilizados nos diversos fatores matemáticos envolvidos no estudo da teoria das filas como o exponencial e o de Poisson, que não serão abordadas neste trabalho.

“A teoria das filas é um processo estocástico, onde as atividades de chegada e atendimento estão governadas por uma distribuição de probabilidade. É um ramo da probabilidade que estuda a formação de filas, através de análises matemáticas precisas e propriedades mensuráveis das filas, e modela previamente o comportamento de um sistema, afim de dimensionar de forma a satisfazer os clientes e evitar desperdícios econômicos para o provedor do serviço.” (CAMPELO E CAVALCANTI, 2015, p.3).

2.1.3 Perda de tempo para o ser humano

De acordo com Donald A. Norman (2008), a respeito da psicologia da espera em filas:

“Filas são bem estudadas no campo de operações (...), mas nesses estudos, a ênfase está na matemática da eficiência: qual é o melhor esquema para lidar com clientes com o menor custo? Quantos funcionários alguém deve contratar para lidar com um número esperado de clientes, cada um com uma taxa probabilística esperada de chegada e alguma distribuição de probabilidade da quantidade de tempo que levará para ser atendido. Tudo isso é realmente necessário, mas falta o elemento humano; como é a experiência para o cliente e o funcionário. A ênfase geral na matemática resultou nas experiências horríveis que temos hoje, onde custo e eficiência são as métricas críticas e a justiça, a equidade e a experiência das pessoas são ignoradas (...) A experiência é mais importante que a eficiência para a sobrevivência do negócio.” (NORMAN, 2008, p.1-2).

Vale mencionar que a experiência com os clientes é importante para todos os serviços, não apenas para os que dependem de filas de espera. Infelizmente, a margem matemática deixada na Teoria das filas se prende muita à eficiência, esquecendo as emoções dos clientes. Cliente com fome não quer esperar; cliente com pressa não tem paciência; e, o pior de todos, cliente irritado não quer saber de ser educado. Emoções são dominantes e marcantes, por exemplo, caso um cliente tenha passado por uma experiência ruim com um serviço, a raiva ficará marcada e as chances de ele realizar o mesmo serviço no mesmo lugar são quase nulas; emoções são contagiantes, faça as pessoas ficarem alegres e sorrirem e todas as pessoas a sua volta estarão contentes e sorridentes. Tenha uma pessoa irritada e impaciente e essas emoções irão se espalhar (NORMAN, 2008, p.2-4).

“É certo que as diversas questões que cercam nosso cotidiano demandam algum tempo para ser solucionadas, o que nos leva a afirmar que é perfeitamente normal “perder” ou “investir” nosso tempo para tratar das questões do dia-a-dia (...) Mas, quais são os efeitos que sofremos quando a solução de simples demandas de consumo requer tempo considerável, extravasando os limites da razoabilidade? Como vem ocorrendo, é razoável exigir do consumidor que perca um tempo precioso para solucionar questões dessa natureza, quando ao mesmo tempo há outros afazeres e problemas mais sérios a solucionar no decorrer do dia?” (GUGLINSKI, 2012, p.1)

Pode-se dizer que o tempo livre passou a ser muito mais valorizado nos últimos anos, podendo até chegar a ser considerado crime o mau atendimento ao consumidor que gere a perda de tempo útil. Como citado por Guglinski (2012), já

mencionado por Moutinho (s.d.):

“O tempo é hoje um bem jurídico e só o seu titular pode dele dispor. Quem injustificadamente se apropria deste bem causa lesão que, dependendo das circunstâncias pode causar dano que vai além do simples aborrecimento do cotidiano, ou seja, dano moral” (GUGLINSKI, 2012, p.1, apud MOUTINHO, s.d.)

Ao contrário de antigamente, hoje o tempo é considerado algo valioso, caro, precioso e gastá-lo indevidamente é uma afronta ao ser humano. Tempo permite as pessoas realizarem seus afazeres, descansarem ou até mesmo, não fazer nada. Logo, os serviços devem ser bem pensados para não gastar o tempo de seus clientes, evitando dores de cabeça e melhorando a qualidade de vida de todos, afinal, ninguém mais tem tempo de sobra para perder em filas (BASTOS, 2017).

2.2 FUNDAMENTAÇÃO TEÓRICA DAS TECNOLOGIAS

Esta seção tem por objetivo apresentar aos leitores os conceitos teóricos, referentes as tecnologias utilizadas para o desenvolvimento do aplicativo Easy Wash.

2.2.1 Unified Modeling Language – UML

A partir dos anos 70, com a evolução da capacidade dos computadores, tornou-se possível construir sistemas mais robustos e complexos e que as aplicações passaram a ter um papel crucial dentro das organizações (MARTINS, 2007).

Suponha um grupo de 3 pessoas em que cada um seja solicitado a descrever um mesmo sistema. Pode-se afirmar que haverá três visões diferentes do mesmo software.

A partir disso, notou-se a necessidade de modelar a estrutura de um software, que fosse um guia para sua construção, assim como há modelos para construção de equipamentos na engenharia.

Deste modo, na década de 90 Grady Booch, Jim Rumbaugh e Ivar Jacobson criaram uma linguagem para modelagem de sistemas, através de convenções utilizadas a época (LIMA, 2013).

A UML (Linguagem de Modelagem Unificada) é uma linguagem-padrão para descrever e documentar projetos de software. Ela, pode ser usada para visualizar,

especificar, construir e documentar os artefatos de um sistema de software-intensivo (BOOCH, 2006).

Segundo Larman (2005) a UML (Linguagem de Modelagem Unificada) pode ser aplicada de três maneiras:

- Como rascunho, onde os diagramas são esboçados à mão;
- Como planta de software, onde os diagramas podem ser utilizados como codificação do sistema ou engenharia reversa;
- Linguagem de programação, na qual a especificação completa de um sistema de software em UML, contempla todo o comportamento analisado, sendo gerada em forma de executável.

Para um melhor entendimento referente a UML, segundo Martins (2007), deve-se conhecer seus elementos básicos que são compostos por: coisas, relacionamentos e diagramas.

- Coisas: Representam os elementos básicos que compõem o sistema, que estão atrelados a sua estrutura, comportamento ou anotações. Como exemplo, pode-se citar as classes, objetos, telas e seus respectivos estados dentro da aplicação;
- Relacionamentos: Referem-se aos vínculos entre as coisas, podendo ser divididas em 4 tipos:
 - Associação: Utilizado para demonstrar um relacionamento estrutural, representando uma ligação cardinal entre as partes, como por exemplo: um professor possui várias disciplinas. Nos diagramas geralmente é representado por uma linha contínua;
 - Agregação/Composição: Relacionamento entre 2 coisas e muito similar a associação, havendo uma dependência entre as partes. Na agregação uma precisa da outra, mas a outra não necessariamente precisa dela para existir. Na composição uma não existe sem a outra, como por exemplo: itens de uma nota fiscal não existem sem a nota fiscal e a nota fiscal não existe sem seus itens;
 - Generalização: Empregada para representar a abstração ou herança de características entre 2 coisas.

- Dependência: Relacionamento em 2 coisas, onde a ocorrência de alteração em uma reflete diretamente na outra. Representada através de uma linha pontilhada;
- Diagramas: São os agrupamentos entre as coisas e os relacionamentos. Através destes, é possível visualizar graficamente um sistema por diversas perspectivas. A seguir, segue a definição de Martins (2007), para os principais diagramas definidos na UML:
 - Diagrama de Caso de Uso: Utilizado para mapear as funcionalidades de um sistema, pelo ponto de vista do usuário;
 - Diagrama de Classe: Um dos diagramas mais utilizados em projetos de software. Seu principal objetivo é demonstrar classes, interfaces e seus respectivos relacionamentos;
 - Diagrama de Sequência: Representa de modo mais aprofundamento o fluxo do sistema para atender um determinado caso de uso. É composto também pelas mensagens trocadas entre os objetos ao longo da sua execução;
 - Diagrama de Estados: Utilizados para representar os estados de um objeto ou classe dentro da aplicação;
 - Diagrama de Pacote: Revela a arquitetura de um sistema mostrando o agrupamento de suas classes;
 - Diagrama de Implementação: Demonstra os componentes e artefatos em relação ao local em que eles são utilizados no sistema.

2.2.2 Banco de dados

Um banco de dados é uma coleção de informações ou dados estruturados que se relacionam entre si para atender determinado objetivo específico (ELMASRI, 2011).

Em meados de 1960, as primeiras aplicações armazenavam suas informações em arquivos. Porém, com a expansão delas, tanto em questões de complexidade, quanto em quantidade de usuários, esse tipo de solução acabou trazendo uma série de limitações e desvantagens, tais como:

- Problema para garantir a segurança dos dados, pois muitos usuários acabam acessando o mesmo arquivo para obtenção das informações;
- Velocidade, devido transações de escrita e leituras de arquivos exigirem muito dos sistemas operacionais;
- Limitação quanto ao relacionamento das informações.

Com base nessas dificuldades, além do advento do paradigma da programação orientada a objetos começaram a surgir novas soluções. Uma delas foram os bancos de dados relacionais (ELMASRI, 2011).

Resumidamente, seu funcionamento é realizado pela criação de tabelas que são um conjunto específico de informações e que são dispostas em colunas e linhas.

As colunas possuem um nome e correspondem a informação que será armazenada. Já as linhas consistem em uma sequência de valores, sendo um valor para cada coluna da tabela.

Para a utilização dessa abordagem necessita-se de um SGBD e há vários disponíveis no mercado. Segundo o *ranking* publicado pelo site *db-engines* (2019), na FIGURA 4 estão os principais SGBD relacionais utilizados ao redor do mundo, em levantamento realizado em outubro de 2019.

FIGURA 4 - BANCO DE DADOS RELACIONAIS MAIS UTILIZADOS AO REDOR DO MUNDO

Rank			DBMS	Database Model	Score		
Oct 2019	Sep 2019	Oct 2018			Oct 2019	Sep 2019	Oct 2018
1.	1.	1.	Oracle +	Relational, Multi-model	1355.88	+9.22	+36.61
2.	2.	2.	MySQL +	Relational, Multi-model	1283.06	+3.99	+104.94
3.	3.	3.	Microsoft SQL Server +	Relational, Multi-model	1094.72	+9.66	+36.39
4.	4.	4.	PostgreSQL +	Relational, Multi-model	483.91	+1.66	+64.52
5.	5.	5.	IBM Db2 +	Relational, Multi-model	170.77	-0.79	-8.91
6.	6.	6.	Microsoft Access	Relational	131.18	-1.53	-5.62
7.	7.	7.	SQLite +	Relational	122.62	-0.74	+5.88
8.	8.	↑9.	MariaDB +	Relational, Multi-model	86.77	+0.71	+13.64
9.	9.	↑10.	Hive +	Relational	84.74	+1.64	+23.64
10.	10.	↓8.	Teradata +	Relational, Multi-model	78.74	+1.78	+0.11
11.	11.	↑12.	FileMaker	Relational	56.67	-1.47	+0.63
12.	12.	↓11.	SAP Adaptive Server	Relational	55.84	-0.26	-2.73
13.	13.	13.	SAP HANA +	Relational, Multi-model	55.35	-0.04	+0.98
14.	14.	14.	Microsoft Azure SQL Database	Relational, Multi-model	27.51	-0.03	+1.24
15.	15.	15.	Informix	Relational, Multi-model	26.00	+0.60	-0.24
16.	16.	↑20.	Google BigQuery +	Relational	25.63	+1.08	+8.96
17.	17.	↓16.	Vertica +	Relational, Multi-model	22.51	-0.04	+1.15
18.	↑19.	↓17.	Firebird	Relational	21.40	+1.43	+1.42
19.	↑20.	19.	Amazon Redshift +	Relational	20.67	+0.80	+3.26
20.	↓18.	↓18.	Netezza	Relational	19.88	-0.44	+2.03

Fonte: Adaptado de db-engines (2019)

2.2.2.1 MySQL

O MySQL é um dos sistemas gerenciadores de banco de dados mais populares do mundo como visto anteriormente. Segundo a Oracle Corporation (ORACLE, 2019), que é a empresa que mantém e distribui o mesmo explica que a ferramenta é altamente recomendada para pequenas ou grandes aplicações. A seguir, algumas de suas vantagens para sua utilização:

- Alto desempenho e escalável;
- Integrado e seguro, garantindo a atomicidade dos dados;
- Diversas ferramentas para facilitar o trabalho de desenvolvedores e DBA'S como ferramentas visuais e vários drivers para tornar compatível com a grande maioria das linguagens de programação disponíveis no mercado;
- Compatibilidade com diversos sistemas operacionais.

De acordo com a FIGURA 5, seguem algumas empresas que utilizam o MySQL, como recurso de armazenamento de dados para suas aplicações.

FIGURA 5 - EMPRESAS AO REDOR DO MUNDO QUE UTILIZAM O MYSQL

Powered by MySQL		
Web/End-User	Embedded/ISV	Telecom
<ul style="list-style-type: none"> • Alibaba • Airbnb • Disney • Facebook • Google • Tencent • Twitter • Visa • Wells Fargo • YouTube • Uber • Zappos 	<ul style="list-style-type: none"> • Adobe • Airbus • EMC • F-Secure • F5 Networks • Hewlett Packard • IBM • Intel • Intuit • NetApp • Palo Alto Networks • Symantec 	<ul style="list-style-type: none"> • AT&T • Cisco • Comcast • Ericsson • France Telecom • Huawei • Italtel • Nokia • Oi Telecom • Siemens • Telenor • Teligent

FONTE: MySQL (2019)

2.2.3 React

React é uma biblioteca Javascript utilizada para criação de interfaces mais amigáveis e interativas aos usuários para páginas web (RISINGSTACK, 2019).

Sua criação inicia-se em 2011, pelos desenvolvedores do Instagram e que foi sendo aprimorada após a sua venda para o Facebook.

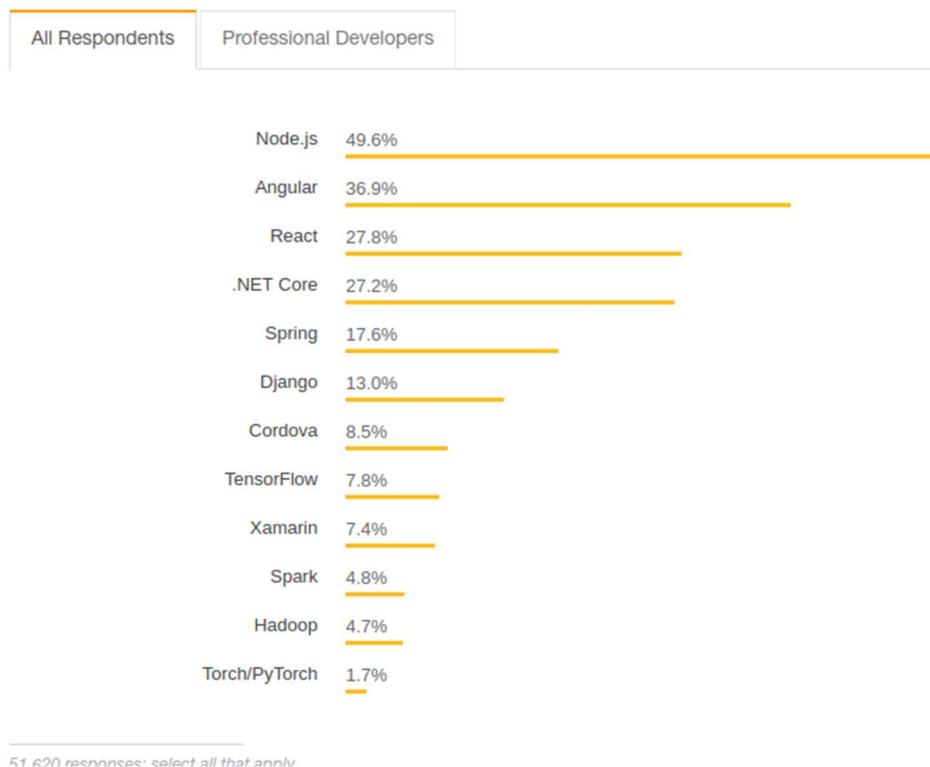
Foi desenvolvida devido principalmente a dificuldades que analistas e desenvolvedores estavam enfrentando para expansão e manutenção de seu código. E essas adversidades aconteceram pelo crescimento rápido da equipe de desenvolvimento e da própria aplicação que foi ganhando vários recursos ao longo do tempo (RISINGSTACK, 2019).

A biblioteca tem por principal conceito a criação de componentes reutilizáveis. Por exemplo, um item de uma página web, pode ser encapsulado, possuindo sua própria estrutura, estilo e comportamento customizados. Com isso, ela pode ser codificada uma única vez e ser reaproveitada por toda a aplicação. Essa abordagem traz uma grande produtividade principalmente em questões de manutenção, pois qualquer mudança ou alteração é realizada somente em um local, sendo posteriormente replicada em todos os demais onde é utilizada (RISINGSTACK, 2019).

De acordo com a pesquisa realizada pelo Stack Overflow (2019), o React foi a terceira biblioteca ou framework mais pesquisado em seu site em 2019, conforme FIGURA 6.

FIGURA 6 - BIBLIOTECAS, FRAMEWORKS E FERRAMENTAS MAIS POPULARES NO STACK OVERFLOW

Frameworks, Libraries, and Tools



FONTE: Stack Overflow (2019)

2.2.4 React Native

Em 2015, o Facebook lançou o React Native que é uma biblioteca para criação de interfaces móveis Android e IOS utilizando somente Javascript. Assim, ela possibilita que um mesmo código funcione para aparelhos que operem tanto sob a sistema operacional Android, quanto para dispositivos IOS. Sua arquitetura e utilização em si é muito semelhante com biblioteca React.js. No entanto, uma das suas poucas diferenças é que as estruturas dos componentes não são realizadas com HTML, mas sim com JSX, combinado com Javascript puro (FACEBOOK, 2019).

2.2.5 Linguagem de programação Java

Em 1992, a Sun Microsystems deu início ao desenvolvimento da linguagem Java, por meio de uma equipe interna da empresa que ficou conhecida como Green

Team e que era liderada, pelo então desenvolvedor canadense James Gosling (ORACLE, 2019).

A ideia inicial da linguagem era permitir que um mesmo código pudesse ser compartilhado pela maioria dos aparelhos eletrônicos da época, como televisores, videocassetes e outros pequenos dispositivos portáteis.

Contudo, no mesmo período iniciou-se também a popularização dos navegadores *web*, os quais permitiam o acesso à internet. Logo, a Sun viu que poderia aproveitar a nova onda e difundir a sua mais nova criação. Assim em 1995, foi anunciado que a linguagem seria suportada no navegador NetScape e partir de então, passou a dominar o mercado (ORACLE, 2019).

A linguagem trouxe várias soluções para problemas relacionados a tecnologias da época, pode-se citar como exemplo, o gerenciamento de memória, de ponteiros, compartilhamento de bibliotecas e a capacidade de rodar a mesma aplicação em diversos dispositivos com sistemas operacionais diferentes.

Em 2009, a Oracle Corporation adquiriu a Sun Microsystems e no momento a empresa que mantém e distribui a tecnologia.

Segundo a sua própria detentora, a tecnologia Java é recomendada para sistemas de médio e grande porte e está presente atualmente desde sistemas corporativos a aplicações de Big Data, jogos e dispositivos móveis.

A seguir segue algumas informações interessantes referentes a tecnologia Java, retirada de sua página na web (ORACLE, 2019):

- 97% dos computadores em todo o mundo executam o Java;
- 3 bilhões de celulares executam Java;
- 9 milhões de desenvolvedores ao redor do mundo;
- 125 milhões de televisores executam Java.

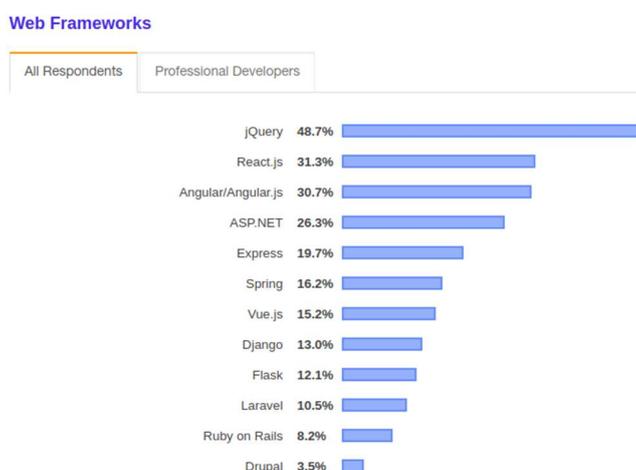
2.2.6 Spring

O Spring é uma família de projetos de código aberto que contempla um conjunto de módulos com a finalidade de facilitar a criação de aplicações, seguindo a normativa Java EE (PIVOTAL, 2019). O Spring possui soluções que contemplam *big data*, como o Spring Data até questões de segurança, através do projeto Spring Security.

Vale ressaltar que o Spring não é um concorrente do Java EE, mas sim um complemento, devido ao *framework* implementar muitos recursos da especificação do mesmo (PIVOTAL, 2019).

Por se tratar de um framework de código aberto, possui uma grande comunidade de desenvolvedores que fornecem avaliações e propõem melhorias de modo constante. Conforme pesquisa elaborada pelo Stack Overflow (STACK OVERFLOW, 2019), o Spring é o *framework* Java mais popular no mercado para aplicações *web*, conforme FIGURA 7.

FIGURA 7 - FRAMEWORKS WEB MAIS POPULARES NO STACK OVERFLOW EM 2019



Fonte: STACK OVERFLOW (2019)

2.2.6.1 Spring Framework

O *Spring* é um *framework* de código aberto desenvolvido por Rod Johnson, para a criação de aplicações seguindo a normativa Java EE.

Sua criação foi motivada pela exigência de grandes configurações para poder iniciar um projeto Java EE. Em muitos casos, por mais simples que seja o projeto, essa solução requer muitas dependências e configurações que as vezes nem são utilizadas e que por consequência gera uma aplicação muito robusta, a qual acaba sendo muito difícil de manipular e testar localmente.

Uma das principais características do framework é que o mesmo é baseado no padrão de projeto de injeção de dependência. Este que utiliza algum tipo de controle para o gerenciar as dependências entre classes de um projeto, não sendo necessário mais que esse controle seja realizado pelo programador.

De acordo com a documentação do Spring (PIVOTAL, 2019), para o bom uso

da tecnologia, além de saber sua definição é importante também conhecer seus conceitos, que são:

- Flexibilidade para alterações de arquitetura e configuração do projeto sem alterar seu código;
- Mantem uma forte compatibilidade com suas versões anteriores. O *Spring* suporta várias bibliotecas e versões do JDK, que facilitam a manutenção da aplicação ao longo do tempo;
- Alto padrão de qualidade de código, pela construção de um javadoc preciso e significativo. Assim é possível criar um código limpo e bem documentado, sem a necessidade de adição de bibliotecas externas ou outros recursos para atingir esse objetivo.

2.2.6.2 *Spring Boot*

O *Spring Boot* é mais um projeto disponível dentro do ecossistema *Spring* e que nos permite criar projetos baseados no framework *Spring* de uma maneira muito mais rápida (PIVOTAL, 2019).

Conforme sua documentação de referência no GitHub (GITHUB, 2019), os principais objetivos desse projeto são:

- Fornecer uma rápida e acessível introdução para o desenvolvimento de projetos, utilizando o Spring;
- Disponibilizar uma série de recursos facilmente, como por exemplo, servidores, medidores de métricas, verificações de integridade e segurança, entre outros;
- Permitir a criação de uma aplicação sem a necessidade de nenhum arquivo para configuração.

Além destes objetivos, sua utilização para criação de projetos Java, traz consigo uma série de vantagens, principalmente se tratando de produtividade no desenvolvimento. Sua utilização já adiciona gratuitamente um nível de segurança, qualidade e arquitetura interessantes. Assim os desenvolvedores, podem utilizar seu

tempo em questões que realmente importam, como por exemplo a lógica de negócio.

Para iniciar uma aplicação, nenhuma configuração manual é necessária, pois o próprio *framework* realiza isso, baseando-se em convenções. Essas convenções nada mais são do que o encapsulamento de bibliotecas para atender determinado objetivo e que são compatíveis entre si. Um bom exemplo disso é quando há a necessidade de realizar a configuração de um banco de dados para a aplicação. Quando adicionado o “spring-boot-starter-data-jpa” como dependência do projeto, ele já configura a persistência dos dados da aplicação.

2.2.7 Metodologia Ágil

Com o passar dos anos, principalmente com o avanço da tecnologia, percebe-se o aumento da competitividade entre as empresas e necessidade de estar sempre entregando resultado com mais qualidade e com menos recursos (MARTINS, 2007).

Sendo assim, a metodologia tradicional já não era mais suficiente e a partir dos anos 90, várias empresas começaram a criar e inserir novos conceitos nesses processos. Notou-se que com essas ações houve uma série de melhorias significativas no desenvolvimento de projetos (ALLIANCE, 2019).

A partir desse momento passam a surgir vários frameworks de metodologias e processos no mercado, como *Extreming Programming*, *Scrum*, *Feature Driven Development*, entre outros (ALLIANCE, 2019).

Em 2001, um grupo de profissionais, avessos a metodologia tradicional se reuniram nos Estados Unidos para discutir melhores práticas para a construção de softwares. Nesse momento é publicado o Manifesto Ágil o qual possui 4 valores essenciais, os quais são:

“Indivíduos e interações, mais do que processos e ferramentas; Software em funcionamento, mais do que documentação abrangente; Colaboração com o cliente, mais do que negociação de contratos; Responder a mudanças, mais do que seguir um plano.” (MANIFESTO ÁGIL, 2019).

Cada método ágil contém os valores e princípios enraizados no Manifesto Ágil. Em complementação, os 12 princípios do software ágil, de acordo com o Manifesto Ágil (2019) são:

- A maior prioridade é satisfazer o cliente, através da entrega adiantada e contínua de *software* de valor;
- Aceitar mudanças de requisitos, mesmo no fim do desenvolvimento. Processos ágeis se adequam a mudanças, para que o cliente possa tirar vantagens competitivas;
- Entregar *software* funcionando com frequência, na escala de semanas até meses, com preferência aos períodos mais curtos;
- Pessoas relacionadas a negócios e desenvolvedores devem trabalhar em conjunto e diariamente, durante todo o curso do projeto;
- Construir projetos ao redor de indivíduos motivados. Dando a eles o ambiente e suporte necessário, e confiar que farão seu trabalho;
- O método mais eficiente e eficaz de transmitir informações para, e por dentro de um time de desenvolvimento, é através de uma conversa cara a cara;
- *Software* funcional é a medida primária de progresso;
- Processos ágeis promovem um ambiente sustentável. Os patrocinadores, desenvolvedores e usuários, devem ser capazes de manter indefinidamente, passos constantes;
- Contínua atenção à excelência técnica e bom *design*, aumenta a agilidade;
- Simplicidade: a arte de maximizar a quantidade de trabalho que não precisou ser feito;
- As melhores arquiteturas, requisitos e *designs* emergem de times auto organizáveis;
- Em intervalos regulares, o time reflete em como ficar mais efetivo, então, se ajustam e otimizam seu comportamento.

2.2.7.1 Scrum

O termo *Scrum* surgiu em um artigo escrito por Takeuchi e Nonaka na Harvard Business Review em 1986, onde descrevem equipes de projeto compostas de pequenas equipes multifuncionais, trabalhando com sucesso rumo a um objetivo comum, que os autores compararam à formação *scrum* do *rugby*. (PHAM, 2011)

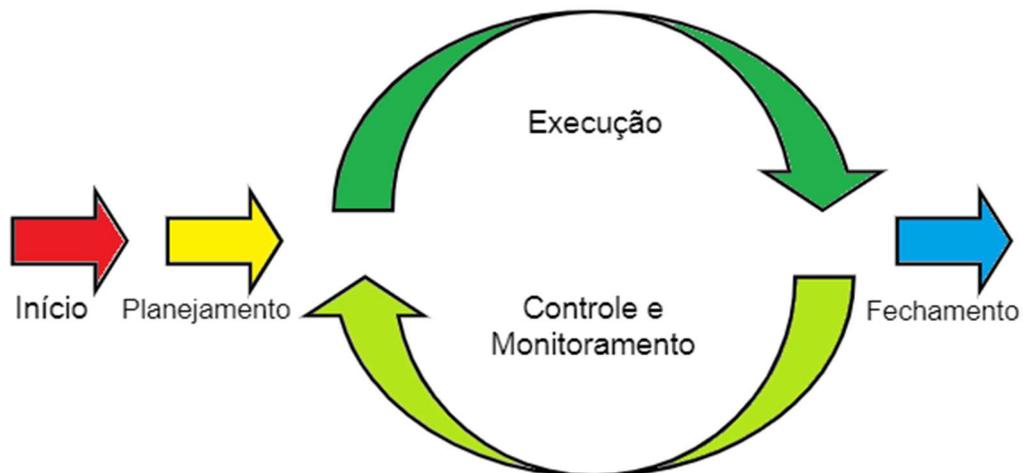
Scrum é o *framework* ágil mais popular nos processos de desenvolvimento de software. O *framework* junta as práticas de mercado já conhecidas de maneira organizada e eficiente. A proposta é uma nova estrutura de gestão de software, que

está baseada em auto-organização, motivação, propriedade e orgulho de uma equipe em realizar suas aquisições (SABBAGH, 2013).

Segundo SCHWABER (2004) a gestão do projeto é dividida em três papéis:

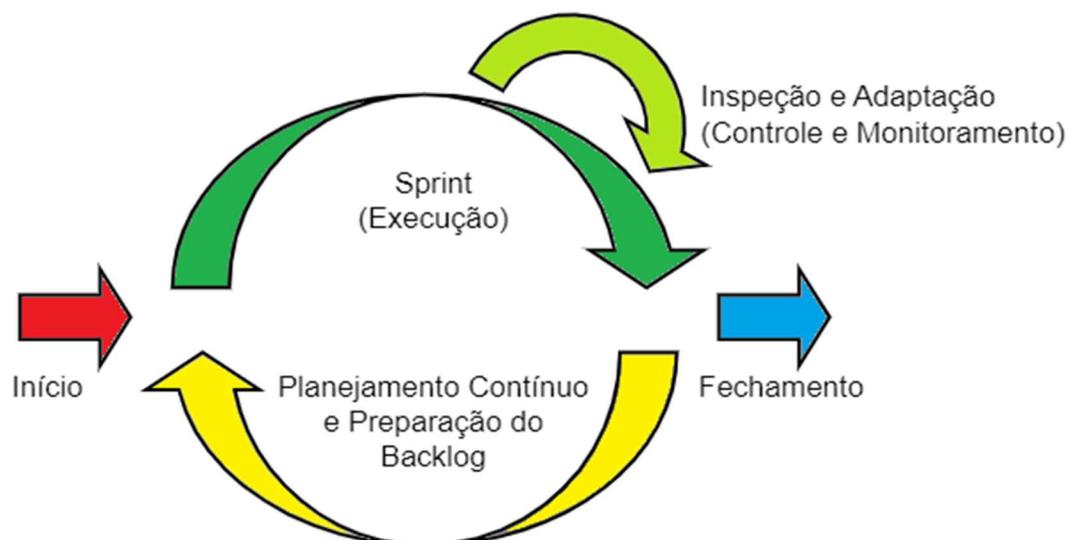
- *Product Owner* (PO): Representa a importância de todos que têm participação no projeto assim como o sistema resultante através do levantamento de requisitos gerais do programa, essa lista é chamada de *Product Backlog*. O PO também deve prover fundos iniciais e contínuos para o desenvolvimento do sistema e o retorno do investimento aplicado;
- *Scrum Master* (SM): Responsável por ensinar e aplicar os processos do *Scrum*, adequando a cultura de da organização para que os benefícios esperados sejam atingidos e garantindo que o framework seja efetivo;
- Equipe de desenvolvimento: Responsáveis por construir as funcionalidades projetadas e possuem autonomia para se auto gerenciar com habilidades multifuncionais, podendo também transformar o *Product Backlog* em um incremento das funcionalidades do produto.

FIGURA 8 - PROCESSO DE GESTÃO DE PROJETOS TRADICIONAL



FONTE: Adaptada de Pham (2011)

FIGURA 9 - PROCESSO DE GESTÃO DE PROJETOS TRADICIONAL



FONTE: Adaptada de Pham (2011)

O fluxo do *Scrum* é dividido em *sprints* e cada uma delas é formada por trinta dias consecutivos de interação, podendo variar de acordo com a necessidade de cada projeto. Cada *sprint* é iniciada através da reunião de planejamento de todo o ciclo, onde o *Project Owner* e a equipe definem as atividades a serem desempenhadas na mesma, considerando as principais prioridades (SCHWABER, 2004).

Essas reuniões são divididas em duas partes. Na primeira parte, o *Project Owner* apresenta à equipe as prioridades a serem desenvolvidas e a equipe debate a viabilidade e circunstâncias do desenvolvimento a ser realizado. Na segunda parte, a equipe planeja a *sprint* a ser desenvolvida (SCHWABER, 2004).

A *scrum* diária é a reunião que ocorre durante todos os dias da *sprint*, onde cada membro deve responder às três questões:

- O que foi realizado desde a última *scrum* diária?
- O que será feito entre a *scrum* diária atual e a próxima?
- Quais dificuldades estão enfraquecendo o comprometimento com a *sprint* atual e com o projeto?

Com base nas respostas desses questionamentos, o *Scrum Master* e todo o grupo podem mensurar os progressos da atividade e também se o objetivo da *sprint* será alcançado (SCHWABER, 2004).

2.3 COMPARATIVO COM APLICAÇÕES SEMELHANTES

Para construção deste trabalho, foram pesquisados aplicativos similares disponíveis no mercado, relacionados ao mesmo serviço proposto pela plataforma Easy Wash.

2.3.1 App Wash

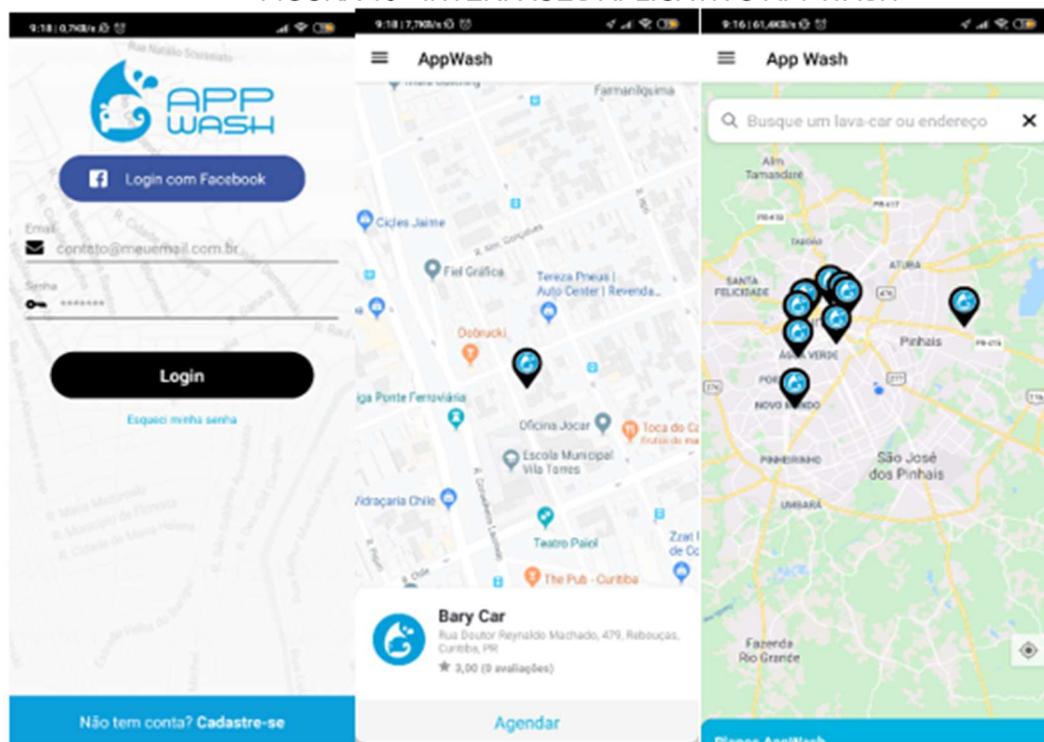
Esse aplicativo (FIGURA 10) permite o agendamento de serviços referentes a lavagem de veículos, onde basicamente seu automóvel pode realizar lavagens em lava cars conveniados com a plataforma. Suas principais características e funcionalidades são:

- Utilização a partir de assinatura mensal;
- Variedade de serviços disponíveis;
- Sistema de avaliação e classificação dos prestadores de serviços cadastrados no aplicativo.

As principais limitações ou desvantagens encontradas nessa plataforma foram:

- O agendamento e contratação de serviços só podem serem realizados após a assinatura do plano;
- Um cliente não pode levar mais de um veículo para lavagem;
- Caso o cliente não utilize suas lavagens disponíveis para o mês, está não é acumulada para o mês seguinte;
- Atuação apenas em Curitiba.

FIGURA 10 - INTERFACES APLICATIVO APPWASH



FONTE: APP WASH (2020)

2.3.2 LAVÔ

Este aplicativo também permite a lavagem de veículos. A empresa possui um modelo de negócio na qual a mesma possui total controle, referente aos seus prestadores.

Esse fato ocorre, pois, a empresa oferece todo o treinamento, suporte e material para realização da lavagem através de seus parceiros. Além disso, os lavadores atuam de forma individual e realizam os serviços na própria casa do cliente ou em qualquer outro local indicado pelo mesmo, já que as lavagens são realizadas a seco, com um intuito ecológico e sustentável (LAVÔ,2019).

As principais características e recursos deste serviço, são listados a seguir:

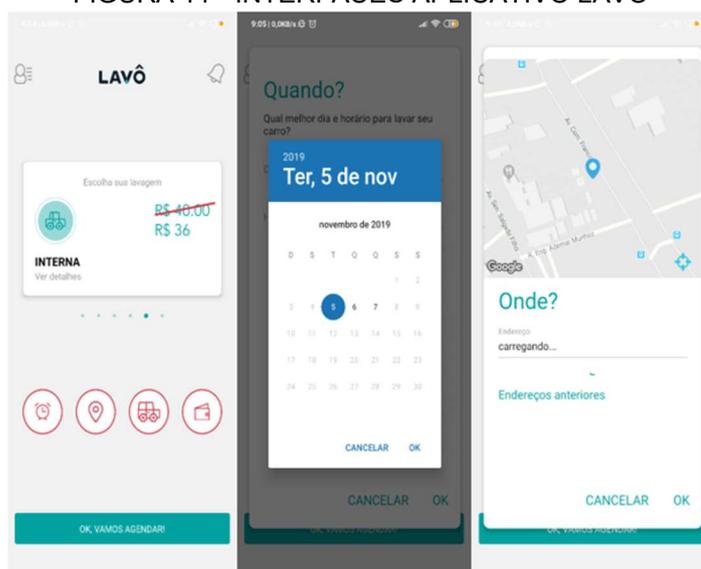
- Serviço sobre demanda, ou seja, o cliente só paga pelo serviço que realiza, não sendo necessário adesão a nenhum plano;
- Cupons de desconto;
- Garantia da padronização e qualidade dos serviços, pois todo o ciclo é controlado pela empresa;
- Possui uma variedade de serviços e preços;
- Pagamento flexível, podendo ser realizado em débito, crédito ou dinheiro;

- Possui facilidade para a entrada de qualquer pessoa para se tornar um parceiro da empresa, como lavador desde que passe na avaliação da mesma;
- Possui prêmios semestrais e permite aos prestadores ganhos com gorjeta e bônus.

A seguir as principais desvantagens encontradas, relacionadas ao aplicativo:

- Lavagem realizada exclusivamente por lavadores parceiros da empresa;
- O agendamento deve ser realizado com no máximo 36 horas de antecedência.

FIGURA 11 - INTERFACES APLICATIVO LAVÔ



FONTE: LAVÔ (2019)

O App Wash, realiza a lavagem através de empresas parceiras, sendo somente uma plataforma de divulgação e não um serviço sob demanda, devido a necessidade da assinatura de um plano mensal.

Já a Lavô, possui um outro enfoque que têm por objetivo controlar e padronizar todo o ciclo de lavagens, não por meio de empresas parceiras, mas sim através de colaboradores individuais.

O principal objetivo com a construção do aplicativo Easy Wash é unificar esses dois conceitos. Ser uma plataforma de comunicação entre clientes e prestadores que são o foco comum desta nova plataforma.

A seguir, o QUADRO 1, realiza um comparativo entre os aplicativos encontrados no Brasil para o mesmo segmento de serviços. Como é possível observar, os 2 aplicativos realizam a mesma atividade, porém com conceitos diferentes. O Easy Wash possui todas as funcionalidades descritas, mostrando-se a melhor opção em relação aos já existentes no mercado.

QUADRO 1 - COMPARATIVO DE APLICATIVOS

Recursos	AppWash	Lavô	Easy Wash
Agendamento	X	X	X
Chat com o lavador		X	X
Lavagem em domicílio		X	X
Lavagem em empresas parceiras	X		X
Serviço com agendamento leva e trás			X
Contemplam as seguintes formas de pagamento (Crédito, Débito, Dinheiro)		X	X
Prestadores definem o preço de seus serviços			X

FONTE: Os autores (2020)

Podemos observar no quadro 1 que o aplicativo Easy Wash oferece mais do que seus concorrentes. O Easy Wash atende a todos os recursos apresentados, trazendo maior flexibilidade, eficácia e eficiência nos serviços prestados. O capítulo a seguir apresenta detalhes dos materiais e metodologias aplicadas.

3 MATERIAIS E MÉTODOS

O sucesso de um projeto consiste na sua entrega e que esta englobe todas as funcionalidades definidas em seu escopo. Além disso, devem atender todas as estimativas de custos e prazo.

Para cumprir todos esses objetivos, deve-se utilizar os processos de software, que consistem em um conjunto de atividades, métodos e práticas que guiem os colaboradores por todo o percurso do projeto.

Outro ponto importante para que o sucesso de um projeto seja atingido é referente as ferramentas utilizadas para a realização do mesmo.

Por fim, o objetivo deste capítulo é descrever todos os processos e ferramentas que foram utilizadas para o desenvolvimento do aplicativo Easy Wash.

3.1 SCRUM

Os métodos utilizados como guia para os processos e atividades deste projeto foram o Scrum, em conjunto com o *Kanban*.

A equipe chegou em consenso pela escolha do Scrum. E essa decisão deve-se ao fato de todos os membros da equipe terem certa familiaridade com este framework tanto pelas experiências adquiridas ao longo da faculdade, quanto pelas profissionais.

Além disso, no momento de definição do tema, também foi entendido que seria necessário a escolha de uma metodologia que permitisse a adaptabilidade a mudanças ao longo do projeto. Era consenso que haveria a necessidade de adaptação dos requisitos ao decorrer da execução, principalmente em questões relacionadas as métricas estipuladas para cada atividade, pois nenhum membro dominava as tecnologias e ferramentas utilizadas. Sendo assim, provavelmente haveria a necessidade de um prazo maior do que anteriormente projetado para concluir alguma tarefa.

Com relação aos eventos que compõem o Scrum, como descrito no Capítulo 2, foi necessário a adaptação, principalmente devido a disponibilidade de cada integrante da equipe.

A seguir, segue como os eventos do Scrum foram adaptados para o projeto:

- *Sprints*: Foram definidas pela equipe para o prazo de 15 dias, sendo entregue um incremento de valor ao final de cada Sprint;
- Reuniões diárias: A equipe conseguiu mantê-las diariamente, eram realizadas no Setor de Educação Tecnológica (SEPT) da Universidade Federal do Paraná, de modo muito informal geralmente entre os intervalos das aulas ou no final da mesma, ou online quando não era possível todos os integrantes da equipe se reunirem;
- Reunião de Planejamento e Revisão da Sprint: A equipe devido a sua disponibilidade, preferiu unificar esses dois encontros em um único, o qual era realizado em conjunto com o Professor Orientador do projeto. Neste encontro, eram entregues os artefatos desenvolvidos, tirar possíveis dúvidas da mesma e organizar os próximos ciclos.

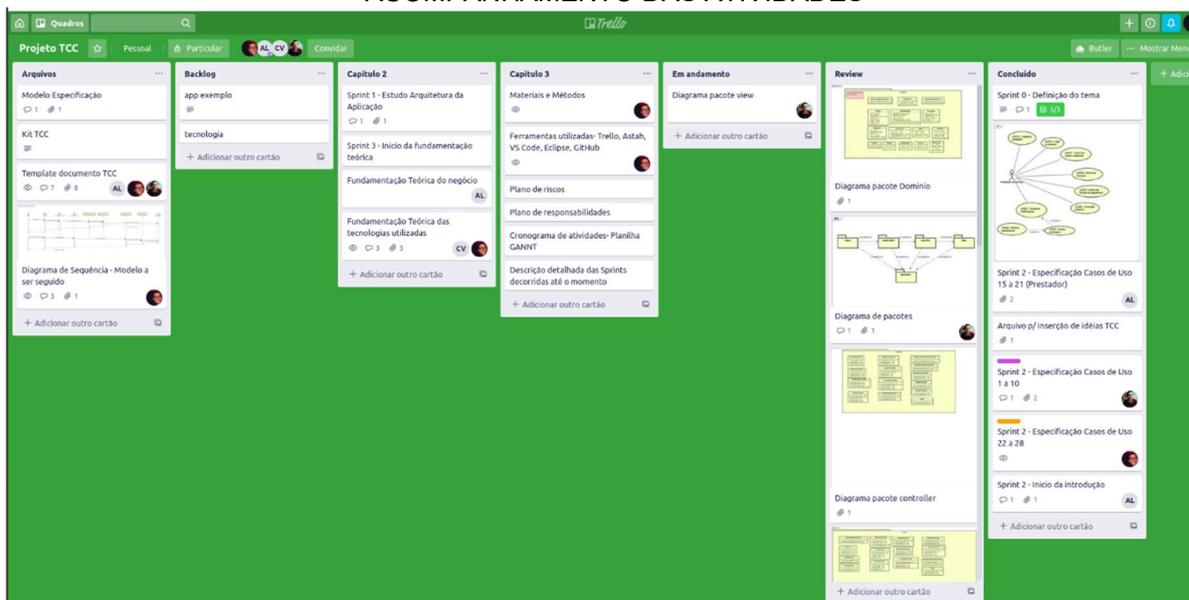
3.2 TRELLO

É uma ferramenta criada pela Fog Creek Software e a partir de 2017 passa a ser mantida e disponibilizada pela Atlassian, incluindo versões gratuitas e pagas. Sua primeira versão foi lançada em 2011, a qual foi criada para atender uma das principais necessidades identificadas no mercado naquele momento, referente a gerenciamento de projetos (ATLASSIAN, 2019).

O Trello auxilia no controle e gerenciamento de quaisquer tarefas, de maneira muito fácil e intuitiva. Possui versões tanto para web quanto para smartphones. De acordo com sua página oficial, em 2014 já possuía a marca de mais de 4.750 milhões de usuários (ATLASSIAN, 2019).

O aplicativo permite o acompanhamento das atividades, utilizando o conceito *Kanban*, criado pelo Toyota para gerenciamento de sua produção. Com sua aplicação é possível criar quadros, listas de tarefas e cartões. A equipe utilizou esta ferramenta para inserir e controlar todas as atividades desenvolvidas durante as *sprints*, somente através deste canal. A FIGURA 12 mostra como a equipe utilizou a ferramenta para atribuição, divisão e controle das atividades realizadas ao longo do projeto.

FIGURA 12 - UTILIZAÇÃO DA FERRAMENTA TRELLO PARA ATRIBUIÇÃO E ACOMPANHAMENTO DAS ATIVIDADES



FONTE: Os Autores (2020)

A equipe realizou a divisão do Trello inicialmente 5 listas que eram as seguintes: *Backlog*, a fazer, em andamento, review e concluído.

No entanto com a evolução das *sprints*, principalmente a partir da terceira, os membros notaram que o canal começou a ficar poluído. Isso pode ser explicado, devido os itens que eram concluídos, estarem constantemente sendo alterados ao longo do projeto, conforme refinamento e amadurecimento dos requisitos. Portanto, não era interessante arquivar, ou esconder esses itens. Ao contrário, era muito mais pertinente mantê-los sempre visíveis para todos os membros da equipe, pois também em alguns casos houve dependência de um item construído durante uma tarefa para a execução do próximo.

Desta maneira, houve criação de outras listas, como por exemplo os capítulos desse documento que facilitaram a visualização, principalmente relacionado ao *backlog* como um todo.

Acerca das sub tarefas de cada lista, a equipe tentou dividi-las em um nível onde fosse possível ser realizadas por um único membro. Além disso, para cada sub tarefa criada também era estipulado um prazo de entrega.

3.3 ASTAH

De acordo com o autor (BOOCH, 2006), o principal produto elaborado por uma

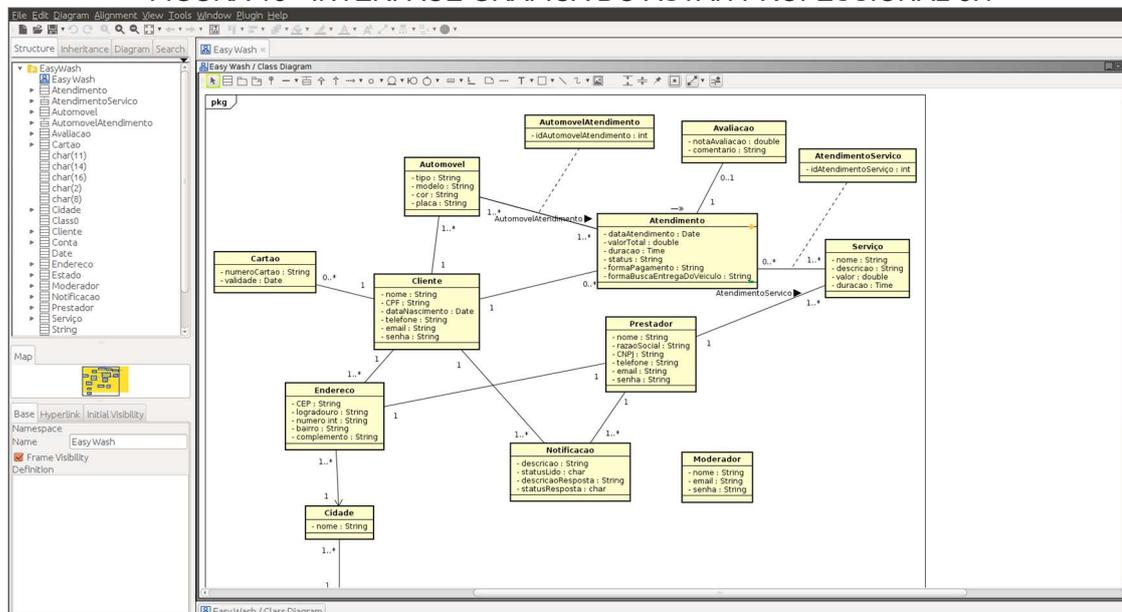
equipe de desenvolvimento, não é uma documentação perfeita ou as melhores linhas de código. O que realmente importa é o *software* em funcionamento e que atendam todas as reais necessidades de seus usuários. Para atingir esse objetivo, há várias técnicas para criar o produto de forma assertiva e com um alto nível de qualidade.

Uma dessas técnicas é através da modelagem de *software* que visa antecipar antes da codificação, tudo o que deverá ser construído. Essa modelagem é um grande auxílio para definir quais as funcionalidades, componentes, arquitetura e fluxos que a aplicação deverá possuir.

Logo, para realizar a modelagem UML do aplicativo Easy Wash a equipe definiu a utilização da ferramenta Astah Professional, na versão 8.1, a qual permite a criação de todos os diagramas UML requeridos para conclusão deste projeto de maneira muito intuitiva.

A ferramenta é paga, no entanto ela possui uma licença para estudantes e professores, a qual permite o total acesso aos seus recursos de maneira grátis. Essa licença possui um prazo de um ano e para obtê-la é necessário utilizar seu e-mail acadêmico ou enviar um comprovante de matrícula obtido junto a sua instituição de ensino. A FIGURA 13, ilustra a interface gráfica do Astah.

FIGURA 13 - INTERFACE GRÁFICA DO ASTAH PROFESSIONAL 8.1



FONTE: Os Autores (2020)

Para o aplicativo Easy Wash, o grupo utilizou o Astah para a criação dos seguintes diagramas:

- Diagramas de Casos de Uso;
- Diagramas de Classe;
- Diagramas de sequência;
- Diagramas de implementação;
- Diagramas de pacote.

3.4 VISUAL STUDIO CODE

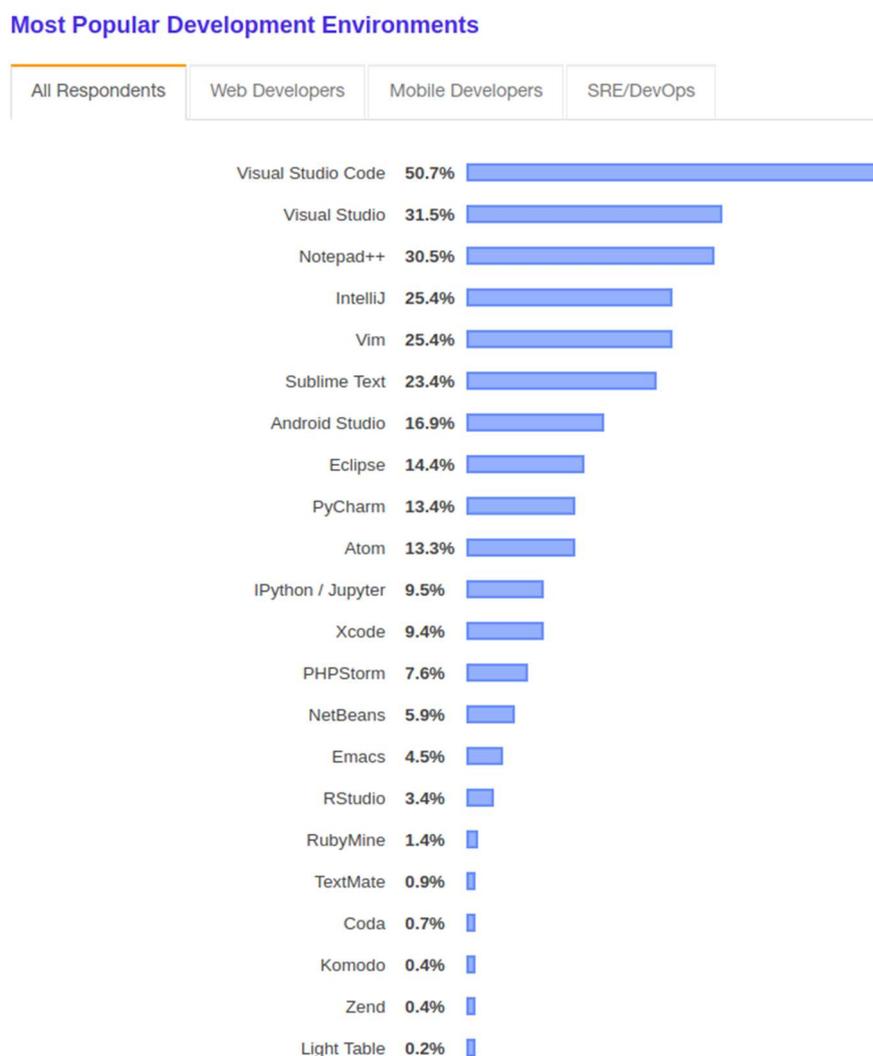
O Visual Studio Code é uma ferramenta para edição de código fonte, criado pela Microsoft em 2015. Este editor oferece suporte para várias linguagens de programação. Conforme sua página oficial na internet (MICROSOFT,2019), ela oferece suporte às seguintes linguagens de forma nativa: Javascript, Node.js e *Typescript* e ainda possui um vasto ecossistema que permite adicionar extensões para oferecer a capacidade de empregar este editor para outras linguagens de programação como Python, C++ e Java.

Ademais, também possui uma série de extensões e facilidades que aumentam muito a produtividade do desenvolvimento. Dentre elas ressaltam-se:

- Auto complete inteligente para funções e variáveis;
- Fácil sincronização com repositórios GIT;
- Permite customizações de acordo com as necessidades e preferências de cada desenvolvedor;
- Disponível para os 3 principais sistemas operacionais (Windows, Linux e Mac).

De acordo com a pesquisa elaborada anualmente pela Stack Overflow (2019) e conforme a FIGURA 14, o Visual Studio Code é o editor de código mais popular entre os usuários da plataforma.

FIGURA 14 - AMBIENTES DE DESENVOLVIMENTO MAIS POPULARES, SEGUNDO A PLATAFORMA STACK OVERFLOW



FONTE: Stack Overflow (2020)

Levando em consideração todas essas vantagens e estatísticas abordadas anteriormente a equipe decidiu utilizar esse editor de código para desenvolver toda a parte *front-end* da aplicação, que será realizada para versões web e móvel, utilizando React.js e React Native respectivamente.

3.5 ECLIPSE IDE

O eclipse é um ambiente de desenvolvimento integrado (IDE), para desenvolvimento de aplicações utilizando a linguagem de programação Java. Além disso, pode ser customizada para criação de projetos que façam uso de outras linguagens de programação, através de extensões.

Sua primeira versão foi criada pela IBM em 2002 e atualmente é mantida e disponibilizada pela Eclipse Foundation (ECLIPSE FOUNDATION, 2019).

Possui uma série de *plug-ins* e extensões que facilitam o desenvolvimento aumentando muito a produtividade. Entre eles, é citado de modo especial os *plug-ins* para adição de suporte a aplicações SpringBoot e também outro para sincronização código fonte com o GitHub, os quais foram utilizados fortemente para a construção do aplicativo Easy Wash.

3.6 GITHUB

O Github é uma plataforma utilizada por desenvolvedores e empresas para hospedagem de código fonte de suas aplicações, baseado no sistema de controle distribuído GIT.

Sua aplicação em projetos de software traz uma série de vantagens. Uma delas é a possibilidade de permitir que vários desenvolvedores atuem no mesmo código fonte (YUILL, 2008). Logo, neste cenário é possível comparar as versões e unificá-las através de um comando conhecido como “*merged*”.

Outra funcionalidade muito importante dessa ferramenta é com relação a rastreabilidade e armazenamento de histórico de um código fonte. Sendo assim, quando uma versão é disponibilizada com erros é possível por exemplo, regredir para versões anteriores e também saber qual programador criou determinada versão ou ainda o que de fato foi alterado na mesma.

O GitHub foi criado por Chris Wanstrath, J. Hyett, Tom Preston-Werner e Scott Chacon em 2009 e em junho de 2018 foi adquirida pela Microsoft.

A plataforma foi aprimorada com o passar dos anos e de acordo com sua página oficial (GITHUB, 2019), em 2019 já passou a marca de 40 milhões de usuários e 100 milhões de repositórios armazenados. Entre eles alguns projetos mundialmente conhecidos, tais como: o sistema operacional Linux, o gerenciar de conteúdo Wordpress e o editor de código Atom.

Levando em consideração todos esses dados citados anteriormente, somados a facilidade de seu uso e sincronização com as outras ferramentas utilizadas durante a realização desse projeto, como o Visual Studio Code e o Eclipse, a equipe optou por realizar todo o versionamento e controle do código fonte construído ao longo do projeto, através do uso desta plataforma.

3.7 INFRAESTRUTURA DE DESENVOLVIMENTO

Para realizar o desenvolvimento deste projeto foram utilizados três notebooks e um desktop com as seguintes especificações:

- Notebook 1:
 - Nome da máquina: ALLAN-PC;
 - Proprietário: Allan Lerner Goldenberg
 - Fabricante: Acer;
 - Sistema Operacional: Windows 10 Pro;
 - Memória RAM: 8 GB;
 - Processador: Intel Core i5 4200U 1.6 GHz;
 - Espaço de Armazenamento: SSD 240GB.
 - Placa de Vídeo: Nvidia GeForce 940MX.
- Notebook 2:
 - Nome da máquina: DPADILHA-PC;
 - Proprietário: Diego Lemos Padilha;
 - Fabricante: Samsung;
 - Sistema Operacional: Ubuntu 18.04.3 LTS;
 - Memória RAM: 8 GB;
 - Processador: Intel Core i7 4210U 2.40GHz;
 - Espaço de Armazenamento: 1000 GB;
 - Placa de Vídeo: NVIDIA GEFORCE 920MX;
- Notebook 3:
 - Nome da máquina: JEAN-PC;
 - Proprietário: Jean de Jesus Freire;
 - Fabricante: Lenovo;
 - Sistema Operacional: Ubuntu 18.04.3 LTS;
 - Memória RAM: 8 GB;
 - Processador: Intel Core i5-6200 CPU @ 2.30GHz;
 - Espaço de Armazenamento: 1000 GB.
- Desktop:
 - Nome da máquina: CLAUDINEI-PC;

- Proprietário: Claudinei Varela;
- Sistema Operacional: Windows 10 Pro;
- Memória RAM: 6 GB;
- Processador: Phenom x4 955 3.2(oc 3.6);
- Espaço de Armazenamento: SSD 120GB + 1TB HD externo;
- Placa de Vídeo: HD 6850.

3.8 PLANO DE RISCOS

Ao iniciar o desenvolvimento de um projeto de *software*, também estão atrelados a ele uma série de riscos, podendo possuir tanto um efeito positivo, quanto negativo. Então, logo após o levantamento de requisitos e definição do escopo da aplicação, a equipe realizou um estudo, para identificar os principais riscos a serem enfrentados durante a construção da mesma. Em conjunto com os possíveis obstáculos, também foram adicionados um plano de ação, seu nível de probabilidade e impacto para cada situação de risco mapeada para a construção do aplicativo Easy Wash.

QUADRO 2 - PLANO DE RISCOS

Número	Descrição	Solução para o Risco	Probabilidade do Risco	Impacto
1	Falta de tempo hábil para o desenvolvimento do projeto	Priorização das atividades de acordo com seu nível de importância para o projeto.	Alta	Grave
2	Desinteresse no desenvolvimento do projeto por parte dos integrantes	Repasse de atividades referentes a outro integrante.	Alta	Grave
3	Mudanças nos requisitos do projeto	Avaliação de todos os integrantes para debater viabilidade de mudança e impacto nas atividades finalizadas e a serem realizadas dentro do projeto. .	Alta	Grave
4	Falta de conhecimento da tecnologia utilizada	Reunião com outros membros da equipe para discussão do tema.	Alta	Grave
5	Irregularidade de comunicação entre os integrantes	Reuniões presenciais realizadas semanalmente para acompanhamento do progresso das atividades entre todos os integrantes.	Alta	Moderada

6	Esforço estimado de maneira incorreta para cada atividade	Replanejamento de esforço para a atividade com a priorização de acordo com sua importância para o projeto	Alta	Grave
---	---	---	------	-------

FONTE: Os Autores (2020)

3.9 RESPONSABILIDADES

Após levantamento de requisitos e definição do *backlog* do produto. A equipe em reunião definiu um escopo macro de todas as atividades, atribuindo as posições onde cada membro atuaria, conforme mostrado no QUADRO 3.

QUADRO 3 - PLANO DE RESPONSABILIDADES

Número	Integrante	Responsabilidade
1	Allan Lerner	Desenvolvimento Back-end Móvel
		Documentação do projeto
		Testes de Software
		Prototipação das telas
2	Claudinei Varela	Desenvolvimento Back-end Móvel
		Documentação do projeto
		Testes de Software
		Diagramas UML
3	Diego Lemos Padilha	Desenvolvimento Back-end Web
		Desenvolvimento Front-end Web
		Documentação do projeto
		Testes de Software
		Modelagem do banco de dados
4	Jean de Jesus Freire	Prototipação das telas
		Diagramas UML
		Documentação do projeto
		Desenvolvimento Back-end Móvel
		Desenvolvimento Front-End Móvel

FONTE: Os Autores (2020)

Um ponto a ser enfatizado nesse plano de responsabilidades, como pode ser verificado no QUADRO 2, refere-se à participação de cada membro praticamente em todas as fases do projeto. Esse fato ocorreu, primeiramente pelo seu próprio escopo,

o qual era relativamente grande para que um membro atuasse de forma individual. Além disso, outro motivo que foi definido pela própria equipe foi oportunizar a difusão do conhecimento de cada etapa entre todos os integrantes do projeto. Isso trouxe várias interpretações para cada etapa que contribuíram posteriormente e muito na elevação de qualidade das mesmas.

3.10 PLANO DE ATIVIDADES

O plano de atividades para o desenvolvimento do aplicativo Easy Wash, foi criado para mapeamento de todas as tarefas que deveriam ser entregues, estimar suas durações e encaixá-las dentro das *sprints*, seguindo uma sequência organizada de execução para que a entrega do projeto fosse concluída com êxito.

A equipe se organizou através de um cronograma ágil de atividades, o qual foi mais empregado como referência de planejamento. Nesta planilha cada *sprint*, era composta por nome, descrição, data de execução. Além disso, contemplava também suas sub tarefas e o *increment product*, que representa os artefatos gerados ao final de cada ciclo, conforme ilustram as figuras a seguir.

FIGURA 15 - CRONOGRAMA DE ATIVIDADES SPRINT 0 A SPRINT 2

Sprint #0 - 1 semana - 05/08 à 12/08 Meta da Sprint Definição de tema e criação de um cronograma ágil de todo o projeto		Increment product
Product backlog 0.1 Criação de cronograma com todas as tarefas a serem desenvolvidas durante o TCC parte 1 0.2 Pesquisa e levantamento de idéias para definição de tema 0.3 Pesquisa e definição de tecnologias para aplicação do tema		Cronograma das atividades de todo semestre + Tema definido + Tecnologias principais da aplicação definidas
Sprint #1 - 2 semanas 13/08 à 27/08 Meta da Sprint Gerar uma lista de requisitos da aplicação e pesquisa bibliográfica de softwares semelhantes.		Increment product
Product backlog 1.1 Levantamento inicial de requisitos que a aplicação deverá possuir. 1.2 Pesquisa bibliográfica de softwares semelhantes no mercado 1.3 Criação de um gráfico de GANNT 1.4 Estudo da arquitetura da aplicação 1.5 1.6		Escopo do projeto com as funcionalidades que a aplicação deve possuir + Levantamento de todos os softwares disponíveis no mercado e suas principais características + funcionalidades + Gráfico Gannt
Sprint #2 - 2 semanas 28/08 à 10/09 Meta da Sprint Entrega da Especificação e Diagramas de Casos de uso.		Increment product
Product backlog 2.1 Especificação de Caso de Uso 2.2 Prototipação das telas 2.3 Refinamento e correções dos requisitos		Entrega da Especificação de Caso de Uso com todos os protótipos das telas da aplicação.

FONTE: Os Autores (2020)

FIGURA 16 - CRONOGRAMA DE ATIVIDADES SPRINT 3 A SPRINT 8

Sprint #3 - 2 semanas 11/09 à 24/09		
Meta da Sprint Entrega da do Capítulo 1 e definição de histórias p/ pock		
Product backlog		Increment product
3.1	Desenvolvimento do item 1.1.Problema da Introdução do TCC	Entrega do Capítulo 1 e definição de histórias para poc
3.2	Desenvolvimento do item 1.2.Objetivos Gerais da Introdução do TCC	
3.3.	Desenvolvimento do item 1.3.Objetivos Especificos da Introdução do TCC	
3.4	Desenvolvimento do item 1.4. Justificativa	
3.5	Definição das histórias que serão usadas na poc	
Sprint #4 - 2 semanas 25/09 à 08/10		
Meta da Sprint Entrega do Diagrama de Classes e Diagramas de Sequência		
Product backlog		Increment product
4.1	Desenvolvimento Diagramas de classe	Entrega Diagramas de classe e Diagramas de Sequência da aplicação + Início do desenvolvimento das telas para POC Front-end
4.2	Desenvolvimento Diagramas de sequência	
4.3	Criação de ambiente de desenvolvimento (repositório de versionamento)	
4.4	Início da criação das telas p/ POC Front-end	
Sprint #5 - 2 semanas 09/10 à 22/10		
Meta da Sprint Entrega do Capítulo 2 do TCC - Fundamentação Teórica		
Product backlog		Increment product
5.1	Parte 1 do Capítulo 2 (Fundamentos do negócio e da aplicação	Entrega do Capítulo 2, contendo a fundamentação do negócio, da aplicação e das tecnologias utilizadas + Fundamentação de comparação com softwares já existentes
5.2	Parte 2 do Capítulo 2 (Fundamentos da tecnologia envolvida	
5.3	Parte 2 do Capítulo 2 (Comparação c/ softwares já existentes - Vantagens e desvantagens	
5.4	Desenvolvimento da camada front-end	

FONTE: Os Autores (2020)

FIGURA 17 - CRONOGRAMA DE ATIVIDADES SPRINT 6 A SPRINT 8

Sprint #6 - 2 semanas 23/10 à 05/11		
Meta da Sprint Criação de banco de dados e script inicial + início do desenvolvimento back-end		
Product backlog		Increment product
6.1	Criação do diagrama físico do banco de dados	Criação do banco de dados e script inicial + Início do desenvolvimento Back-End
6.2	Geração de Script SQL com carga inicial	
6.3	Testes do desenvolvimento Front-end	
6.4	Início desenvolvimento Back-end	
Sprint #7 - 2 semanas 06/11 à 19/11		
Meta da Sprint Entrega do Capítulo 3 e 4 do TCC - Metodologias e métodos/Considerações Finais		
Product backlog		Increment product
7.1	Desenvolvimento do item 1 do capítulo 3 - Descrição da metodologia utilizada durante o trabalho e suas adaptações	Entrega do Capítulo 3, contendo Parte 1 (Descrição das metodologias utilizadas durante o trabalho) + Entrega do Capítulo 4 + Inserção e fundamentação de todos os diagramas + Testes iniciais na camada Back-end
7.2	Desenvolvimento do Item 2 do Cap - Anexação e fundamentação de todos os diagramas criados	
7.3	Desenvolvimento do Capítulo 4 - Considerações Finais	
7.4	Geração de documentação com formatação final p/ entrega	
7.5	Testes da camada Back-end da aplicação	
7.6	Criação de slides para apresentação	
Sprint #8 - 2 semanas 20/11 à 26/11		
Meta da Sprint Revisão Final da parte documental e da POC da aplicação		
Product backlog		Increment product
8.1	Revisão do capítulo 1	Revisão documental
8.2	Revisão do capítulo 2	
8.3	Revisão do capítulo 3	
8.4	Revisão do capítulo 4	

FONTE: Os Autores (2020)

FIGURA 18 - CRONOGRAMA DE ATIVIDADES SPRINT 9 A SPRINT 10
TCC2 - início 2020

Sprint #9 - 2 semanas 02/03 à 15/03																																			
Meta da Sprint	Preparar o ambiente de implementação e correção documentação																																		
	<table border="1"> <thead> <tr> <th colspan="2">Product backlog</th> </tr> </thead> <tbody> <tr><td>9.1</td><td>Refatoração de banco de dados</td></tr> <tr><td>9.2</td><td>Criação de repositório para API Backend</td></tr> <tr><td>9.3</td><td>Criação de repositório github para aplicação react.js</td></tr> <tr><td>9.4</td><td>Criação dos quadros para cada frente do projeto no trello</td></tr> <tr><td>9.5</td><td>Criação de script inicial da aplicação</td></tr> <tr><td>9.6</td><td>Criação de estrutura básica do projeto react.js</td></tr> <tr><td>9.7</td><td>Criação de rotas básicas da aplicação</td></tr> <tr><td>9.8</td><td>Criação de script SQL para estados e cidades</td></tr> <tr><td>9.10</td><td>Organização de todas sugestões dos professores para melhorias na documentação</td></tr> </tbody> </table>	Product backlog		9.1	Refatoração de banco de dados	9.2	Criação de repositório para API Backend	9.3	Criação de repositório github para aplicação react.js	9.4	Criação dos quadros para cada frente do projeto no trello	9.5	Criação de script inicial da aplicação	9.6	Criação de estrutura básica do projeto react.js	9.7	Criação de rotas básicas da aplicação	9.8	Criação de script SQL para estados e cidades	9.10	Organização de todas sugestões dos professores para melhorias na documentação	<table border="1"> <thead> <tr> <th>Increment product</th> </tr> </thead> <tbody> <tr> <td>Repositórios, quadros, script do banco criados e início do ambiente backend</td> </tr> </tbody> </table>	Increment product	Repositórios, quadros, script do banco criados e início do ambiente backend											
Product backlog																																			
9.1	Refatoração de banco de dados																																		
9.2	Criação de repositório para API Backend																																		
9.3	Criação de repositório github para aplicação react.js																																		
9.4	Criação dos quadros para cada frente do projeto no trello																																		
9.5	Criação de script inicial da aplicação																																		
9.6	Criação de estrutura básica do projeto react.js																																		
9.7	Criação de rotas básicas da aplicação																																		
9.8	Criação de script SQL para estados e cidades																																		
9.10	Organização de todas sugestões dos professores para melhorias na documentação																																		
Increment product																																			
Repositórios, quadros, script do banco criados e início do ambiente backend																																			
	<table border="1"> <thead> <tr> <th colspan="2"> Sprint #10 - 2 semanas 16/03 à 29/03 </th> </tr> </thead> <tbody> <tr> <td>Meta da Sprint</td> <td>Iniciar implementação backend</td> <td></td> </tr> <tr> <td></td> <td> <table border="1"> <thead> <tr> <th colspan="2">Product backlog</th> </tr> </thead> <tbody> <tr><td>10.1</td><td>Criação de endpoint consulta de estado e cidades</td></tr> <tr><td>10.2</td><td>Criação de consulta para tipos de veículos</td></tr> <tr><td>10.3</td><td>Criação de consulta para formas de retirada de veículo</td></tr> <tr><td>10.4</td><td>Criação de CRUD moderadores</td></tr> <tr><td>10.5</td><td>Criação CRUD de clientes</td></tr> <tr><td>10.6</td><td>Criação CRUD para clientes/{id}/endereco</td></tr> <tr><td>10.7</td><td>Criação CRUD clientes/{id}/cartoes-credito</td></tr> <tr><td>10.8</td><td>Criação CRUD clientes/{id}/automoveis</td></tr> <tr><td>10.9</td><td>Criação CRUD /prestadores</td></tr> <tr><td>10.10</td><td>Criação CRUD /prestadores{id}/servicos</td></tr> <tr><td>10.11</td><td>Correções na documentação sugeridas pelos professores p1 (intro 1.0, obj 1.1, prob 1.2, just 1.3, 2, 2.1.1, ...)</td></tr> </tbody> </table> </td> <td> <table border="1"> <thead> <tr> <th>Increment product</th> </tr> </thead> <tbody> <tr> <td>Criação endpoint e CRUDS.</td> </tr> </tbody> </table> </td> </tr> </tbody> </table>	Sprint #10 - 2 semanas 16/03 à 29/03		Meta da Sprint	Iniciar implementação backend			<table border="1"> <thead> <tr> <th colspan="2">Product backlog</th> </tr> </thead> <tbody> <tr><td>10.1</td><td>Criação de endpoint consulta de estado e cidades</td></tr> <tr><td>10.2</td><td>Criação de consulta para tipos de veículos</td></tr> <tr><td>10.3</td><td>Criação de consulta para formas de retirada de veículo</td></tr> <tr><td>10.4</td><td>Criação de CRUD moderadores</td></tr> <tr><td>10.5</td><td>Criação CRUD de clientes</td></tr> <tr><td>10.6</td><td>Criação CRUD para clientes/{id}/endereco</td></tr> <tr><td>10.7</td><td>Criação CRUD clientes/{id}/cartoes-credito</td></tr> <tr><td>10.8</td><td>Criação CRUD clientes/{id}/automoveis</td></tr> <tr><td>10.9</td><td>Criação CRUD /prestadores</td></tr> <tr><td>10.10</td><td>Criação CRUD /prestadores{id}/servicos</td></tr> <tr><td>10.11</td><td>Correções na documentação sugeridas pelos professores p1 (intro 1.0, obj 1.1, prob 1.2, just 1.3, 2, 2.1.1, ...)</td></tr> </tbody> </table>	Product backlog		10.1	Criação de endpoint consulta de estado e cidades	10.2	Criação de consulta para tipos de veículos	10.3	Criação de consulta para formas de retirada de veículo	10.4	Criação de CRUD moderadores	10.5	Criação CRUD de clientes	10.6	Criação CRUD para clientes/{id}/endereco	10.7	Criação CRUD clientes/{id}/cartoes-credito	10.8	Criação CRUD clientes/{id}/automoveis	10.9	Criação CRUD /prestadores	10.10	Criação CRUD /prestadores{id}/servicos	10.11	Correções na documentação sugeridas pelos professores p1 (intro 1.0, obj 1.1, prob 1.2, just 1.3, 2, 2.1.1, ...)	<table border="1"> <thead> <tr> <th>Increment product</th> </tr> </thead> <tbody> <tr> <td>Criação endpoint e CRUDS.</td> </tr> </tbody> </table>	Increment product	Criação endpoint e CRUDS.
Sprint #10 - 2 semanas 16/03 à 29/03																																			
Meta da Sprint	Iniciar implementação backend																																		
	<table border="1"> <thead> <tr> <th colspan="2">Product backlog</th> </tr> </thead> <tbody> <tr><td>10.1</td><td>Criação de endpoint consulta de estado e cidades</td></tr> <tr><td>10.2</td><td>Criação de consulta para tipos de veículos</td></tr> <tr><td>10.3</td><td>Criação de consulta para formas de retirada de veículo</td></tr> <tr><td>10.4</td><td>Criação de CRUD moderadores</td></tr> <tr><td>10.5</td><td>Criação CRUD de clientes</td></tr> <tr><td>10.6</td><td>Criação CRUD para clientes/{id}/endereco</td></tr> <tr><td>10.7</td><td>Criação CRUD clientes/{id}/cartoes-credito</td></tr> <tr><td>10.8</td><td>Criação CRUD clientes/{id}/automoveis</td></tr> <tr><td>10.9</td><td>Criação CRUD /prestadores</td></tr> <tr><td>10.10</td><td>Criação CRUD /prestadores{id}/servicos</td></tr> <tr><td>10.11</td><td>Correções na documentação sugeridas pelos professores p1 (intro 1.0, obj 1.1, prob 1.2, just 1.3, 2, 2.1.1, ...)</td></tr> </tbody> </table>	Product backlog		10.1	Criação de endpoint consulta de estado e cidades	10.2	Criação de consulta para tipos de veículos	10.3	Criação de consulta para formas de retirada de veículo	10.4	Criação de CRUD moderadores	10.5	Criação CRUD de clientes	10.6	Criação CRUD para clientes/{id}/endereco	10.7	Criação CRUD clientes/{id}/cartoes-credito	10.8	Criação CRUD clientes/{id}/automoveis	10.9	Criação CRUD /prestadores	10.10	Criação CRUD /prestadores{id}/servicos	10.11	Correções na documentação sugeridas pelos professores p1 (intro 1.0, obj 1.1, prob 1.2, just 1.3, 2, 2.1.1, ...)	<table border="1"> <thead> <tr> <th>Increment product</th> </tr> </thead> <tbody> <tr> <td>Criação endpoint e CRUDS.</td> </tr> </tbody> </table>	Increment product	Criação endpoint e CRUDS.							
Product backlog																																			
10.1	Criação de endpoint consulta de estado e cidades																																		
10.2	Criação de consulta para tipos de veículos																																		
10.3	Criação de consulta para formas de retirada de veículo																																		
10.4	Criação de CRUD moderadores																																		
10.5	Criação CRUD de clientes																																		
10.6	Criação CRUD para clientes/{id}/endereco																																		
10.7	Criação CRUD clientes/{id}/cartoes-credito																																		
10.8	Criação CRUD clientes/{id}/automoveis																																		
10.9	Criação CRUD /prestadores																																		
10.10	Criação CRUD /prestadores{id}/servicos																																		
10.11	Correções na documentação sugeridas pelos professores p1 (intro 1.0, obj 1.1, prob 1.2, just 1.3, 2, 2.1.1, ...)																																		
Increment product																																			
Criação endpoint e CRUDS.																																			

FONTE: Os Autores (2020)

FIGURA 19 - CRONOGRAMA DE ATIVIDADES SPRINT 10 A SPRINT 11

Sprint #11 - 2 semanas 30/03 à 12/04		
Meta da Sprint	Seqüência da criação dos CRUDS e endpoints	Increment product
Product backlog		
11.1	Criação consulta formas-pagamento	CRUDS, endpoints, configuração Swagger
11.2	Criação endpoint GET/PUT/POST clientes/{id}/notificacoes	
11.3	Criação GET/PUT/POST prestadores/{id}/notificacoes	
11.4	Criação endpoint de login para moderadores, clientes e prestadores	
11.5	Criação endpoints para recuperação de senha moderadores, clientes e prestadores	
11.6	Criação e configuração de conta para envio de e-mails	
11.7	Criação CRUD /prestadores/{id}/formas-pagamento	
11.8	Configuração Swagger na aplicação back-end	
11.9	Criação de endpoint para adição de imagens de foto de perfil a AWS (Moderadores, clientes, prestadores)	
11.10	Correções na documentação sugeridas pelos professores p2 (materias e métodos 3, 3.1, requisitos e diagramas UC)	
Sprint #12 - 2 semanas - 13/04 à 26/04		
Meta da Sprint	Seqüência endpoint e paginas restantes	Increment product
Product backlog		
12.1	Criação de endpoint de GET/POST/PUT atendimentos	Deploy da aplicação servidor Heroku e paginas de consultas
12.2	Criação de endpoint de GET/POST/PUT notificações	
12.3	Deploy aplicação servidor HEROKU	
12.4	WEB - Criação de deploy automatizado para aplicação	
12.5	WEB - Criação página de login	
12.6	WEB - Criação de página para reenvio de senha	
12.7	WEB - Criação CRUD moderadores	
12.8	WEB - Criação página de consultas de clientes (Ler documentação TCC)	
12.9	WEB - Criação página de consultas de prestadores (Ler documentação TCC)	
12.10	WEB - Criar página para inserção/alteração de foto de perfil	
12.11	Revisar cronograma	

FONTE: Os Autores (2020)

FIGURA 20 - CRONOGRAMA DE ATIVIDADES SPRINT 13 A SPRINT 14

Sprint #13 - 2 semanas - 27/04 à 09/05		
Meta da Sprint	Criar as funcionalidades de listagem/exclusão e inicio da estrutura mobile	Increment product
Product backlog		
13.1	WEB - Criação funcionalidade listagem e exclusão de formas de pagamento na tela de listagem de prestadores	Funcionalidade de listagem e exclusão de formas de pagamento, serviços, aprovação de solicitação, filtro de notificações
13.2	WEB - Criação funcionalidade listagem e exclusão de serviços na tela de listagem de prestadores	
13.3	WEB - Criado página para listagem de prestadores pendentes e funcionalidade de aprovação ou recusa de solicitação	
13.4	WEB - Criação página listagem de notificações por filtro	
13.5	WEB - Cadastro de notificações	
13.6	MOBILE - Criada estrutura básica projeto mobile (repositório do projeto e configuração de bibliotecas básicas)	
13.7	documentação cap 4, print das telas sistema web	
Sprint #14 - 2 semanas - 10/05 à 23/05		
Meta da Sprint	Inicio das atividades mobile	Increment product
Product backlog		
14.1	Criação de navegação interna perfil cliente	pagina de perfil do cliente e possibilidade de cadastrar cartão usado
14.2	Integração de página de login com API	
14.3	Criação de pagina profile (cliente)	
14.4	Criação de página Meus Cartões	
14.5	Criação de integração com consulta de Meus Cartões a API	
14.6	Criação Modal genérico aplicação	
14.7	Criação Splash e ícones do aplicativo	
14.8	Continuação do cap4, prints das telas mobile mobile	
14.9	documentação cap 5, iniciar considerações finais e recomendações para trabalhos futuros	

FONTE: Os Autores (2020)

FIGURA 21 - CRONOGRAMA DE ATIVIDADES SPRINT 15 A SPRINT 16



FONTE: Os Autores (2020)

FIGURA 22 - CRONOGRAMA DE ATIVIDADES SPRINT 17



FONTE: Os Autores (2020)

3.10.1 Planejamento

O planejamento das sprints do projeto, iniciou-se após a realização do primeiro encontro com o professor orientador. Nesse momento, a equipe foi contextualizada com uma visão geral e objetivos a serem concluídos nesta etapa inicial da disciplina.

Os membros do grupo possuíam várias ideias para projeto, porém sem muito aprofundamento de suas funcionalidades ou ainda se a mesma era de fato uma opção viável. Essas incertezas também se aplicam com relação as tecnologias que seriam utilizadas.

Assim sendo, foi alinhado entre membros e professor orientador que a entrega

dessa primeira *sprint* seria construir um debate para alinhamento de tema e tecnologia e ainda criar um plano de atividades para ser seguido durante toda a etapa. Levando em consideração que a partir desse momento já era conhecido todo o escopo obrigatório a ser entregue, principalmente para esta primeira fase da disciplina, a qual possui enfoque primordial na documentação do sistema.

3.10.2 Sprint 1

A segunda *sprint* aqui foi considerada como sendo a primeira, devido que a partir desse momento já se tem um tema inicial definido. Após uma série de pesquisas, conversas, dúvidas e incertezas chegou-se a uma escolha: um aplicativo para lavagem de automóveis sob demanda.

O próximo passo era levantar todos os requisitos que a aplicação deveria possuir. Para isso, foi pesquisado no mercado se havia aplicativos para o mesmo segmento ou ainda por outros que pudessem servir como inspiração. Aqui é citado de modo especial Uber e Ifood. Já para aplicativos do mesmo nicho de serviços, foi encontrado o Lavô e o App Wash no Brasil. A equipe também pesquisou por soluções estrangeiras e descobriu o Washos e o Mobile Wash.

Com relação as tecnologias, a camada *back-end* da aplicação foi um pouco mais fácil de definir e foi optado por utilizar Java, através da utilização do framework Spring Boot. O consenso foi atingido, devido todos os membros já terem algum contato com a linguagem Java, logo a aprendizagem do framework teria sua curva de aprendizagem bastante reduzida.

Já para a camada *front-end*, nesse momento a equipe estava mais propensa a utilização do *framework* Angular, para o módulo web do sistema. Com relação a camada móvel o grupo também estava propenso a utilização do Angular, por meio de outro *framework* que é o NativeScript, o qual permite a construção de um único código que é compilado para as linguagens nativas Android e Swift.

Nessa *sprint* também foi criado um Gráfico de Gantt para todas as atividades levantadas na etapa anterior, como sugestão do professor orientador, para a equipe ter a possibilidade de estimar o tempo das atividades e seus respectivos responsáveis.

3.10.3 Sprint 2

Na segunda *sprint*, após o grupo já possuir uma lista de requisitos iniciais para a aplicação, estes foram muito mais debatidos e aprofundados e de fato definidos. Como resultado, nesta etapa foi gerado o documento com o escopo do projeto (APÊNDICE A, B, C).

Posteriormente, a equipe desenvolveu os Diagramas de Casos de Uso (APÊNDICE D, E, F) para o aplicativo, levando em consideração os seguintes atores fixados para o sistema: Cliente, prestador e moderador.

Em seguida, após a conclusão destes dois artefatos a equipe começou a desenhar os protótipos das aplicações, tanto para o sistema móvel quanto o web. Para isso, foi utilizado o Balsamiq Wireframes. Neste momento, estas atividades foram subdivididas em 3 partes iguais, com todos os membros trabalhando em conjunto nesta tarefa.

Além dos itens gerados anteriormente nessa *Sprint*, no cronograma do projeto, também estavam estimadas as entregas referentes construção de toda especificação dos Casos de Uso. No entanto, a quantidade de telas a serem prototipadas e validadas foram um impedimento para a conclusão deste item, o qual foi realocado para a próximo ciclo.

3.10.4 Sprint 3

Nesta etapa a equipe estava composta por 3 membros. Como alguns dos artefatos definidos como saída para a *sprint* anterior não foram concluídas dentro do prazo foi realizado a disposição da equipe da seguinte maneira:

- Allan: Nos três primeiros dias da *sprint*, finalizou a prototipação e após isso ficou responsável pela realização completa do Capítulo 1 desta documentação;
- Diego e Jean: Trabalharam alocados na conclusão da prototipação, construção da especificação de Casos de uso e ajustes na documentação de requisitos, conforme necessidade.

Como artefatos de saída dessa etapa, foram gerados uma primeira versão do

Capítulo 1 desta documentação, que posteriormente foi sendo corrigido e aprimorado ao longo do projeto e também a conclusão da Especificação de Casos de Uso, com a prototipação das telas toda finalizada (APÊNDICE G, H, I).

3.10.5 Sprint 4

A quarta *sprint* foi utilizada para realização de ajustes e aprimoramento apontados pelo professor orientador referentes ao Capítulo 1 desta documentação. Esta atividade ficou sob responsabilidade do integrante Allan.

Em paralelo, os outros dois membros iniciaram a construção dos Diagramas de classe (APÊNDICE J), para permitir uma visualização macro de estrutura de objetos do sistema e também seus relacionamentos.

Com a conclusão deste diagrama, a equipe estava habilitada para criar um esboço do modelo lógico do banco de dados (APÊNDICE O). Apesar desta atividade não ser crítica, foi decidido realizá-la nesta etapa para aproveitar todo o conhecimento adquirido com a construção do primeiro diagrama e até mesmo a especificação das telas.

Na sequência deu-se início a construção dos Diagramas de Sequência (APÊNDICE K, L, M). O objetivo era ter uma percepção mais clara de quais seriam os fluxos entre as classes da aplicação de acordo, com cada funcionalidade do sistema.

Todas as atividades desta *sprint* foram finalizadas com sucesso nesta etapa, incluindo aqui também a criação do Diagrama de pacotes (APÊNDICE N).

3.10.6 Sprint 5

Na quinta *sprint*, houve a entrada de mais um integrante na equipe. Neste momento, junto ao professor orientador, foi verificado que o prazo restante para a conclusão do projeto já estava em menos de 50% e ainda o conteúdo referente a esta documentação estava em um estado bem inicial, somente com o primeiro capítulo concluído.

Logo, optou-se por atribuir todas as refatorações e criação dos diagramas restantes para um integrante somente. Com o aprofundamento na construção dos diagramas de sequência foi necessário a atualização de vários diagramas criados anteriormente, como por exemplo o de classes, de casos de uso e até mesmo o

modelo físico do banco de dados.

Desta forma, os outros 3 integrantes da equipe começaram a pesquisar o conteúdo de referência para elaboração do Capítulo 2. Ficando a subdivisão desta atividade da seguinte maneira:

- Allan: Fundamentação teórica do negócio;
- Claudinei: Fundamentação teórica das tecnologias utilizadas referentes aos processos e pesquisa por aplicativos/ferramentas próximas do mesmo negócio ou ainda que tenham alguma possível semelhança com o Easy Wash mesmo sendo de outro serviço;
- Diego: Fundamentação Teórica das tecnologias utilizadas.

Esta etapa coincidiu com a semana de provas e entregas de trabalhos de outras disciplinas da faculdade e não foi possível gerar os artefatos de saída referente ao Capítulo 2 que foram anteriormente previstos. Neste momento, seu nível de conclusão atingido foi de aproximadamente 60%.

No entanto, todos os diagramas solicitados pelo professor nesta etapa e que não eram conhecidos, foram concluídos, como por exemplo os Diagrama de Implementação de Pacotes de (APÊNDICE N).

3.10.7 Sprint 6

Nesta *sprint*, após entrega dos diagramas da etapa anterior, o professor orientador realizou uma série de apontamentos e ajustes para melhorias. Conseqüentemente, um integrante da equipe já foi deslocado para atuar em todas essas correções.

Por outro lado, outros dois continuaram com o desenvolvimento do segundo capítulo, tanto na parte de pesquisa, quanto na elaboração do próprio documento.

Como o prazo final de entrega da primeira versão estava se aproximando outro membro passou a trabalhar diretamente neste atual capítulo.

Os artefatos de saída nessa *sprint*, foram a conclusão do Capítulo 2, uma primeira versão do Capítulo 3 e a conclusão de todos os diagramas deste projeto.

3.10.8 Sprint 7

A sétima *sprint* conforme descrito no cronograma de atividades, seria a última utilizada para produção de conteúdo vinculada a esta documentação.

Nesta etapa, após *feedback* do professor orientador, foram realizadas todas as correções mapeadas pelo mesmo, referente aos Capítulos 2 e 3.

Além disso, um membro atuou na elaboração do capítulo referente as considerações finais, que na primeira parte da disciplina seria o quarto e último. Este capítulo traz o compartilhamento da equipe, relacionada a toda experiência absorvida ao longo deste projeto. Todos tiveram a oportunidade de contribuir na elaboração deste capítulo.

Para atender o objetivo final, neste momento também foi unificado todo o conteúdo produzido para realização da primeira etapa do projeto, com geração de toda formatação requerida e criação de outros itens obrigatórios como: Sumário, Lista de ilustrações, lista de tabelas e Referências.

3.10.9 Sprint 8

Esta foi a *sprint* de conclusão do primeiro semestre e teve duração de 1 semana. O foco nesse momento era atuar na construção da POC (Prova de Conceito), que tem por objetivo principal nesse processo, validar a ideia do projeto, principalmente relacionada a utilização de todas as tecnologias envolvidas.

Como funcionalidade a ser ofertada nessa etapa, a equipe optou por desenvolver a operação de Gerenciar Cartões de Crédito, a qual permite ao cliente cadastrar e editar seus cartões de crédito deixando-os salvos no aplicativo para momentos que o mesmo desejar realizar transações de pagamento no aplicativo.

Esta funcionalidade como um todo foi subdividida nas seguintes atividades:

- Desenvolvimento *front-end*;
- Desenvolvimento *back-end*;
- Testes.

No meio tempo entre a criação desta prova de conceito, a equipe também teve o retorno do professor orientador, com o levantamento de todas as correções que

seriam necessárias para entrega final.

3.10.10 Sprint 9

Esta foi a primeira *sprint* referente ao segundo semestre, retomando as atividades. Nesta etapa o foco principal foi configurar o ambiente de programação, e também a criação das estruturas básicas da aplicação, tais como:

- Quadros no Trello, organização e planejamento;
- Criação de estrutura básica do projeto react.js;
- Criação de repositórios Github para API BACKEND e aplicação react.js;
- Criação de scripts iniciais SQL.

A equipe levantou todas as correções sugeridas de cada professor e organizou em cartões no Trello.

Já com relação as atividades de desenvolvimento para esta aplicação, vale ressaltar que os desenvolvedores durante as férias, realizaram treinamentos voltados as tecnologias escolhidas para este projeto, a fim de aprofundarem seus conhecimentos para atingirem os objetivos deste projeto.

3.10.11 Sprint 10

Nesta sprint, após a definição de estrutura e padrão a ser seguidos, principalmente com relação ao sistema back-end, foram desenvolvidas as seguintes atividades:

- Criação de *endpoint* consultas de estado, cidades, tipos e veículo e formas de retiradas de veículos;
- Criação de CRUD's /moderadores, /clientes, /clientes/id/endereço, /clientes /id/cartoes-credito, /clientes/idprestadores, /prestadores e /prestadores/id/serviço;
- Início das correções da documentação levantadas pelos professores.

O grande desafio desta etapa, relacionado as atividades de desenvolvimento foi equalizar o conhecimento dos padrões entre todos os membros da equipe e até mesmo a sincronia e comunicação entre as atividades. Nesta semana, a equipe teve alguns problemas relacionados a conflito e sobrescrita de código, pequenas falhas na utilização do GitHub entre os integrantes da equipe, mas logo foi notado e resolvido.

3.10.12 Sprint 11

Nesta fase foram finalizados todos os endpoints relacionados aos perfis de clientes e prestadores.

Além disso, relacionado a camada back-end da aplicação, a equipe passou a atuar na construção de funcionalidades mais avançadas, como envio de e-mail e salvamento das imagens dos perfis dos usuários. Como solução decidimos por utilizar o Gmail e a nuvem da AWS (Amazon Web Services) para estes fins.

Esta etapa, foi um grande desafio para os desenvolvedores. Comunicar nossa aplicação com outros serviços externos. Foi um grande estímulo, primeiramente por analisar as principais possibilidades e já implementar pequenos exemplos, para avaliar como essas integrações se comportariam.

Em paralelo, foi feito o término das correções da documentação, restando apenas alguns diagramas para serem corrigidos posteriormente, mais próximo ao final do projeto, após possíveis alterações no banco de dados e nas classes.

Esta sprint a equipe de forma geral, conseguiu dar uma alta vazão nas atividades, sendo umas das sprints mais produtivas do projeto.

3.10.13 Sprint 12

Na *sprint* 12, o restante dos *endpoints* foram criados, estes vinculados as funcionalidades de atendimentos e notificações. Nesta fase, após grande evolução do sistema back-end a equipe foi dividida e iniciou-se a construção das páginas do sistema *WEB*. Algumas atividades que valem ressaltar foram:

- Deploy da aplicação *back-end* servidor Heroku;
- Criação página de login, reenvio de senha, consulta de clientes e prestadores;

- Criação da página de inserção/alteração de foto de perfil e página CRUD de moderadores;
- Revisão do cronograma.

O cronograma foi revisto e a equipe planejou o que ainda precisava ser construído. Algumas questões foram levantadas, como possíveis simplificações e até corte de funcionalidades, devido ao projeto se demonstrar muito robusto. As dificuldades com as novas tecnologias apareceram e atrasaram o desenvolvimento.

Outro ponto importante nesta etapa, foi a realização de atividades que inicialmente não estavam previstas, porém eram muito importantes para evolução do escopo. Exigiu uma grande adaptabilidade da equipe e extrema organização de todos os membros para dar vazão a demanda.

3.10.14 Sprint 13

Nesta fase a equipe deu sequência ao desenvolvimento do sistema *web*. Além disso, após avaliação do cronograma foi decidido pelos membros dar início a construção do aplicativo para dispositivos móveis. A seguir um resumo das atividades:

- Criação das funcionalidades listagem e exclusão de formas de pagamento e serviços, referente ao perfil prestador;
- Criação página para listagem de prestadores pendentes e aprovar/recusar solicitação;
- Início do projeto mobile como repositório e configuração de bibliotecas básicas.

O início das atividades do aplicativo móvel estava atrasado, no entanto o novo cronograma foi ajustado e contemplava todas as estas alterações. Houve uma grande evolução da aplicação *web* nesta fase. Logo, a equipe optou por realizar a captura das imagens do sistema para em seguida começar a redação do Capítulo 4.

3.10.15 Sprint 14

Após a criação da estrutura básica do projeto móvel. Iniciou-se a construção de algumas funcionalidades, entre as quais:

- Criação de navegação interna para o perfil cliente;
- Criação e integração de página de login, esqueci minha senha e consulta de meus cartões com a API *back-end*;
- Criação de modal genérico da aplicação;
- Criação de ícones e logo para o aplicativo.

Como o escopo estava levemente atrasado, algumas capturas das telas do aplicativo que estavam prontas foram sendo adicionadas a documentação e explanadas no Capítulo 4. Acerca da documentação nesta fase, iniciou-se neste o levantamento das informações requeridas para a redação do Capítulo 5 bem como as considerações finais.

Com o início da construção do aplicativo o principal desafio, foi primeiramente escolher e posteriormente utilizar bibliotecas externas para agilizar e ajudar a implementação de algumas funcionalidades do sistema.

3.10.16 Sprint 15

Com base na *sprint* anterior, houve um grande aprendizado no que se refere as tecnologias utilizadas no aplicativo o que proporcionou uma maior evolução da equipe. Entre as principais atividades desenvolvidas nesta etapa estão:

- Criação de navegação interna para o perfil prestador;
- Listagem e exclusão de automóveis, referentes ao perfil cliente;
- Listagem e exclusão de formas de pagamento e serviços, referentes ao perfil prestador;
- Logout da aplicação;
- Página para upload e alteração de foto de perfil;
- Criação de formulário para cadastro e edição de novos clientes.

Os grandes desafios relacionados as atividades de desenvolvimento desta etapa para a equipe, foram principalmente relacionados ao aprendizado para acessar alguns recursos do dispositivo do usuário, como câmera e galeria de imagens para upload de arquivos.

Outra funcionalidade que foi extremamente crítica para a equipe, está relacionada ao cadastro de cliente. A aplicação back-end foi construída exigindo uma grande quantidade de informações. Logo, o aplicativo requeria um formulário passo a passo, o qual foi o grande desafio. Identificar uma biblioteca externa para resolver este problema ou criar um componente todo do zero.

Quanto as atividades de documentação, com a estrutura back-end bem madura nesta etapa, os membros optaram por criar um diagrama deste sistema para sua explanação no Capítulo 4.

3.10.17 Sprint 16

Nesta fase do projeto, várias listagens e a criação de contas, tanto para o perfil cliente, quanto prestador já estavam finalizadas. Logo o principal foco foi iniciar a integração do aplicativo com os recursos de mapa e geolocalização para o cliente conseguir pesquisar por prestadores próximos a sua localização.

Com o aprendizado da utilização dos formulários com react-native, a equipe aproveitou para construir alguns formulários de criação e edição que estavam pendentes. Entre as principais atividades, vale destacar:

- Construção de formulário de cadastro e edição para cartão de crédito, automóveis, serviços e formas de pagamento;
- Construção de tela para listagem de prestadores no mapa, com base na região de localização do usuário.

3.10.18 Sprint 17

Após a finalização de todas as atividades da sprint 16, dando sequência no escopo do projeto foram desenvolvidas as seguintes funcionalidades:

- Listagem de informações detalhadas de prestador;
- Listagem de prestadores (para casos em que o cliente não possui

prestadores próximos a sua geolocalização),

- Listagem de atendimentos para cliente;
- Listagem de atendimentos para prestador;
- Listagem de formas-pagamento de um prestador;
- Listagem de serviços de um prestador;
- Listagem de automóveis de um cliente.

Os principais desafios desta etapa para a equipe estavam diretamente relacionados com questões de estilização e responsividade das telas do aplicativo, afim de garantir sempre o mesmo comportamento independente do dispositivo utilizado, tentando gerar uma boa experiência de uso para os usuários.

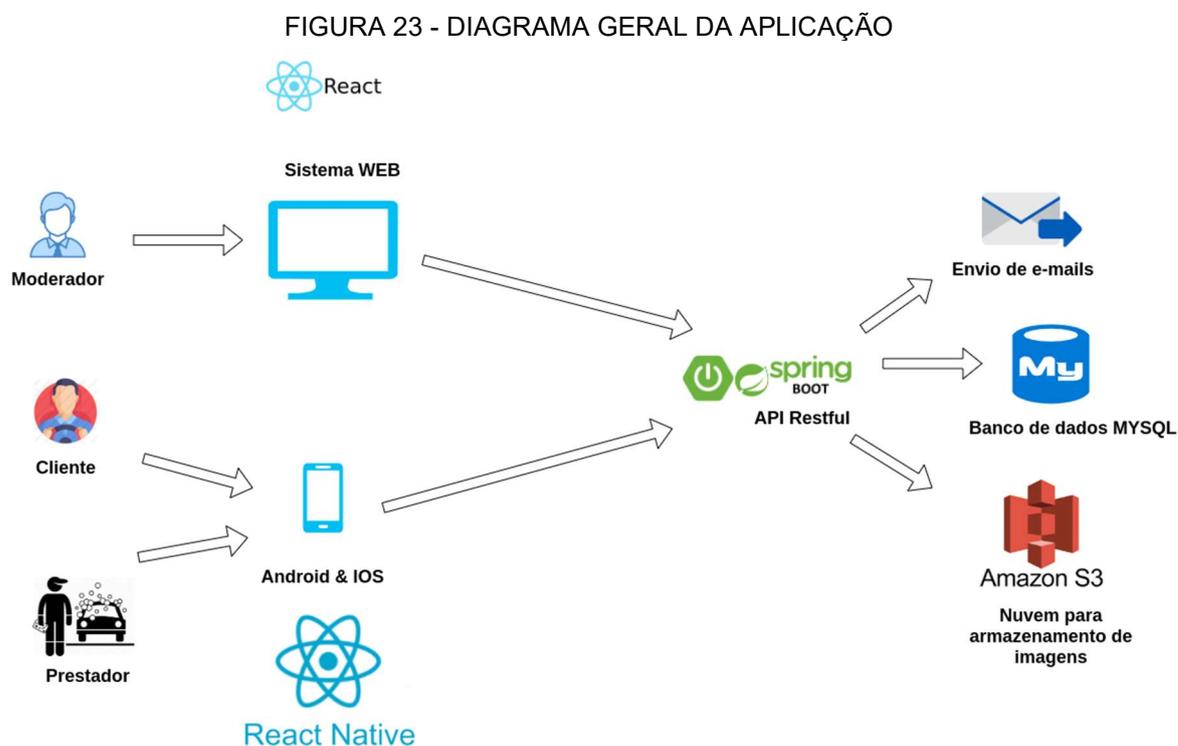
4 APRESENTAÇÃO DO SISTEMA

Neste capítulo será apresentada uma descrição completa da arquitetura do sistema desenvolvido e do funcionamento do mesmo, demonstrando as funcionalidades presentes e sua utilização.

Foram analisados diferentes *softwares* existentes no mercado para a lavagem de veículos durante a pesquisa deste projeto, nesta análise ficaram evidentes alguns pontos que nenhum deles possuía: lavagem com serviço leva a traz, preços definidos pelo prestador e um controle preciso por meio de um moderador.

Visando essa deficiência no mercado, o Easy Wash veio com o intuito de suprir essas necessidades e trazer mais conforto para o usuário, permitindo pesquisas e agendamentos rápidos, por meio de uma interface leve e intuitiva. Tanto o usuário como o prestador de serviços possuem um cadastro simples e seguro, precisando de aprovação do moderador para o cadastro do prestador.

O sistema possui uma aplicação *web* para moderadores, e uma aplicação móvel para clientes e prestadores de serviço. Na FIGURA 23 é possível visualizar de forma geral, o projeto como um todo e as tecnologias utilizadas.



FONTE: Os Autores (2020)

Nos itens seguintes serão realizadas a apresentação desta aplicação de que conecta lavadores de automóveis a clientes.

4.1 ARQUITETURA DO SISTEMA

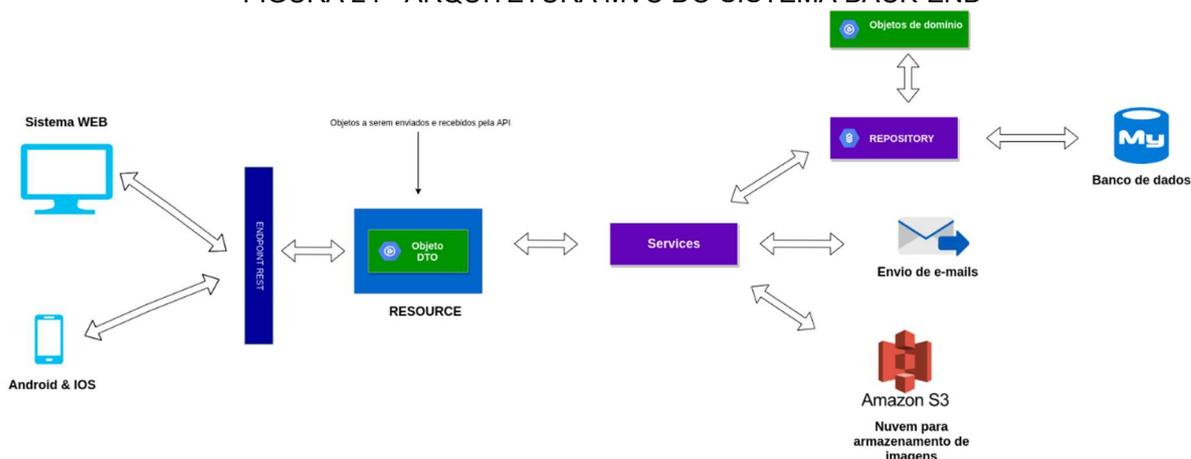
O sistema foi desenvolvido utilizando as linguagens de programação Java e Javascript, sendo a biblioteca React para a aplicação *web* e React Native para a aplicação móvel. O framework Spring Boot API Restful controla a camada backend da aplicação.

A aplicação back-end neste projeto, possui como principal objetivo ser a responsável de forma centralizada por realizar todas as operações de acesso ao banco de dados do aplicativo, além de direcionar a comunicação com outros serviços externos que foram necessários neste projeto, como por exemplo envio de e-mails e salvamento de imagens dos perfis dos usuários em nuvem.

A estrutura desta camada utiliza o padrão de arquitetura MVC (*Model-View-Controller*) mostrado na FIGURA 24.

Conforme demonstrado na imagem é possível visualizar que as classes *Resource*, são as controladoras da aplicação e possuem a principal responsabilidade por receber e responder todas as requisições. Estas classes recebem as informações enviadas e direcionam para as camadas *Services* que geralmente realizam algum tratamento sobre estas informações como conversões e direcionam para a próxima camada que de fato vai resolver esta requisição. Neste caso pode ser um direcionamento das informações para as classes *Persistence* que são responsáveis para a execução de operações relacionadas a banco de dados. Ou ainda as classes *Services*, podem direcionar para outras classes que realizam operações de acesso a serviços externos como envio de e-mails ou imagens para a nuvem.

FIGURA 24 - ARQUITETURA MVC DO SISTEMA BACK-END



FONTE: Os Autores (2020)

4.2 FUNCIONALIDADES

O sistema Easy Wash é composto de funcionalidades dedicadas para o usuário poupar seu tempo com agendamentos, e o prestador controlar melhor seu negócio. A seguir, segue a listagem destas funções:

- Sistema *web*
 - Login na aplicação;
 - Recuperação de senhas;
 - Criação de moderadores;
 - Listagem de moderadores do sistema;
 - Edição e exclusão dos moderadores;
 - Listagem de clientes do sistema;
 - Listagem de prestadores do sistema;
 - Exclusão de informações do prestador;
 - Listagem de novos prestadores para aprovações ou recusa;
 - Listagem de notificações enviadas pelos usuários;
 - Criação de notificações para clientes, prestadores ou ambos;
 - Relatórios com informações relevantes ao moderador do sistema.
- Sistema Mobile
 - Login no aplicativo para perfis cliente e prestador;
 - Recuperação de senhas para ambos os perfis;
 - Cadastro e edição de clientes;

- Cadastro e edição de prestadores;
- Cadastro e edição de veículos;
- Cadastro e edição de formas de pagamento;
- Cadastro e edição de imagem de perfil;
- Cadastro de notificação;
- Listagem de notificação;
- Pesquisa por prestador, via mapa ou listagem simples;
- Listagem de serviços de um prestador;
- Agendamento de serviço do prestador selecionado;
- Realizar atendimento do serviço.

4.3 SOBRE O SISTEMA WEB

O sistema *web*, exclusivo para o moderador, pode ser acessado pela seguinte url: <https://easywash.netlify.app>.

Na tela de login, inicia-se apresentando a interface de autenticação (FIGURA 25), na qual após o moderador se autenticar acessar e utilizar os recursos do sistema, através do preenchimento dos campos login e senha, dados que já devem estar previamente cadastrados no sistema. Caso o usuário tenha esquecido a senha, poderá solicitar uma nova que será gerada pelo sistema e enviada ao e-mail cadastrado (FIGURA 26). O armazenamento de senha é protegido pela criptografia md5 e os dados dos usuários são protegidos pela validação de sessão.

FIGURA 25 - PAGINA DE LOGIN

Faça seu logon

E-mail

Senha

Logon

↪ Esqueci minha senha



FONTE: Os Autores (2020)

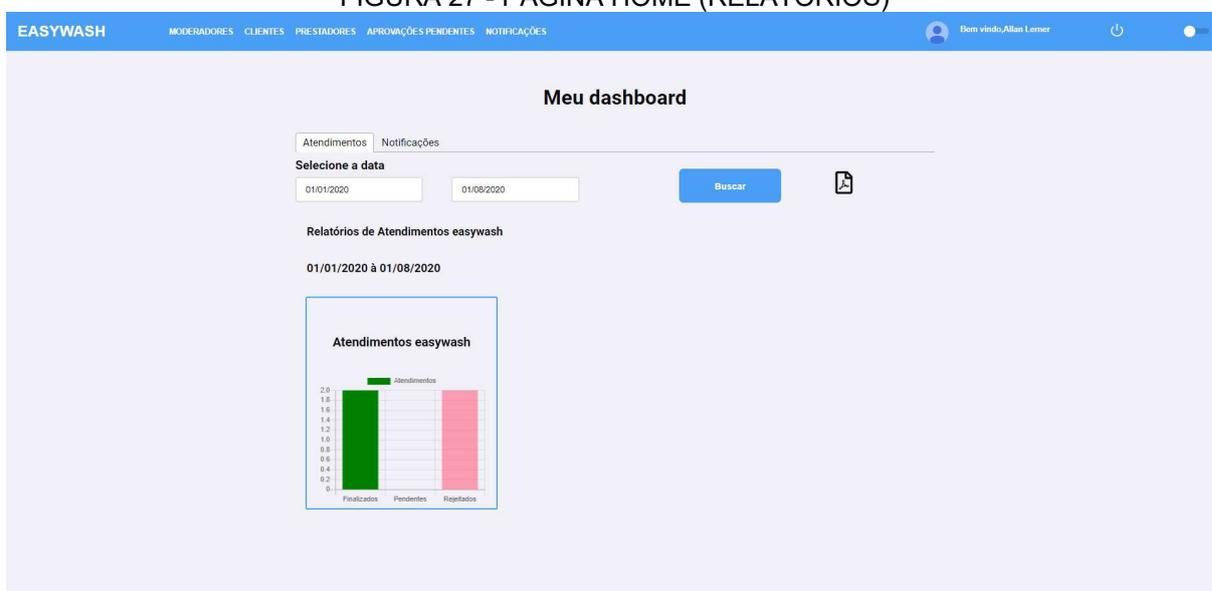
FIGURA 26 - ESQUECI A SENHA



FONTE: Os Autores (2020)

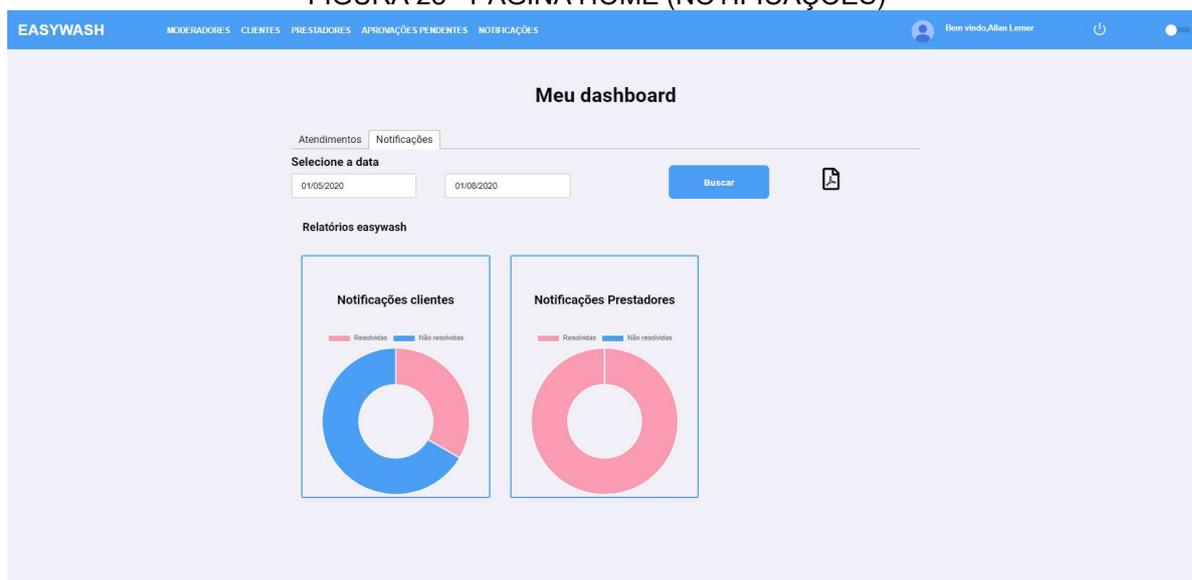
Após realização de autenticação, o usuário é redirecionado para a página inicial da aplicação (FIGURA 27 e 28), onde estão disponíveis os relatórios do sistema. O moderador pode visualizar e exportar para PDF gráficos de atendimentos e notificações.

FIGURA 27 - PAGINA HOME (RELATÓRIOS)



FONTE: Os Autores (2020)

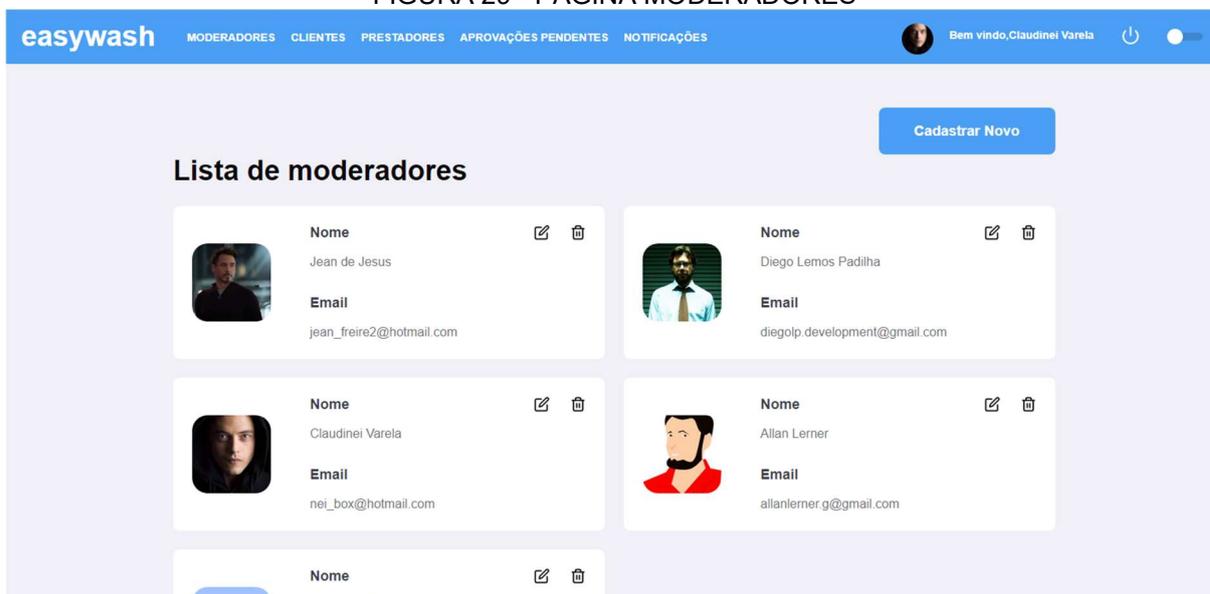
FIGURA 28 - PAGINA HOME (NOTIFICAÇÕES)



FONTE: Os Autores (2020)

Ao clicar na aba “moderadores”, uma lista com todos os moderadores será carregada. Nessa página, poderão ser realizadas alterações, exclusões e cadastros de novos moderadores (FIGURA 29 e 30).

FIGURA 29 - PÁGINA MODERADORES



FONTE: Os Autores (2020)

FIGURA 30 - PÁGINA NOVO MODERADOR

FONTE: Os Autores (2020)

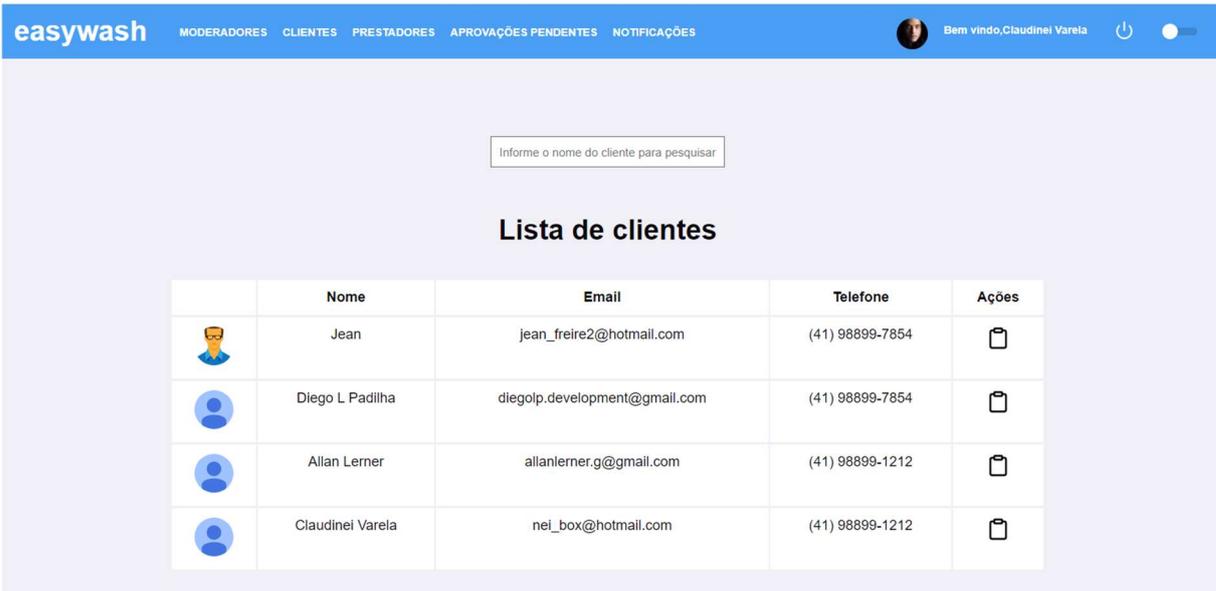
Ao clicar no ícone no campo superior direito, é possível ativar o “modo noturno” do site (FIGURA 31), para oferecer descanso das vistas dos usuários.

FIGURA 31 - PÁGINA MODERADORES MODO NOTURNO

FONTE: Os Autores (2020)

Ao clicar na aba “clientes”, é disponibilizada uma lista com todos os clientes cadastrados no sistema (FIGURA 32). Nessa página os moderadores poderão acessar informações detalhadas sobre um cliente (FIGURA 33) e poder enviar notificações (FIGURA 34).

FIGURA 32 - PÁGINA CLIENTES



easywash MODERADORES CLIENTES PRESTADORES APROVAÇÕES PENDENTES NOTIFICAÇÕES Bem vindo, Claudinei Varela

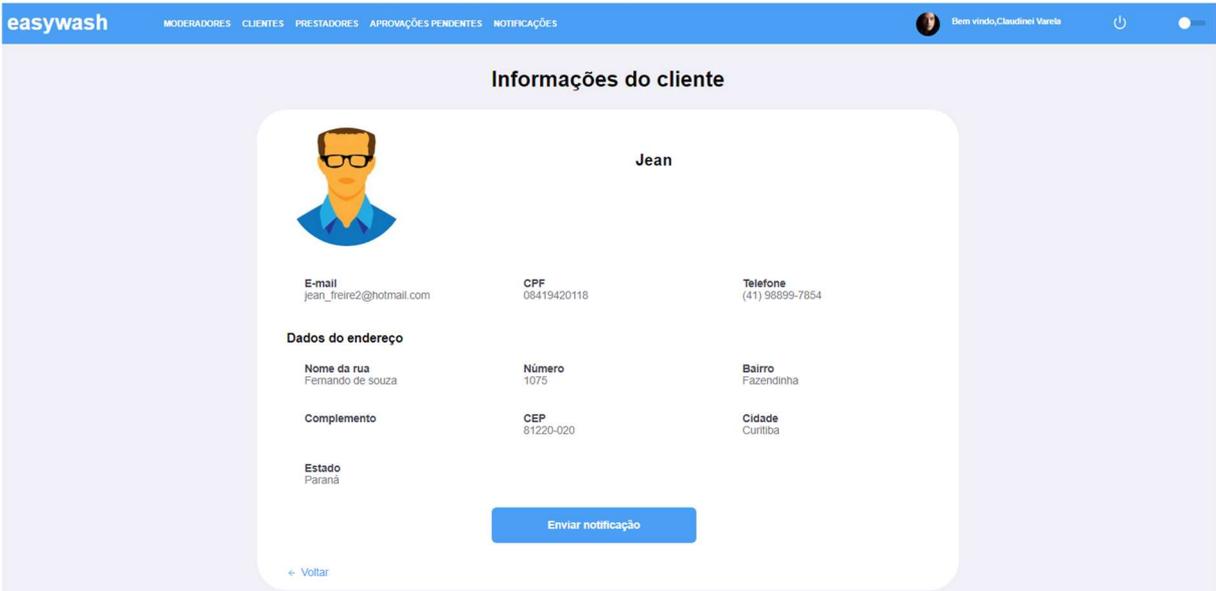
Informe o nome do cliente para pesquisar

Lista de clientes

	Nome	Email	Telefone	Ações
	Jean	jean_freire2@hotmail.com	(41) 98899-7854	
	Diego L Padilha	diegolp.development@gmail.com	(41) 98899-7854	
	Allan Lerner	allanlerner.g@gmail.com	(41) 98899-1212	
	Claudinei Varela	nei_box@hotmail.com	(41) 98899-1212	

FONTE: Os Autores (2020)

FIGURA 33 - PÁGINA CLIENTE DETALHES



easywash MODERADORES CLIENTES PRESTADORES APROVAÇÕES PENDENTES NOTIFICAÇÕES Bem vindo, Claudinei Varela

Informações do cliente



Jean

E-mail: jean_freire2@hotmail.com CPF: 08419420118 Telefone: (41) 98899-7854

Dados do endereço

Nome da rua: Fernando de souza Número: 1075 Bairro: Fazendinha

Complemento: CEP: 81220-020 Cidade: Curitiba

Estado: Paraná

[Enviar notificação](#)

[+ Voltar](#)

FONTE: Os Autores (2020)

FIGURA 34 - PÁGINA CLIENTES ENVIAR NOTIFICAÇÃO

Enviar notificação

Use este formulário para enviar notificações aos usuários da plataforma Easywash.

Mensagem

Insira uma mensagem aqui

Enviar

[← Voltar](#)

FONTE: Os Autores (2020)

Ao clicar na aba “prestadores”, é apresentada a lista com todos os prestadores de serviço cadastrados (FIGURA 35). Nessa página os moderadores poderão acessar informações detalhadas sobre um prestador (FIGURA 36 e 37), podendo enviar notificações para o mesmo e excluir informações, caso necessário.

FIGURA 35 - PÁGINA PRESTADORES

Lista de prestadores

	Nome	Email	Telefone	Ações
	Lava Car Coxa Doido LTDA	diegolp.development@gmail.com	(41) 99801-8745	
	Lava Car Top Gear	diegolemos@ufpr.br	(41) 98888-8745	

FONTE: Os Autores (2020)

FIGURA 36 - PÁGINA PRESTADORES 1

The screenshot shows the 'easywash' web application interface. The top navigation bar includes 'MODERADORES', 'CLIENTES', 'PRESTADORES', 'APROVAÇÕES PENDENTES', and 'NOTIFICAÇÕES'. The user is logged in as 'Bem vindo, Allan Lerner'. The main content area is titled 'Informações do prestador' and displays the following details for 'Lava Car Coxa Doido LTDA':

- Status:** Pendente
- E-mail:** diegoip.development@gmail.com
- CNPJ:** 640734580777309
- Telefone:** (41) 99801-8745
- Dados do endereço:**
 - Logradouro:** Fernando de souza
 - Número:** 10
 - Bairro:** Fazendinha

FONTE: Os Autores (2020)

FIGURA 37 - PÁGINA PRESTADORES 2

The screenshot shows a table of services and payment methods. The table has the following structure:

Código	Descrição	Duração	Valor	Ações
1	Lavagem Interna	120 min	R\$ 30,00	
11	Lavagem completa	140 min	R\$ 80,00	

Below the table, there is a section for 'Formas de pagamento' with the following details:

Código	Descrição	Ações
2	Dinheiro	

At the bottom, there is a form for 'Inserir justificativa' with a text input field containing 'Insira uma justificativa aqui', a dropdown menu for 'Selecione sua decisão para esta solicitação', and a blue 'Enviar' button.

FONTE: Os Autores (2020)

Ao clicar na aba “aprovações pendentes”, é apresentada todas as solicitações pendentes dos novos prestadores de serviço (FIGURA 38). Clicando na ação detalhes, o moderador terá acesso aos dados do prestador e aos serviços oferecidos (FIGURA 39).

FIGURA 38 – PÁGINA APROVAÇÕES PENDENTES



Solicitações de prestadores pendentes

	Nome	Email	Telefone	Ações
	Lava Car Coxa Doido LTDA	diegolp.development@gmail.com	(41) 99801-8745	

FONTE: Os Autores (2020)

FIGURA 39 - PÁGINA APROVAÇÕES PENDENTES DETALHES P1



Informações do prestador

 **Lava Car Coxa Doido LTDA**
Pendente

E-mail
diegolp.development@gmail.com

CNPJ
640734580777309

Telefone
(41) 99801-8745

Dados do endereço

Logradouro
Fernando de souza

Número
1075

Bairro
Fazendinha

Complemento

CEP
81220-020

Cidade
Curitiba

Estado
Paraná

FONTE: Os Autores (2020)

No final da página, o moderador deverá aprovar ou rejeitar o cadastro do prestador (FIGURA 40), sempre inserindo uma justificativa. As opções disponíveis são: aprovar, recusar ou manter o cadastro pendente (solicitar ajustes do prestador).

FIGURA 40 - PÁGINA APROVAÇÕES PENDENTES DETALHES P2

Serviços atuais

Código	Descrição	Duração	Valor	Ações
1	Lavagem Interna	120 min	R\$ 30,00	
11	Lavagem completa	140 min	R\$ 80,00	

Formas de pagamento

Código	Descrição	Ações
1	Cartão de crédito	
2	Dinheiro	

Inserir justificativa

Insira uma justificativa aqui

Selecione sua decisão para esta solicitação

Enviar

FONTE: Os Autores (2020)

Ao clicar na aba “notificações”, será apresentado uma lista contendo todas as notificações enviadas pelos usuários do aplicativo (FIGURA 41). Também é possível filtrar essas informações por perfil de usuário ou *status* da notificação.

FIGURA 41 - PÁGINA NOTIFICAÇÕES

easywash MODERADORES CLIENTES PRESTADORES APROVAÇÕES PENDENTES NOTIFICAÇÕES Bem vindo, Claudinei Varela

Lista de notificações

Cadastrar Novo

Selecione um status para pesquisa

Código	Data	Tipo de usuário	Mensagem	Ações
51	05/05/2020 03:05	Prestador	Boa noite não ta funcionando	
71	18/05/2020 01:48	Prestador	Olá prestador. Estamos estornando pagamentos em sua conta devido erro do nosso sistema	

FONTE: Os Autores (2020)

Clicando na ação, o moderador pode visualizar os detalhes da notificação (FIGURA 42). No final da página, para encerrar a notificação, o moderador pode inserir uma justificativa e preencher se o caso foi resolvido ou não, finalizando o evento.

FIGURA 42 - PÁGINA NOTIFICAÇÕES DETALHES

easywash MODERADORES CLIENTES PRESTADORES APROVAÇÕES PENDENTES NOTIFICAÇÕES Bem vindo, Claudinei Varela

Detalhes da notificação

Dados do usuário

Nome Lava Car Coxa Doido LTDA	CPF / CNPJ 640734580777309
Email diegojb.development@gmail.com	Telefone (41) 99801-8745

Dados da notificação

Data
05/05/2020 03:05

Conteúdo
Boa noite não ta funcionando

Inserir justificativa

Insira uma justificativa aqui

Selecione sua decisão para esta solicitação

Enviar

[← Voltar](#)

FONTE: Os Autores (2020)

Também na página de notificações (FIGURA 42) os moderadores podem criar novas notificações (FIGURA 43), clicando no botão “cadastrar novo”. Essas notificações tem a opção de serem enviadas para todos os usuários, todos os clientes ou todos os prestadores. Esta é uma ferramenta muito útil aos moderadores, que permite o aviso prévio de manutenção, atualizações ou novidades do sistema.

FIGURA 43 - PÁGINA NOTIFICAÇÕES NOVA NOTIFICAÇÃO

easywash MODERADORES CLIENTES PRESTADORES APROVAÇÕES PENDENTES NOTIFICAÇÕES Bem vindo, Claudinei Varela

Enviar notificação

Use este formulário para enviar notificações aos usuários da plataforma Easywash.

[← Voltar](#)

Todos os usuários

Mensagem

Somente nesse final de semana, ganhe \$7 de desconto em qualquer serviço contratado!

Enviar

FONTE: Os Autores (2020)

4.4 SOBRE O SISTEMA MÓVEL

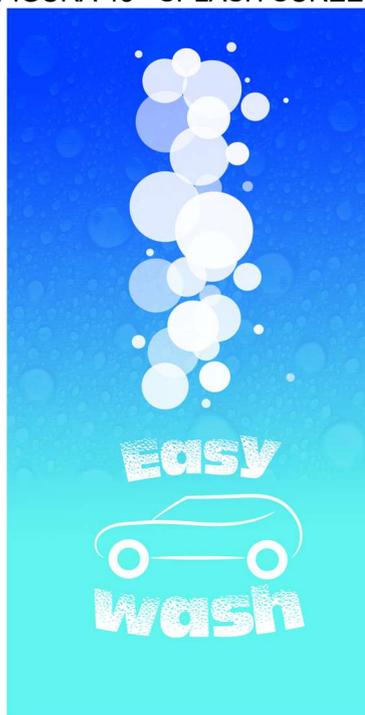
Para ter acesso ao sistema móvel, o usuário deve clicar no ícone do aplicativo (FIGURA 44). Assim que iniciado, a primeira tela a ser carregada é a *Splash Screen* (FIGURA 45), tela que contém o nome e a logo do sistema.

FIGURA 44 - ICONE DO APLICATIVO



FONTE: Os Autores (2020)

FIGURA 45 - SPLASH SCREEN



FONTE: Os Autores (2020)

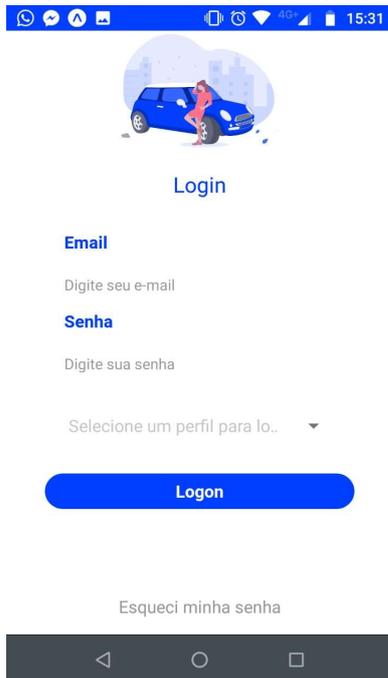
Após alguns segundos, é apresentada ao usuário a tela inicial do aplicativo (FIGURA 46). Caso ele já possua conta, basta clicar no botão “Já tenho uma conta!” que será direcionado para a tela de login (FIGURA 47). Se não possuir uma conta, deverá clicar na opção “Quero me tornar um cliente” para ser conduzido para o cadastro de clientes (FIGURA 48) ou na opção “Quero me tornar um prestador” para ser direcionado ao cadastro de prestadores (FIGURA 49).

FIGURA 46 - TELA DE INICIAL



FONTE: Os Autores (2020)

FIGURA 47 – TELA DE LOGIN



FONTE: Os Autores (2020)

FIGURA 48 - TELAS DE CADASTRO CLIENTE

The figure displays two sequential screens for client registration. The left screen, titled 'Dados Pessoais', features a progress indicator with '1' highlighted and '2' faded. It contains six text input fields: 'Digite seu nome', 'Digite seu cpf', 'Digite data de nascimento', 'Digite seu e-mail', 'Digite sua senha', and 'Digite seu telefone'. A 'Próximo' button is positioned at the bottom. The right screen, titled 'Dados de endereço', has a progress indicator with '1' faded and '2' highlighted with a checkmark. It includes five text input fields: 'Digite seu Cep', 'Digite seu logradouro', 'Digite seu bairro', 'Digite número', and 'Digite seu complemento', followed by two dropdown menus for 'Selecione estado...' and 'Selecione cidade...'. 'Voltar' and 'Salvar' buttons are at the bottom.

FONTE: Os Autores (2020)

FIGURA 49 – TELAS DE CADASTRO PRESTADOR

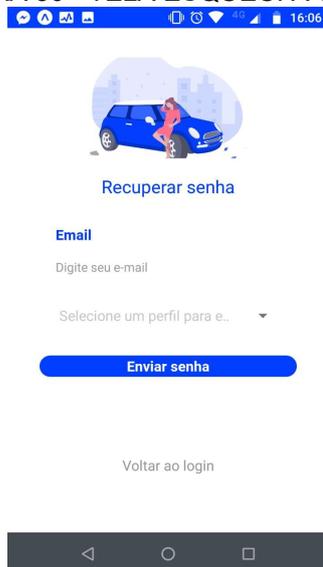
The figure displays two sequential screens for provider registration. The left screen, titled 'Dados empresa', features a progress indicator with '1' highlighted and '2' faded. It contains five text input fields: 'Digite sua Razão social', 'Digite seu cnpj', 'Digite seu e-mail', 'Digite sua senha', and 'Digite seu telefone'. A 'Próximo' button is positioned at the bottom. The right screen, titled 'Dados de endereço', has a progress indicator with '1' faded and '2' highlighted with a checkmark. It includes five text input fields: 'Digite seu Cep', 'Digite seu logradouro', 'Digite seu bairro', 'Digite número', and 'Digite seu complemento', followed by two dropdown menus for 'Selecione estado...' and 'Selecione cidade...'. 'Voltar' and 'Salvar' buttons are at the bottom.

FONTE: Os Autores (2020)

Para concluir o cadastro, é necessário preencher os campos obrigatórios e avançar até a última tela, ambos os cadastros (do cliente e do prestador) são compostos de uma sequência de telas.

Caso o usuário já tenha uma conta, mas não se recorde da senha, basta clicar no link “esqueci minha senha” dentro da tela de login (FIGURA 47) e será redirecionado para a tela de recuperação de senhas (FIGURA 50).

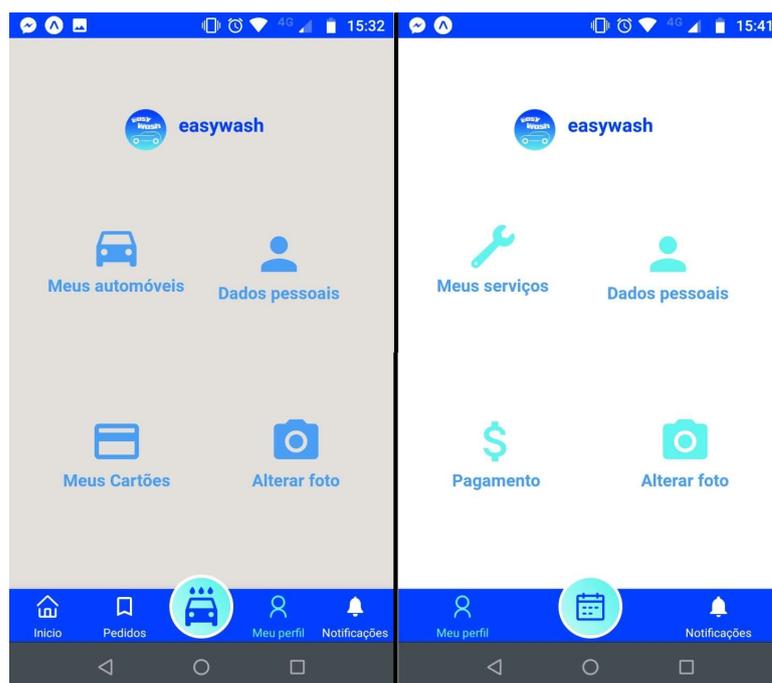
FIGURA 50 - TELA ESQUECI A SENHA



FONTE: Os Autores (2020)

Após o login efetuado com sucesso no sistema, será apresentada a tela principal do aplicativo (FIGURA 51).

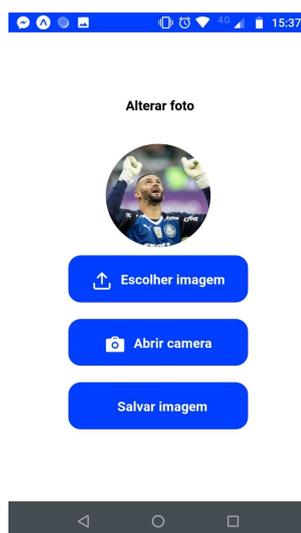
FIGURA 51 - TELA PRINCIPAL DO SISTEMA



FONTE: Os Autores (2020). LEGENDA: a esquerda se encontra a tela principal do cliente; a direita a tela principal do prestador.

Para realizar a edição dos dados pessoais, selecionando o menu “Dados pessoais”, uma tela semelhante ao de cadastro será apresentada com as informações já carregadas para a edição (FIGURAS 48 e 49). O mesmo ocorre para o menu “Alterar foto”, o cadastro e edição ocorrem na mesma tela (FIGURA 52).

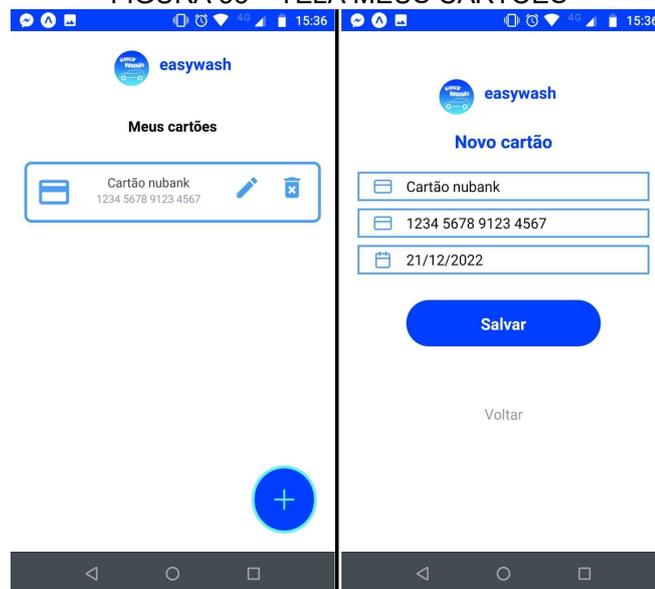
FIGURA 52 – TELA CADASTRO FOTO



FONTE: Os Autores (2020)

Clicando no menu “Meus Cartões” (FIGURA 51), é apresentada uma tela com a listagem dos cartões cadastrados, podendo realizar a inclusão, edição e exclusão dos mesmos (FIGURA 53).

FIGURA 53 – TELA MEUS CARTÕES



FONTE: Os Autores (2020)

Acessando o menu “Meus automóveis” (FIGURA 51), é apresentada uma tela com a listagem dos veículos cadastrados, podendo realizar a inclusão, edição e exclusão dos mesmos (FIGURA 54).



FONTE: Os Autores (2020)

Clicando no menu “Pagamento” (FIGURA 51), é apresentada uma tela com a listagem dos métodos de pagamento aceitos pelo prestador, podendo realizar o gerenciamento desses dados, selecionando as formas de pagamento e clicando em salvar (FIGURA 55).



FONTE: Os Autores (2020)

Ao clicar no botão “Meus serviços” (FIGURA 51), é apresentada uma tela com a listagem dos serviços cadastrados, podendo realizar a inclusão, edição e exclusão dos mesmos (FIGURA 56).



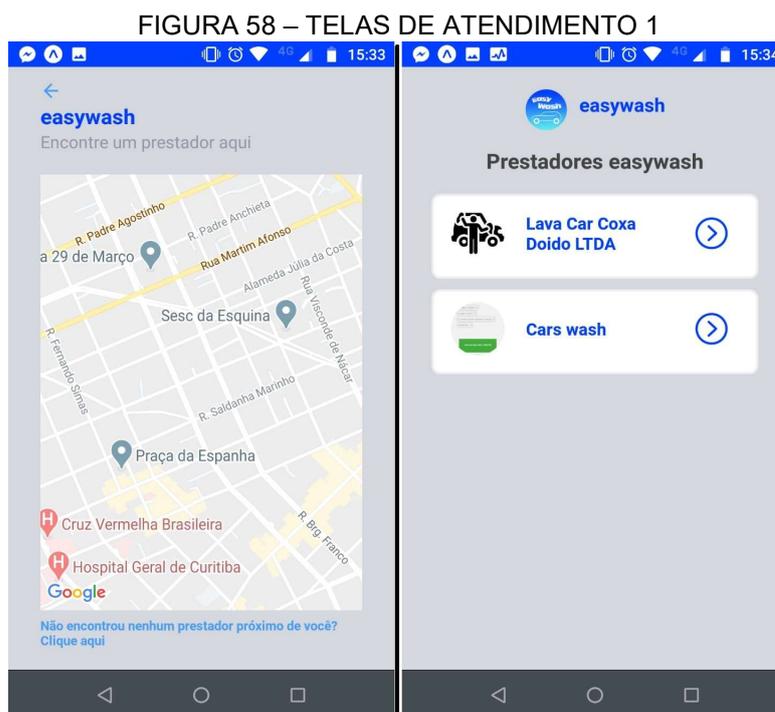
FONTE: Os Autores (2020)

Para visualizar as notificações do sistema, basta clicar no menu inferior direito (FIGURA 51). Nesta tela é possível listar as notificações recebidas e enviar possíveis problemas ao moderador (FIGURA 57).



FONTE: Os Autores (2020)

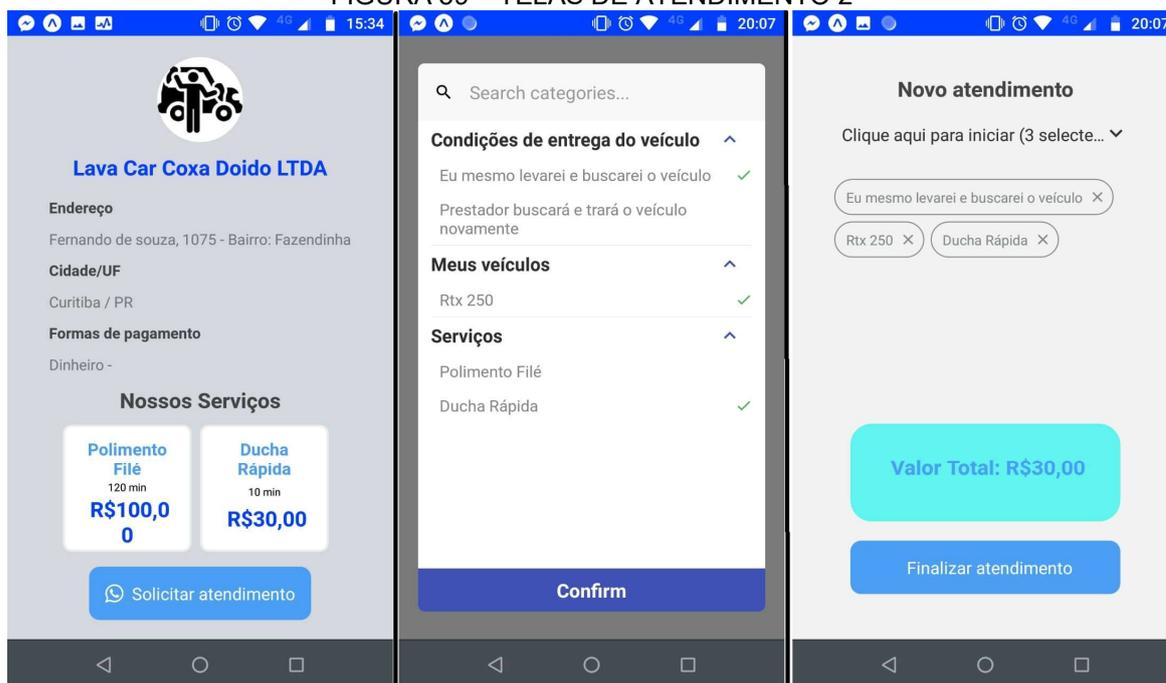
Clicando no botão em destaque na tela, do ícone de carro para o cliente (FIGURA 51), é iniciado o processo de solicitação de atendimento. A primeira tela apresentada é um mapa, em que é possível pesquisar por um prestador. Caso não seja encontrado um próximo, é possível clicar no botão abaixo do mapa para ser carregada uma lista com os prestadores (FIGURA 58).



FONTE: Os Autores (2020)

Ao selecionar o prestador desejado, é possível consultar as informações do mesmo juntamente com os serviços e seus valores. Clicando em “Solicitar atendimento”, o cliente deverá selecionar as informações do seu atendimento e clicar em “Finalizar atendimento” (FIGURA 59).

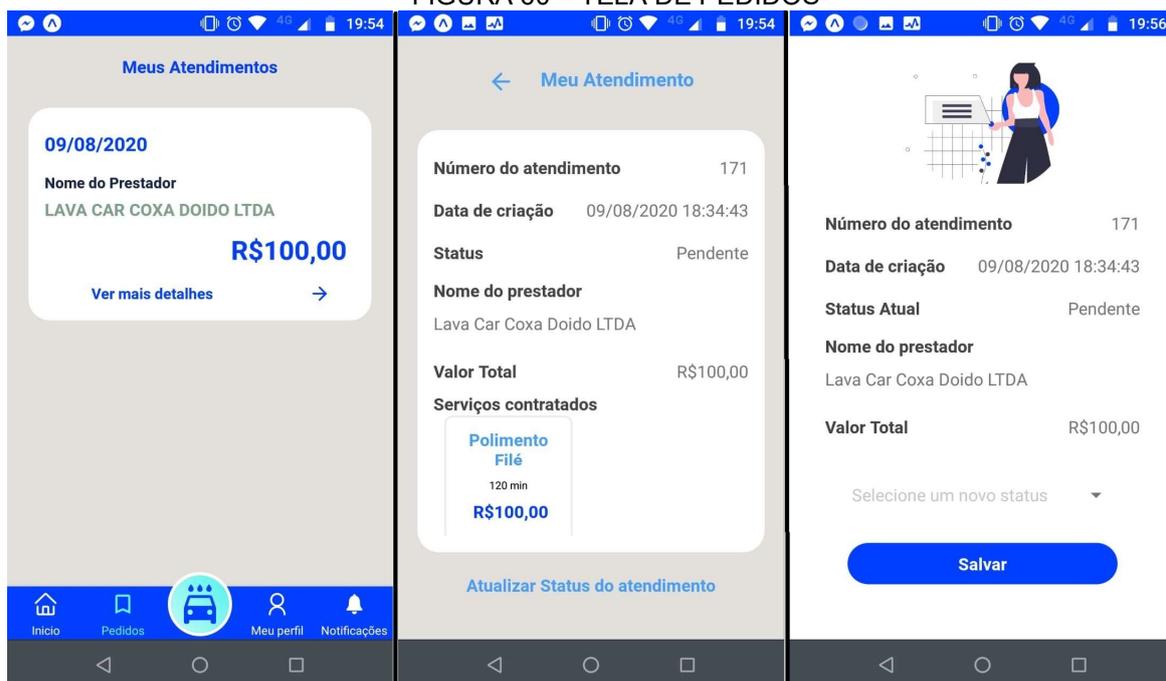
FIGURA 59 – TELAS DE ATENDIMENTO 2



FONTE: Os Autores (2020)

Para consultar o andamento deste novo atendimento, o cliente pode clicar no menu inferior “Pedidos” (FIGURA 51). Nesta tela serão listados todos os pedidos realizados, suas informações e a possibilidade de atualizar o status do pedido, cancelando este atendimento (FIGURA 60).

FIGURA 60 – TELA DE PEDIDOS

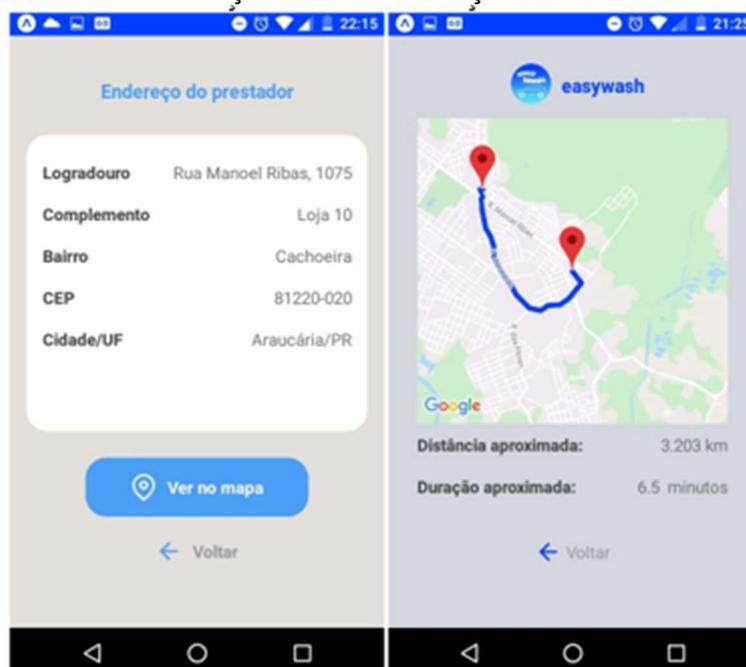


FONTE: Os Autores (2020)

Outro ponto importante que vale ressaltar é que o cliente tem a possibilidade de cancelar o atendimento enquanto este tiver com situação “Pendente”. A partir do momento que o prestador aceita este pedido, o sistema altera o status do mesmo e a partir deste momento a opção de cancelar não é mais apresentada ao cliente.

Além disso, o cliente pode consultar as informações de endereço do prestador. Tanto de forma estática quanto pelo Google Maps (FIGURA 61).

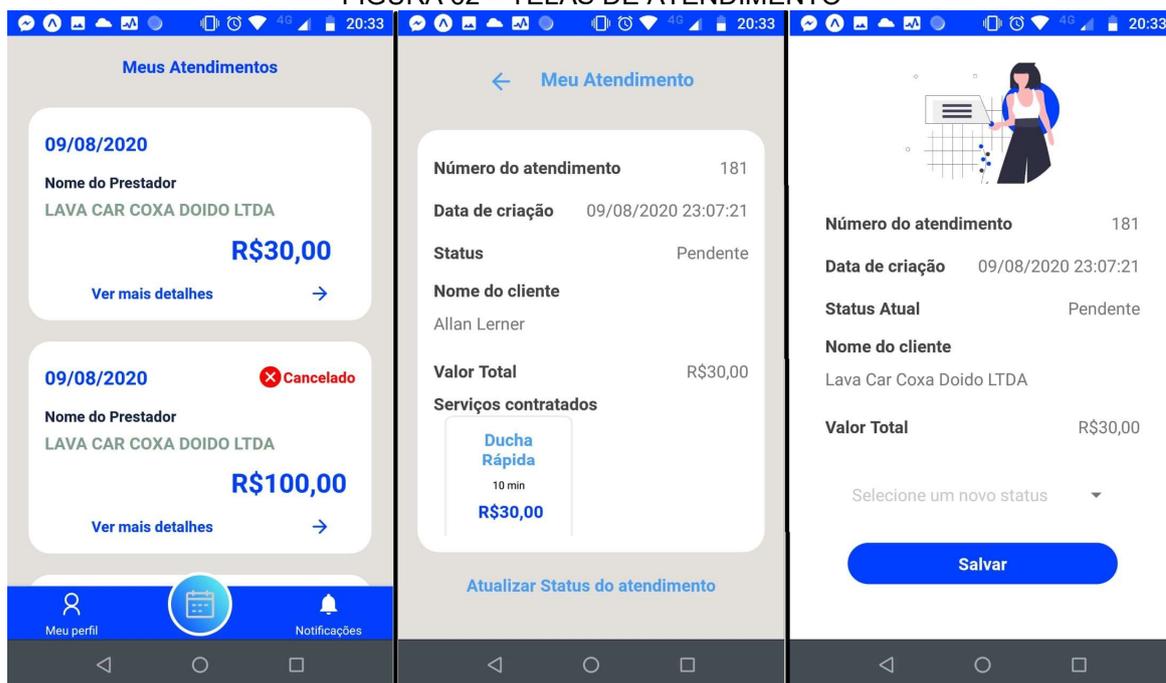
FIGURA 61 – INFORMAÇÕES DE ENDEREÇO DE CLIENTE/PRESTADOR



FONTE: Os Autores (2020)

Os prestadores podem realizar o gerenciamento dos atendimentos clicando no ícone calendário. Em seguida, serão listados os atendimentos com seus detalhes e status que poderão ser atualizados à medida que são finalizados (FIGURA 62).

FIGURA 62 – TELAS DE ATENDIMENTO



FONTE: Os Autores (2020)

5 CONSIDERAÇÕES FINAIS

O aplicativo EasyWash nasceu para valorizar o que temos de mais precioso na vida: nosso tempo. A questão de se perder tempo em tarefas cotidianas, como esperar em filas para lavar seu automóvel, é problemática, logo esta foi uma das motivações para construção deste projeto.

Além disso, identificamos outras necessidades em que o aplicativo poderia auxiliar, como por exemplo ser um elo de conexão entre clientes e prestadores. Para os clientes sendo um facilitador para encontrarem prestadores confiáveis e já para os prestadores, ser uma rede para divulgação de seus serviços.

Desde a idealização do projeto o tema provou-se ser de extrema dificuldade em relação as pesquisas literárias, e foi realmente o que aconteceu: apesar do assunto “lavação de carros” ser algo comum, os textos científicos ou acadêmicos para a construção da monografia mostraram ser escassos, principalmente em relação a obras literárias. Felizmente o tema apareceu em poucos artigos acadêmicos de diferentes vertentes, os quais foram bem aproveitados pelo grupo. Além disso, a construção deste projeto proporcionou a equipe um enorme aprendizado, que vai muito além do conteúdo transmitido pelos professores ao longo de todo o curso.

O tema tecnologia, atrelado com a otimização de tempo nos moldes atuais da sociedade, vem sendo altamente explorado, originando grandes casos de sucesso e trazendo enormes benefícios para a sociedade como, por exemplo, a geração de empregos, melhor qualidade de vida e redução do estresse no dia-a-dia, em função da praticidade dessas novas ferramentas. Muitas atividades cotidianas têm sido transformadas positivamente para se adequar as novas demandas, e nota-se que cada vez mais isso está se tornando uma tendência não só no país, mas ao redor de todo o mundo.

O resultado obtido ao final do projeto foi um sistema robusto, com interface web para moderação e interface móvel para os clientes e prestadores de serviço, mas devido à complexidade e tamanho do sistema, algumas questões precisaram ser adaptadas para atender o cronograma do projeto e funcionalidades foram removidas do escopo inicial, para serem incluídas em versões futuras.

Algumas limitações e dificuldades encontradas durante o desenvolvimento do projeto foram, principalmente, devido a linguagem de programação utilizada. Como a equipe não tinha muita experiência com as tecnologias definidas para este projeto,

houveram longos períodos de aprendizagem e algumas funcionalidades não supriram as necessidades como esperado, principalmente questões relacionadas a design do aplicativo para dispositivos móveis.

Acerca do gerenciamento do projeto, alguns dos grandes desafios enfrentados na construção do mesmo foram: as mudanças de escopo, indefinições de requisitos, o gerenciamento de pessoas, tempo e a comunicação, itens que se tornaram uma grande oportunidade para amadurecimento, crescimento pessoal e profissional de todos os integrantes da equipe.

Por fim, este trabalho também refinou os conhecimentos dos autores e os incitou a novos, como, por exemplo, o estudo de nova linguagem e paradigma de programação, fortalecimento das demais habilidades nas tecnologias já citadas no Capítulo 3, e um melhor entendimento na aplicação de normas técnicas.

5.1 RECOMENDAÇÕES PARA TRABALHOS FUTUROS

As recomendações para trabalhos futuros são dedicadas aos softwares desenvolvidos, onde algumas funcionalidades ou melhorias não foram implementadas devido a relação de escopo x tempo ou a problemas relacionados a complexidade da tecnologia. Algumas destas funcionalidades são:

- WEB
 - Implementação de uma maior variedade de relatórios.
- Back-end
 - Alterar a arquitetura do sistema para conceito de micro-serviços;
 - Implementação de testes unitários.
- Mobile
 - A possibilidade de utilizar Typescript para o desenvolvimento do aplicativo ao invés de Javascript;
 - Funcionalidade para o cliente avaliar o atendimento de um prestador;
 - Adição de cronômetro no atendimento do prestador;
 - Funcionalidade de pagamento pelo aplicativo;
 - Adição de filtros de pesquisa nas telas com listagem de informações;

- Aprimorar o design das interfaces. para uma melhor experiência do usuário

REFERÊNCIAS

ALLIANCE. **Agile Alliance**. Disponível em: <<https://www.agilealliance.org/agile101/>> Acessado em: 07 out. 2019.

APP WASH. **AppWash**. Disponível em: <https://play.google.com/store/apps/details?id=br.com.squarebits.appwash.jnb&hl=pt_BR>. Acesso em: 05 out. 2019.

ATLASSIAN. **Sobre o Trello**. s,d. Disponível em: <<https://trello.com/about>>. Acesso em: 01 nov. 2019.

BASTOS, Daniel Deggau. **A perda do tempo como categoria indenizatória autônoma terminologia jurídica e coerência sistemática**. 184f. Trabalho de Pós-Graduação (Programa de Pós-Graduação em Direito) - Centro De Ciências Jurídicas, Universidade Federal de Santa Catarina UFSC, Florianópolis, Santa Catarina, 2017. Disponível em: <https://www.academia.edu/35996215/A_perda_do_tempo_como_categoria_indenizatoria_autonoma_terminologia_juridica_e_coerencia_sistemica>. Acesso em: 01 out. 2019.

BLOG FREESIDER, Ronaldo. **A falta de tempo no mundo moderno**. 2016. Não paginado. Disponível em: < <https://freesider.com.br/tempo/falta-de-tempo-no-mundo-moderno/> >. Acessado em: 10 ago. 2019.

BOOCH, G; RUMSBAUGH, James; JACOBSON, Ivar. **UML – Guia do Usuário**. Rio de Janeiro: Elsevier, 2006.

BRASIL ECONÔMICO. **Mais da metade dos brasileiros não consegue ficar um dia longe do celular**. 2019. Disponível em: <<https://tecnologia.ig.com.br/2019-02-22/brasileiros-nao-ficam-sem-celular.html>>. Acesso em: 22 out. 2019.

CAMARGO, Ricardo. **Teoria das Filas**. 6f. Disponível em: <https://www.academia.edu/31906493/Teoria_das_Filas>. Acesso em: 10 ago. 2019.

CAMPELO, Ana Carolina; CAVALCANTI, Layon Felipe. **Teoria das filas: Métodos aplicados**. Disponível em: <https://www.academia.edu/19621942/Artigo_Teoria_das_Filas> Acesso em: 23 mar. 2020.

CHANGEVISION INC. **Astah Professional**. Disponível em: <<http://astah.net/editions/professional>>. Acesso em: 28 out. 2019.

CURRITO, Eduardo. **Satisfação, Valor e Retenção do cliente**. Disponível em: <https://www.academia.edu/6857824/SATISFA%C3%87%C3%83O_VALOR_E_RETEN%C3%87%C3%83O_DO_CLIENTE>. Acesso em: 15 ago. 2019.

DB-ENGINES. **Db-engines**. Disponível em: <<https://db-engines.com/en/ranking>> Acesso em: 10 set. 2019.

ECLIPSE FOUNDATION. **Eclipse IDE The Landing Open Platform for Professional Developers**. Disponível em: <<https://www.eclipse.org/eclipseide/>>. Acesso em: 10 set. 2019.

ELMASRI, R.; NAVATHE. **Sistema de banco de dados**. São Paulo: Pearson Addison Wesley, 2011.

EQUIPE ADMINISTRADORES. **Falta de tempo afeta 62% dos brasileiros**. Disponível em: <<https://administradores.com.br/noticias/falta-de-tempo-afeta-62-dos-brasileiros>>. Acesso em: 10 ago. 2019.

FACEBOOK DEVELOPERS. **Conferência de React**. 2015. Disponível em: <<https://www.youtube.com/watch?v=KVZ-P-ZI6W4&t=0s&list=PLb0IAmt7-GS1cbw4qonIQztYV1TAW0sCr&index=1>>. Acesso em: 02 out. 2019.

FACEBOOK INC. **Tutorial de Introdução ao React**. Disponível em: <<https://pt-br.reactjs.org/tutorial/tutorial.html>>. Acesso em: 14 de out. 2019.

FILHO, Mauro Rezende. **Uma aplicação da teoria das filas com simulação com arena em uma empresa de serviços de lavação automotiva**. Rio de Janeiro, 2016. Disponível em: <https://www.academia.edu/31335120/Uma_aplica%C3%A7%C3%A3o_da_teor%C3%A7a_das_filas_com_simula%C3%A7%C3%A3o_com_arena_em_uma_empresa_de_servi%C3%A7os_de_lava%C3%A7%C3%A3o_automotiva> Acesso em: 07 out. 2019.

GITHUB INC. **Sobre o Github**. Disponível em: <<https://github.com/about>>. Acesso em: 29 out. 2019.

GUGLINSKI, Vitor Vilela. **Danos morais pela perda do tempo útil: uma nova modalidade**. 2012. Disponível em: <<https://www.academia.edu/8926942/201766071-Indenizacao-pela-perda-de-tempo-Revista-Jus-Navigandi-Doutrina-e-Pecas>> Acesso em: 01 out. 2019.

JAVATPOINT. **What is database**. Disponível em :<<https://www.javatpoint.com/what-is-database>>. Acesso em: 20 set. 2019.

LARMAN, Craig. **Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development**. 2005, Pearson Education.

LAVÔ. **LAVÔ Tecnologia e Licenciadora**. Disponível em: <<https://lavo.online>>. Acesso em: 05 out. 2019.

LIMA, Adilson da Silva. **UML 2.3 do Requisito à solução**. 2013. São Paulo. Érica.

MANIFESTO ÁGIL. **Agile Manifesto**. Disponível em: <<https://agilemanifesto.org/iso/ptbr/manifesto.html>> Acessado em: 07 out. 2019.

MARTINS, José Carlos Cordeiro. **Gerenciando projetos de Desenvolvimento de Software com PMI, RUP e UML 4ªed**. 2007. Rio de Janeiro. Brasport.

MICROSOFT. **Documentação Visual Studio Code**. Disponível em: <<https://code.visualstudio.com/docs>>. Acesso: 27 out. 2019.

MOSAVI, Amir. **Mobile Apps - A book by Amir Mosavi**, PhD. 372f. Obusa University, Budapest, 2016.

MYSQL. **Oracle Corportation**. Disponível em: <<https://www.mysql.com/customers/>> Acesso em: 10 set. 2019.

NORMAN, Donald A. **The Psychology of Waiting Lines**. 21f. 2008.

NOVO NEGÓCIO. **Como Montar um lava-Jato de sucesso [Guia Completo 2020]**. Disponível em: <https://novonegocio.com.br/ideias-de-negocios/como-montar-um-lava-jato/#Mercado_de_veiculos_e_lava-jato_no_Brasil> Acesso em: 23 mar. 2020.

ORACLE. **Oracle Corporation**. Disponível em: <<https://www.oracle.com/br>> Acesso em: 10 set. 2019.

PHAM, Andrew; PHAM, Phuong-Van. **Scrum em ação: gerenciamento e desenvolvimento Ágil de projetos de software**. 2011, São Paulo: Novatec Editora.

PIVOTAL SOFTWARE INC. **Spring Boot Reference Documentation**. Disponível em: <<https://docs.spring.io/spring-boot/docs/current/reference/pdf/spring-boot-reference.pdf>>. Acesso em: 05 out. 2019.

PIVOTAL SOFTWARE INC. **Spring Framework Documentation**. Disponível em: <<https://docs.spring.io/spring/docs/current/spring-framework-reference/index.html>>. Acesso em: 05 out. 2019.

Risingstack. **Ferenc Hámori**. Disponível em: <<https://blog.risingstack.com/the-history-of-react-js-on-a-timeline/>> Acessado em: 10 set. 2019.

SABBAGH, Rafael. **Scrum: Gestão ágil para projetos de sucesso**. 2013, São Paulo: Casa do Código.

SCHWABER, Ken. **Agile Project Management with Scrum**. 2004, Microsoft Press.

SEBRAE/ES. **LAVAJATO – LAVARÁPIDO**. Disponível em: <<http://vix.sebraees.com.br/ideiasnegocios/arquivos/lavajato-lavarapido.pdf>>. Acesso em: 23 mar. 2020.

Stack overflow. **Stack Overflow's annual Developer Survey. Other Frameworks, Libraries, and Tools most populars**. Disponível em: <https://insights.stackoverflow.com/survey/2019#technology-_other-frameworks-libraries-and-tools> Acessado em: 10 set. 2019.

STACK OVERFLOW. **Stack Overflow's annual Developer Survey 2019. Most Populars Web Frameworks Technologies**. Disponível em:

<<https://insights.stackoverflow.com/survey/2019#technology--web-frameworks>>
Acessado em: 10 set. 2019.

XU, Jane. **Smartphone**. 28f. 2014. Academia EDU. Disponível em:
<<https://www.academia.edu/8659764/Smartphone>>. Acesso: 01 nov. 2019

YUILL, Simom. **Concurrent Versions System**. Estados Unidos da América, 2008.
Disponível em: <<http://www.lipparosa.org/essays/cvs.pdf>>. Acesso: 01 nov. 2019.

APÊNDICE A - LISTA DE REQUISITOS APLICATIVO MÓVEL CLIENTE

Este documento tem por objetivo descrever todos os requisitos do projeto para construção do aplicativo a ser utilizado pelos clientes da plataforma Easy Wash.

Descrição dos objetivos do sistema

O objetivo do aplicativo Cliente para o projeto Easy Wash, primeiramente e manter separadas as funcionalidades que são utilizadas exclusivamente pelos clientes. A partir do aplicativo Cliente, este poderá se cadastrar na plataforma para visualizar as empresas e seus respectivos serviços e posteriormente interagir com o prestador de serviços para efetivação uma contratação de serviço.

Requisitos Funcionais

RF001 – Realizar Login

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir o usuário realizar login no aplicativo para poder ter acesso a todos os recursos da plataforma. Para isso, o cliente deverá utilizar suas credenciais de e-mail e senha cadastrados no requisito RF002.

RF002 – Cadastrar Cliente

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir o cliente cadastrar-se no aplicativo. O cadastro na plataforma será dividido nas seguintes seções:

- Cadastro de dados pessoais: Nome completo, CPF, data de nascimento, telefone, e-mail e senha.
- Cadastro de dados do endereço do cliente: CEP, logradouro, bairro, complemento, estado e cidade.
- Cadastro de foto no perfil.

RF003 – Cadastrar automóvel

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao cliente cadastrar seu automóvel no sistema. O cliente poderá cadastrar um ou mais veículos. As informações para cadastro de um veículo são: Tipo, modelo, cor e placa.

RF004 – Gerenciar dados cadastrais

Prioridade: Essencial Importante Desejável

Está funcionalidade deverá permitir ao cliente gerenciar seus dados cadastrais no sistema. O gerenciamento de dados cadastrais é dividido em 4 seções, conforme descrito a seguir:

- **Minhas informações:** Nessa seção o cliente poderá visualizar e editar suas informações pessoais e de endereço.
- **Meus automóveis:** Nessa seção o cliente poderá visualizar, editar e excluir os dados referentes aos seus automóveis no aplicativo.
- **Formas de pagamento:** Nessa seção o cliente poderá visualizar, editar e excluir dados relacionados aos seus cartões de crédito cadastrados para realização de pagamentos no aplicativo.
- **Alterar Foto:** Aqui o cliente poderá carregar ou tirar uma foto para ser utilizada em seu perfil pessoal.

RF005 – Pesquisar serviços

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao cliente realizar uma pesquisa de serviços por filtro. Os tipos de filtros que a aplicação deverá possuir são: Pesquisa por localidade, preço e tipo de serviço.

RF006 – Visualizar notificações

Prioridade: Essencial Importante Desejável

Está funcionalidade deverá permitir ao cliente visualizar suas notificações, que serão enviadas tanto pela plataforma, quanto pelo prestador de serviço.

RF007 – Cadastrar atendimento

Prioridade: Essencial Importante Desejável

Está funcionalidade deverá permitir ao cliente solicitar a contratação de um ou mais serviços de um prestador de serviço. As informações necessárias para a contratação são:

- Data e hora do agendamento da lavagem;
- Tipo de retirada e entrega do veículo;
- Observações: Para o cliente relatar informações adicionais relacionadas ao atendimento, podendo ser informações de detalhes do serviço a ser realizado, algum cuidado com seu veículo, etc;
- Lista de serviços da empresa com opção de seleção para contratação.
- Duração prevista do atendimento (Soma total da duração de todos os serviços selecionados pelo cliente);
- Valor Total do pedido: (Soma do valor de todos os serviços);

RF008 – Realizar pagamento

Prioridade: Essencial Importante Desejável

Está funcionalidade deverá permitir ao cliente a seleção de uma forma de pagamento para efetivação de uma contratação de serviço na plataforma.

As formas de pagamento a serem disponibilizadas pelo aplicativo são as seguintes:

- Pagamento com cartão de crédito pelo aplicativo;

- Pagamento com cartão de crédito diretamente com o prestador;
- Pagamento com cartão de débito diretamente com o prestador;
- Pagamento com dinheiro diretamente com o prestador.

Nos pagamentos a serem realizados com cartão de crédito pelo aplicativo, o cliente poderá salvar os dados do seu cartão de crédito para serem utilizados nas próximas compras. Os dados do cartão de crédito a serem armazenados pelo aplicativo são:

- Nome completo (O mesmo nome que está no cartão);
- Número do cartão;
- Mês e ano de validade.

O código de verificação não será armazenado no aplicativo, porém essa informação será solicitada e validada para que o pagamento seja efetivado toda vez que o cliente realizar uma compra.

Outro ponto importante, as opções de forma de pagamento que serão realizadas diretamente com o prestador de serviços podem variar de acordo com a disponibilidade do mesmo.

RF009 – Consultar Histórico de serviços contratados

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao cliente visualizar todo seu histórico de atendimentos solicitados pelo aplicativo. O cliente poderá filtrar sua busca por período.

As informações a serem recuperadas referente a cada atendimento são as seguintes:

- Data e hora de atendimento;
- Modelo do veículo;
- Nome da empresa que realizou o atendimento;
- Lista com os serviços realizados;
- Valor total do pedido;
- Forma de pagamento;
- Nota de avaliação do cliente referente ao atendimento selecionado.

RF010 – Visualizar informações de uma empresa

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao cliente, após realizar uma pesquisa por determinada empresa, visualizar uma página de perfil da empresa com as seguintes informações:

- Nome da empresa;
- Endereço completo;
- Horário de atendimento;
- Listagem com todos serviços oferecidos;
- Avaliação da empresa no aplicativo (composta pela média de todas as avaliações recebidas pela mesma).

RF011 – Avaliar Serviços

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir o cliente avaliar serviços após conclusão do mesmo. A avaliação será composta por uma nota de uma a cinco estrelas e um comentário opcional a ser informado pelo cliente.

RF012 – Enviar mensagem para a moderação

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao cliente enviar mensagens a moderação, podendo ser para fins de elogio, reclamação ou suporte junto ao aplicativo.

Requisitos do sistema

RNF001 – Enviar e-mail com uma senha gerada automaticamente pelo sistema.

Quando o moderador, informar seu e-mail solicitando a recuperação de senha, a própria aplicação deve gerar uma senha e encaminhar para o usuário para o mesmo possa recuperar seu acesso.

RNF002 – Senhas devem estar criptografadas.

Todas as senhas devem estar criptografadas no banco de dados, utilizando o padrão MD5.

APÊNDICE B - LISTA DE REQUISITOS APLICATIVO MÓVEL PRESTADOR

Este documento tem por objetivo descrever todos os requisitos do projeto para construção do aplicativo a ser utilizado pelos prestadores de serviço cadastrados na plataforma Easy Wash.

Descrição dos objetivos do sistema

O aplicativo prestador da plataforma Easy Wash tem por principal objetivo ser um canal para empresas de serviços de lavagem de automóveis divulgarem seus trabalhos. O prestador poderá divulgar e vender seus serviços, de modo antecipado a uma vasta gama de clientes. Além disso, a plataforma ofereça ao prestador um controle referente ao seu histórico de serviços prestados, através de gráficos.

RF001 – Cadastrar-se no sistema

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir o prestador cadastrar-se no aplicativo para poder oferecer seus serviços. O cadastro na plataforma será dividido nas seguintes seções:

- Dados da empresa, compostas pelas seguintes informações: Nome da empresa, Razão social, CNPJ, telefone, e-mail e senha;
- Dados de endereço da empresa, compostos pelas seguintes informações: CEP, Logradouro, número, bairro, complemento, estado e cidade;
- Lista com os tipos de serviços a serem oferecidos pela empresa na plataforma. Cada serviço deverá ser composto pelas seguintes informações: Nome do serviço, descrição, valor e duração;
- Dados financeiros: O cliente deverá inicialmente cadastrar pelo menos uma conta, contendo as seguintes informações: Código do banco, número da agência e número de conta;
- Foto de perfil que será utilizada por todo aplicativo.

RF002 – Realizar login no sistema

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao prestador de serviço realizar login no sistema utilizando suas credenciais de e-mail e senha cadastrados no aplicativo.

RF003 – Gerenciar dados cadastrais do prestador

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao prestador de serviço inserir, editar e deletar seus dados pessoais, de endereço, financeiro.

Alguns dados possuem restrições tais como:

- Nome da empresa, Razão Social e CNPJ não poderão ser alterados ou deletados.

RF004 – Gerenciar serviços

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao prestador de serviço inserir, alterar e deletar um serviço da sua lista. Um prestador deverá ter obrigatoriamente pelo menos um serviço cadastrado no aplicativo.

Os dados a serem armazenados referentes a cada serviço são:

- Dias e horários de atendimento;
- Descrição do serviço;
- Tempo estimado de duração;
- Valor.

RF005 – Gerenciar meios de pagamento

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao prestador de serviço a inserir as formas de pagamento com as quais poderá receber seus pagamentos na plataforma.

As formas de pagamento que o prestador poderá cadastrar: Cartão de crédito, débito e dinheiro.

Todos os prestadores deverão obrigatoriamente ter a opção de receber pagamento pelo aplicativo.

RF006 – Consultar histórico do prestador

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao prestador visualizar seu histórico de serviços realizados pelo aplicativo, utilizando os seguintes filtros para pesquisa: por tipo específico e período ao longo do tempo.

As informações que devem estar disponíveis referente a cada serviço são:

- Data e horário do atendimento;
- Lista com os serviços realizados no atendimento;
- Modelo e cor do veículo;
- Valor total do pedido;
- Forma de pagamento;
- Valor e comentário do cliente referente ao atendimento.

RF007 - Visualizar notificações

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao prestador visualizar suas notificações, que serão enviadas tanto pela plataforma, quanto pelos seus clientes (quando forem relacionadas ao processo de um atendimento/pedido).

RF008 – Realizar atendimento

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao prestador visualizar, iniciar e finalizar um atendimento, alterando a indicação de status para o cliente de acordo com o progresso.

Os status referentes a um atendimento serão os seguintes (para os casos onde o serviço for realizado no espaço da própria empresa):

- Veículo no lava-car;
- Realizando serviço no veículo;
- Serviço concluído;
- Veículo entregue ao cliente.

RF009 – Solicitações Recebidas

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao prestador aceitar ou recusar uma solicitação de serviço do possível cliente.

As informações que devem estar disponíveis:

- Data e horário do atendimento;
- Placa do veículo;
- Modelo e cor do veículo;
- Valor;
- Tamanho (pequeno/médio/grande);
- Descrição.

Requisitos do sistema

RNF001 – Enviar e-mail com uma senha gerada automaticamente pelo sistema.

Quando o moderador, informar seu e-mail solicitando a recuperação de senha, a própria aplicação deve gerar uma senha e encaminhar para o usuário para recuperar seu acesso.

RNF002 – Senhas devem estar criptografadas.

Todas as senhas devem estar criptografadas no banco de dados, utilizando o padrão MD5.

APÊNDICE C - LISTA DE REQUISITOS SISTEMA WEB MODERADOR

Este documento tem por objetivo listar todos os requisitos definidos para a construção do aplicativo Easy Wash.

Descrição do sistema

O principal objetivo do sistema é oferecer meios gerenciais para os proprietários do aplicativo. As principais funcionalidades são:

- Moderador poderá visualizar informações referentes aos clientes e prestadores;
- Aprovar cadastro de novos prestadores na plataforma;
- Responder mensagens de suporte enviadas tanto por clientes quanto prestadores.

RF001 – Realizar Login

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao moderador realizar login com seu e-mail e senha cadastrados no sistema.

RF002 – Gerenciar Moderadores

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir o gerenciamento de moderadores no sistema. O usuário poderá cadastrar, editar e deletar moderadores.

Os dados para cadastro de moderadores são os seguintes: Nome completo, e-mail e senha.

RF003 – Gerenciar clientes

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao moderador pesquisar por um cliente pelo nome e posteriormente visualizar e editar suas informações.

As seguintes informações de cada cliente devem ser apresentadas ao moderador:

- Todos os dados pessoais;
- Todos os dados de endereço.

RF004 – Gerenciar prestadores

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao moderador pesquisar por um prestador pelo nome e posteriormente visualizar e editar suas informações.

As seguintes informações de cada cliente devem ser apresentadas ao moderador:

- Todos os dados cadastrais da empresa;
- Todos os dados de endereço;
- Lista de serviços ofertados pela empresa.

RF005 – Analisar solicitações de prestadores de serviço que desejam utilizar a plataforma.

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao moderador visualizar um pedido de cadastro de um prestador a plataforma para posterior aprovação ou não.

As informações a serem visualizadas pelo moderador são:

- Todos os dados pessoais da empresa;
- Todos os dados referentes a localização da empresa;
- Todos os dados referentes aos serviços a serem ofertados pela empresa;
- Todas as informações bancárias da empresa: Dados da conta e formas de pagamento.

RF006 – Gerenciar mensagens recebidas

Prioridade: Essencial Importante Desejável

Esta funcionalidade deverá permitir ao moderador visualizar e responder mensagens enviadas pelos usuários do aplicativo, tanto mensagens dos clientes quanto dos prestadores.

As informações referentes a mensagem que devem ser mostradas na aplicação são as seguintes:

- Dados do remetente: Nome, CPF ou CNPJ, e-mail e telefone;
- Data do envio da mensagem;
- Assunto e conteúdo da mensagem.

A página deverá conter um campo para o moderador responder a mensagem enviada.

Requisitos do sistema**RNF001 – Enviar e-mail com uma senha gerada automaticamente pelo sistema.**

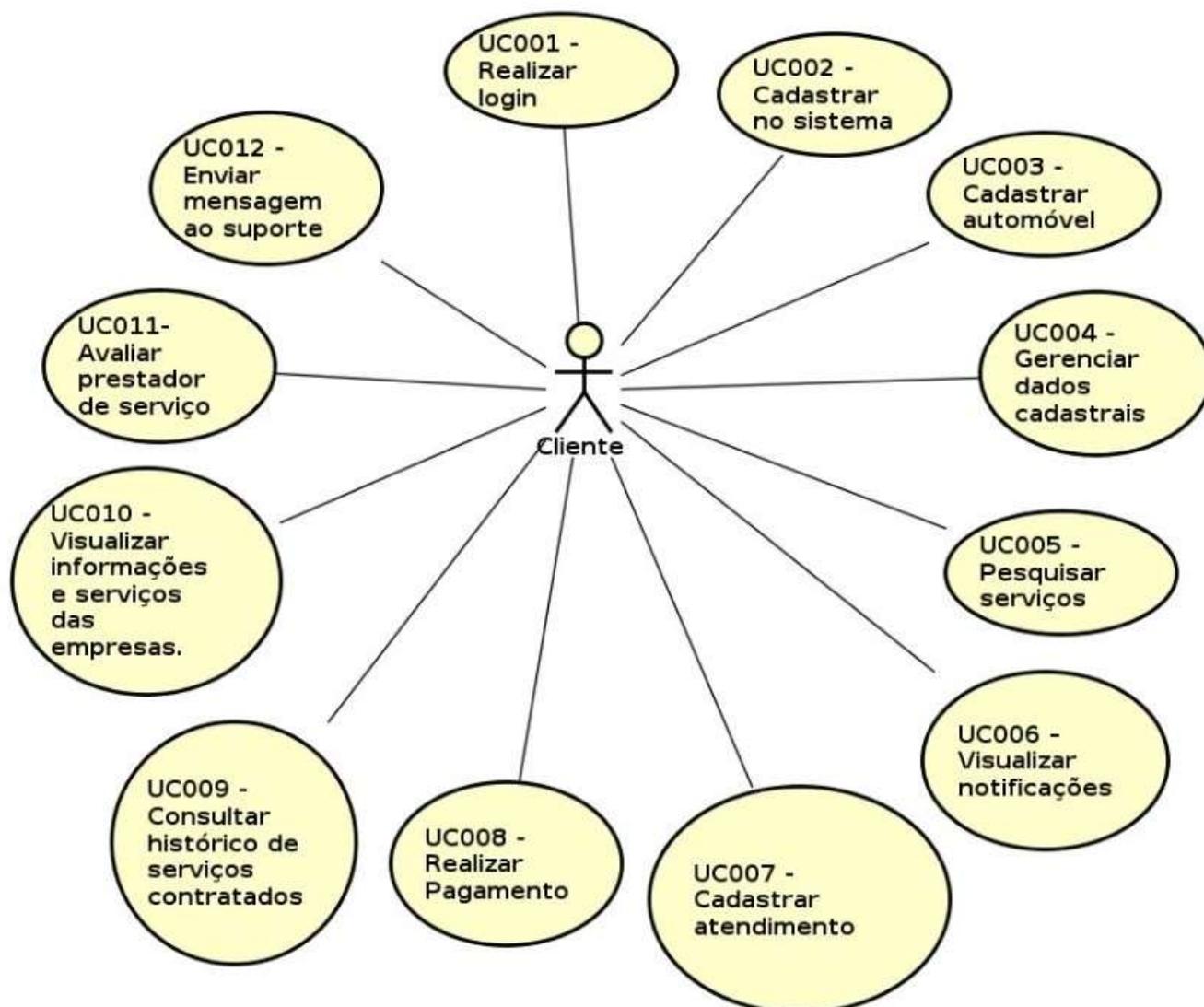
Quando o moderador, informar seu e-mail solicitando a recuperação de senha, a própria aplicação deve gerar uma senha e encaminhar para o usuário para o mesmo possa recuperar seu acesso.

RNF002 – Senhas devem estar criptografadas.

Todas as senhas devem estar criptografadas no banco de dados, utilizando o padrão MD5.

APÊNDICE D - DIAGRAMAS DE CASO USO CLIENTE

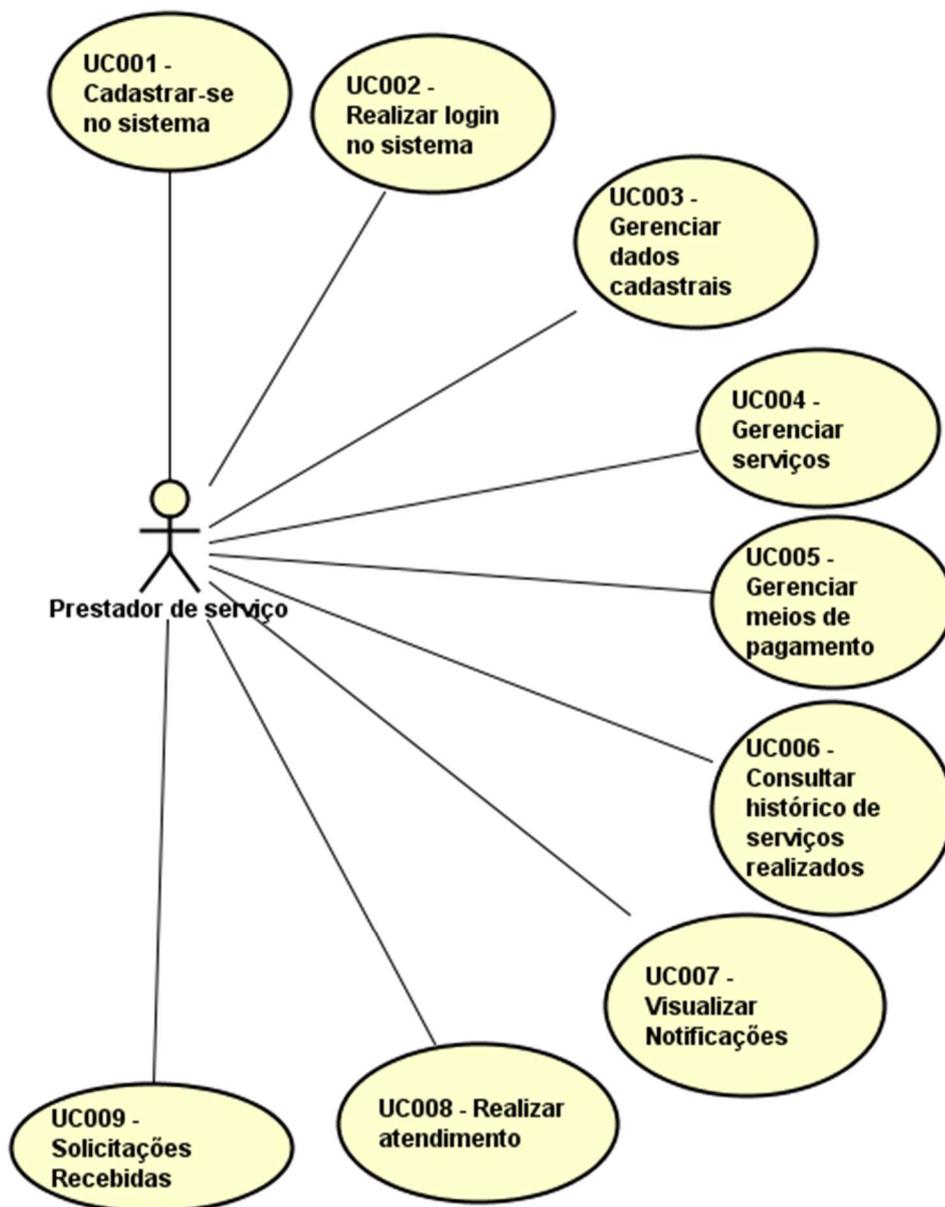
FIGURA 63 - DIAGRAMA DE CASOS DE USO - CLIENTE



Fonte: Os Autores (2020)

APÊNDICE E - DIAGRAMAS DE CASO USO PRESTADOR

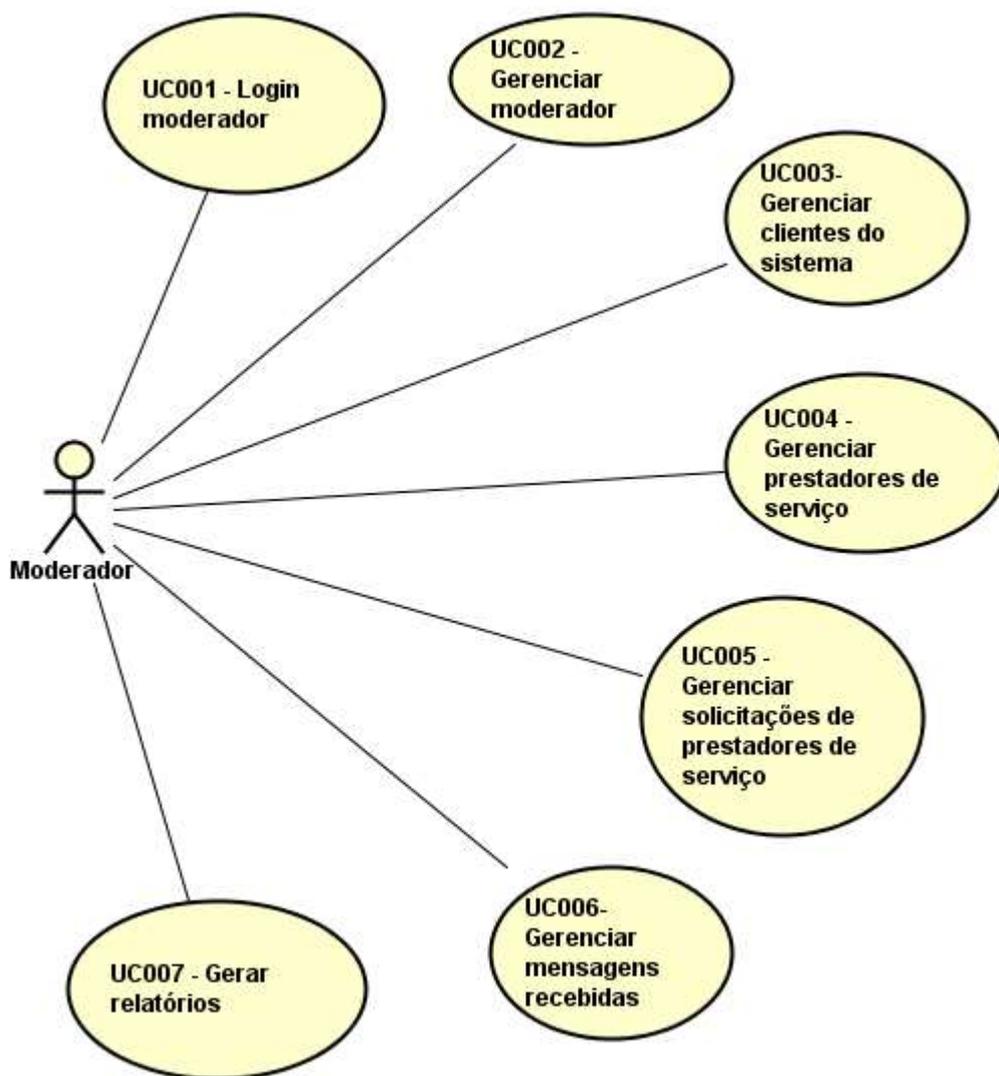
FIGURA 64 - DIAGRAMA DE CASOS DE USO - PRESTADOR



Fonte: Os Autores (2020)

APÊNDICE F - DIAGRAMAS DE CASO USO MODERADOR

FIGURA 65 - DIAGRAMA DE CASOS DE USO - MODERADOR



Fonte: Os Autores (2020)

APÊNDICE G - ESPECIFICAÇÃO APLICATIVO CLIENTE

UC001 – Realizar login

Descrição

Neste caso de uso o cliente poderá realizar login para acessar o aplicativo.

Data View

DV01 – Tela inicial

FIGURA 66 - DV01 - TELA INICIAL APLICATIVO CLIENTE



FONTE: Os autores (2019).

DV02 – Login

FIGURA 67 - DV02 – TELA DE LOGIN



FONTE: Os autores (2020).

DV03 – Recuperar senha

FIGURA 68 - DV03 – TELA DE SOLICITAÇÃO PARA RECUPERAÇÃO DE SENHA



FONTE: Os autores (2020).

Ator Primário

Cliente

Fluxo de Eventos Principal

1. O sistema apresenta DV01 – Tela de início.
2. O cliente clica no link já possuo uma conta.
3. O sistema chama DV02 – Tela de login.
4. O cliente digita seu e-mail no campo e-mail.
5. O cliente digita a senha no campo senha.
6. O cliente clica no botão entrar. (A1) (A2) (A3) (A4)
7. O sistema realiza a verificação de login e senha do cliente no banco de dados.
8. O sistema chama o caso de Uso UC05 - Pesquisar serviço.
9. O caso de uso é encerrado.

Fluxos Alternativos

A1 – O cliente clica no ícone do Facebook.

1. O sistema realiza uma requisição para a API de login do Facebook
2. O sistema recupera os dados da API e envia-os para o UC02 – Cadastrar cliente.
3. O sistema chama o caso de uso UC02 – Cadastrar cliente.

A2 – O cliente clica no ícone do Gmail.

1. O sistema envia uma requisição para a API de login ao Gmail.
2. O sistema recupera os dados e envia-os para o UC02 – Cadastrar cliente.
3. O sistema chama o caso de uso UC02 – Cadastrar cliente.

A3 – O cliente clica no link cadastrar-se.

1. O sistema chama o caso de uso UC02 – Cadastrar cliente.

A4 – O cliente clica em esqueci minha senha.

1. O sistema chama a tela DV03 – Recuperar senha
2. O cliente digita o seu e-mail no campo e-mail.
3. O cliente clica no botão recuperar.
4. O sistema envia as instruções da recuperação de senha para o e-mail informado. (E1) (RN1)
5. O sistema emite a mensagem “as instruções de recuperação de senha foram enviadas para seu e-mail.

Fluxos de Exceção

E1 – E-mail não cadastrado.

1. O sistema persiste o e-mail.
2. O sistema emite mensagem “E-mail não cadastrado”.

Regras de negócio

RN1. O corpo do e-mail deve conter uma nova senha para o usuário ser capaz de acessar o sistema. Esta senha deve conter 8 caracteres.

UC002 – Cadastrar- no sistema

Descrição

Neste caso de uso o cliente poderá cadastrar-se no aplicativo.

Data View

DV01 – Cadastrar dados pessoais

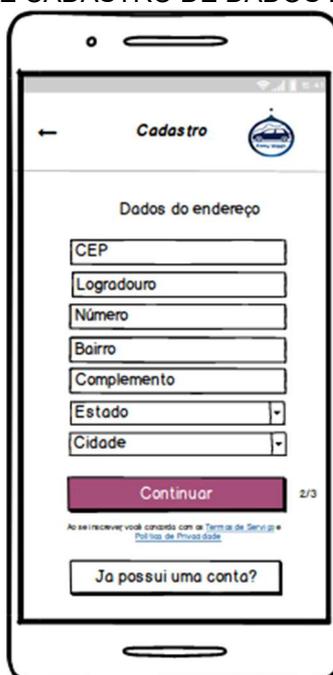
FIGURA 69 - DV01 - TELA DE CADASTRO DE DADOS PESSOAIS DO CLIENTE

A imagem mostra a interface de usuário de um aplicativo em um smartphone. O título da tela é "Cadastro" com um ícone de seta para trás à esquerda e um logotipo à direita. Abaixo do título, há o subtítulo "Dados pessoais". A tela contém os seguintes campos de entrada: "Digite seu nome", "Digite seu CPF", "Data de Nascimento" (com ícone de calendário), "Digite seu e-mail", "Digite sua senha", "Digite seu telefone". Abaixo dos campos, há um botão "Continuar" em um retângulo amarelo. Na base da tela, há um link de texto: "Ao se inscrever, você concorda com os Termos de Serviço e Política de Privacidade" e um botão "Já possui uma conta?".

FONTE: Os Autores (2020).

DV02 – Cadastrar endereço

FIGURA 70 - DV02 - TELA DE CADASTRO DE DADOS DO ENDEREÇO DO CLIENTE



The screenshot shows a mobile application interface for address registration. At the top, there is a back arrow and the word "Cadastro" next to a logo. Below this, the title "Dados do endereço" is centered. The form consists of several input fields: "CEP", "Logradouro", "Número", "Bairro", "Complemento", "Estado" (a dropdown menu), and "Cidade" (a dropdown menu). A purple "Continuar" button is positioned below the fields, with a "2/3" indicator to its right. At the bottom, there is a link for "Termos de Serviço e Política de Privacidade" and a button labeled "Ja possui uma conta?".

FONTE: Os Autores (2020).

DV03 – Carregar foto de perfil.

FIGURA 71 - DV03 - TELA DE UPLOAD DE FOTO DE PERFIL



The screenshot shows a mobile application interface for profile photo upload. At the top, there is a back arrow and the word "Cadastro" next to a logo. Below this, the title "Foto de perfil" is centered. A large grey circle placeholder contains the text "Insira uma imagem". Below the circle are two buttons: "Fotografar" with a camera icon and "Carregar" with an upload icon. A purple "Continuar" button is located at the bottom of the screen.

FONTE: Os autores (2020).

Ator Primário
Cliente.

Fluxo de Eventos Principal

1. O sistema apresenta DV01 – Tela de Cadastro de Dados Pessoais do Cliente.
2. O cliente preenche o campo nome.
3. O cliente preenche o campo CPF.
4. O cliente preenche o campo e-mail.
5. O cliente preenche o campo senha.
6. O cliente preenche o campo telefone.
7. O cliente clica no botão continuar. (A1) (E1) (E2) (E3) (E4)
8. O sistema chama a tela DV02 – Tela de Cadastro de dados de endereço do cliente.
9. O cliente preenche o campo CEP. (RN5)
10. O sistema preenche o campo logradouro, estado, cidade, bairro. (RN6) (RN7)
11. O cliente preenche o campo número.
12. O cliente clica no botão continuar.
13. O sistema chama a tela DV03 – Cadastrar foto de perfil.
14. O cliente clica no botão. (A2)
15. O cliente seleciona uma imagem.
16. O cliente clica no botão continuar.
17. O sistema chama o caso de uso UC003 – Cadastrar Automóvel.
18. O caso de uso é encerrado.

Fluxos Alternativos

A1 – O cliente clica no botão Já possui uma conta?

1. O sistema chama o caso de uso UC01 – DV02 – Login.

A2 – O cliente clica em fotografar

1. O sistema abre a câmera do cliente.
2. O cliente tira a foto.
3. O cliente clica no botão continuar.
4. O sistema chama o UC03 – DV01 – Tela de cadastro de veículo.
5. O caso de uso é encerrado.

Fluxos de exceção

E1 – Campo nome completo do cliente inválido.

1. O sistema valida o campo nome completo do cliente do formulário. (RN1)
2. O sistema emite a mensagem “Nome completo inválido.”.

3. O sistema retorna para o início do fluxo principal.

E2 – Campo CPF inválido.

1. O sistema valida os dados do CPF do cliente. (RN2)

2. O sistema emite a mensagem “CPF inválido”.

3. O sistema retorna para o início do fluxo principal.

E3 – Campo telefone inválido.

1. O sistema valida os dados do telefone do cliente. (RN3)

2. O sistema emite a mensagem “Telefone inválido”.

3. O sistema retorna para o início do fluxo principal.

E4 – Campo e-mail inválido.

1. O sistema valida os dados do telefone do cliente. (RN4)

2. O sistema emite a mensagem “E-mail inválido”.

3. O sistema retorna para o início do fluxo principal.

E5 – Campo CEP inválido.

1. O sistema consiste dos dados de número do endereço do cliente. (RN5)

2. O sistema emite a mensagem “CEP inválido”.

Regras de negócio

RN1. O campo nome deverá possuir entre 8 e 30 caracteres.

RN2. Utilizar a rotina padrão de consistência de DV do CPF da Receita Federal.

RN3. O campo telefone deverá ter o formato brasileiro (XX) XXXX-XXXX.

RN4. O campo e-mail deverá ser no formato xxxx@xxxx.xx

RN5. O campo CEP deverá usar o VIACEP (<https://viacep.com.br>).

RN6. A lista de cidades deverá ser auto preenchida com todas as cidades do estado selecionado.

RN7. A lista de estados deverá conter todos os estados brasileiros.

UC003 – Cadastrar Automóvel.

Descrição

Este caso de uso deverá permitir ao cliente cadastrar um ou mais automóveis no aplicativo.

Data View

DV01 – Cadastro e edição de automóvel

FIGURA 72 - DV01 - TELA DE CADASTRO E EDIÇÃO DE AUTOMÓVEL



FONTE: Os autores (2020).

Ator Primário

Cliente.

Fluxo de Eventos Principal

1. O sistema apresenta DV01 – Tela de cadastro e edição de automóvel.
2. O sistema carrega lista de tipos de automóveis cadastrados no sistema. (RN1)
3. O cliente seleciona uma opção do combobox tipo do veículo.
4. O cliente preenche o campo modelo.
5. O cliente preenche o campo cor.
6. O cliente preenche o campo placa. (E1)
7. O cliente clica no botão continuar.
8. O sistema envia as informações digitadas para o banco de dados.
9. O sistema emite a mensagem “Veículo cadastrado com sucesso!”.
10. O caso de uso é encerrado.

Fluxos de Exceção

E1 – Placa inválida

1. O sistema valida os dados da placa do automóvel. (RN2)

2. O sistema emite a mensagem “Placa inválida”.
3. O sistema retorna para o início do fluxo principal.

Regras de negócio

RN1. O sistema deverá possuir uma lista de tipos de automóveis cadastrados no sistema.

RN2. Utilizar a rotina padrão "[A-Z]{3}-[0-9]{4}" para validação de placa do veículo.

UC004– Gerenciar dados cadastrais

Descrição

Neste caso de uso o cliente poderá gerenciar seus dados cadastrais.

Data View

DV01 – Tela inicial de gerenciamento de informações

FIGURA 73 - DV01 - TELA INICIAL DE GERENCIAMENTO DE INFORMAÇÕES



FONTE: Os autores (2020).

DV02 – Tela de gerenciamento de automóveis

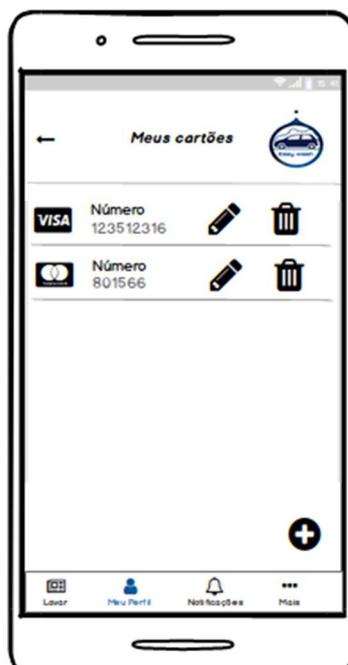
FIGURA 74 - DV02 - TELA DE GERENCIAMENTO DE AUTOMÓVEIS



FONTE: Os autores (2020).

DV03 – Gerenciamento de dados cadastrais / Meus cartões.

FIGURA 75 - DV03 - TELA DE GERENCIAMENTO DE CARTÕES



FONTE: Os autores (2020).

Ator Primário

Cliente

Fluxo de Eventos Principal

1. O sistema apresenta DV01 - Tela inicial de gerenciamento de informações.
2. O cliente clica no ícone “Meus automóveis”. (A1)(A2)(A3)
3. O sistema chama a tela DV02 – Tela de Gerenciamento de automóveis.
4. O sistema realiza uma busca dos carros cadastrados do cliente no banco de dados.
5. O cliente clica no ícone “+”. (A4)(A5)
6. O sistema realiza a chamada para o caso de uso UC003 – Cadastrar Automóvel.
7. O caso de uso é encerrado.

Fluxos alternativos

A1 – O cliente clica no ícone da pessoa.

1. O sistema chama o UC002 com os dados preenchidos.
2. O caso de uso é encerrado.

A2 – O cliente clica no ícone da foto.

1. O sistema chama o UC002 na tela DV03 - Carregar foto de perfil.
2. O caso de uso é encerrado.

A3 – O cliente clica no ícone do cartão.

1. O sistema chama o DV03 - Tela de Gerenciamento de Cartões.
2. O cliente clica no ícone “+”. (A6)(A7)
3. O sistema chama o UC10 – DV03 – Cadastrar cartão.
4. O caso de uso é encerrado.

A4 – O cliente clica no ícone de edição (Automóvel).

1. O sistema chama o UC03 – Tela de Cadastramento e Edição de veículos com os dados do veículo preenchidos.
2. O cliente atualiza os dados do veículo.
3. O sistema habilita o botão salvar.
4. O cliente clica no botão salvar.
5. O sistema atualiza as informações no banco de dados.
6. O sistema emite a mensagem “Dados do veículo atualizados com sucesso! ”.
7. O caso de uso é encerrado.

A5 – O cliente clica no ícone de excluir (Automóvel).

1. O sistema deleta o veículo do banco de dados.
2. O sistema emite a mensagem “Veículo deletado com sucesso! ”.
3. O caso de uso é encerrado.

A6 – O cliente clica no ícone de editar (Cartão).

1. O sistema chama o caso de uso UC10 – DV03 – Cadastrar cartão com os dados preenchidos.
2. O caso de uso é encerrado.

A7 – O cliente clica no ícone de excluir (Cartão).

1. O sistema deleta o cartão do banco de dado.
2. O sistema emite a mensagem “Cartão deletado com sucesso! ”.
3. O caso de uso é encerrado.

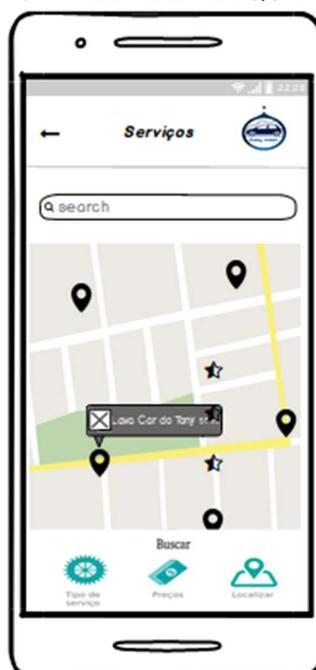
UC005 – Pesquisar serviços

Descrição

Neste caso de uso o cliente poderá fazer pesquisas por serviços e solicitá-los.

DV01 – Tela de Pesquisa de serviços

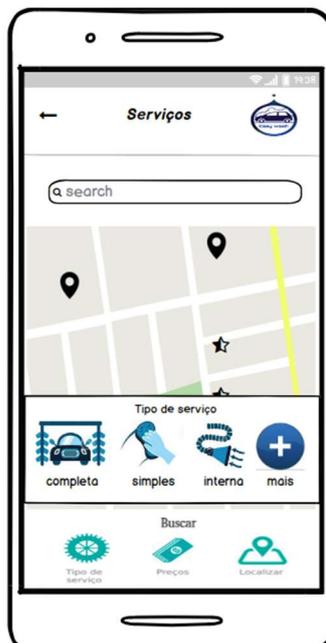
FIGURA 76 - DV01 - TELA DE PESQUISA DE SERVIÇOS



FONTE: Os autores (2020).

DV02 – Pesquisar serviços por filtro

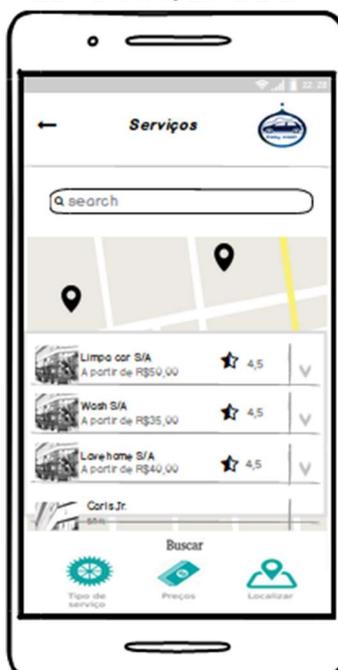
FIGURA 77 - DV02 - TELA DE PESQUISA POR FILTRO



FONTE: Os autores (2020).

DV03 – Listagem de empresas por filtro

FIGURA 78 - DV03 - TELA LISTAGEM DE EMPRESAS POR FILTRO



FONTE: Os autores (2020).

Ator Primário

Cliente

Fluxo de Eventos Principal

1. O sistema apresenta a tela DV01 – Tela de pesquisa de serviços.
2. O cliente clica no em tipo de serviços. (A1)(A2)(A3)
3. O sistema apresenta a tela DV02 – Tela de pesquisa de serviços por filtro.
4. O sistema apresenta lista de tipos de serviços.
5. O cliente seleciona um tipo de serviço.
6. O sistema realiza um filtro e retorna uma lista de empresas baseado no filtro.
7. O sistema chama a tela DV03 - Listagem de empresas por filtro.
8. O cliente clica em uma das empresas da lista.
9. O sistema chama o caso de uso UC11 - Visualizar informações e serviços das empresas.
10. O caso de uso é encerrado.

Fluxos alternativos

A1 – O cliente clica no ícone de preço.

1. O sistema realiza um filtro e retorna uma lista de empresas baseado no filtro selecionado
2. O sistema chama DV03 - Listagem de empresas por filtro.
3. O cliente clica em uma das empresas da lista.
4. O sistema chama o caso de uso UC11 - Visualizar informações e serviços das empresas.
5. O caso de uso é encerrado.

A2 – O cliente clica no localizar

1. O cliente clica no ícone do mapa.
2. O sistema recupera uma lista de empresas próximas ao local.
3. O cliente clica em uma das empresas da lista.
4. O sistema chama o caso de uso UC11 - Visualizar informações e serviços das empresas.
5. O caso de uso é encerrado.

A3 – O cliente digita o nome do prestador no campo pesquisar.

1. O sistema valida o nome digitado.
2. O sistema encontra o prestador. (E1)
3. O sistema localiza a empresa no mapa.

4. O cliente clica em uma empresa da lista.
5. O sistema chama o caso de uso UC11 - Visualizar informações e serviços das empresas.
6. O caso de uso é encerrado.

Fluxos de exceção

E1 – Prestador não encontrado.

1. O sistema emite a mensagem “Nenhuma empresa localizada.”.

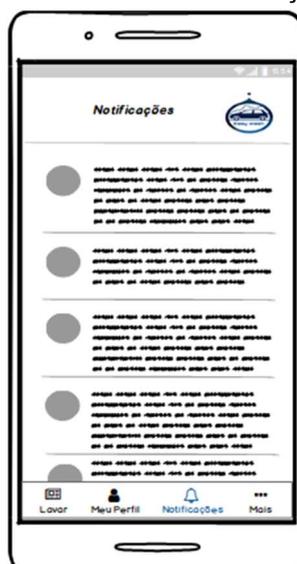
UC006 – Visualizar notificações

Descrição

Neste caso de uso o cliente poderá visualizar suas notificações.

DV01 – Visualizar notificações

FIGURA 79 - DV01 - TELA DE NOTIFICAÇÕES DO CLIENTE



FONTE: Os autores (2020).

Ator Primário

Cliente

Fluxo de Eventos Principal

1. O sistema apresenta DV01 Tela de Visualização de notificações.
2. O sistema realiza uma busca no banco de dados referente as notificações

recebidas de um cliente (RN1)(RN2)

3. O caso de uso é encerrado.

Regras de negócio

RN1. A lista de notificações deve mostrar as últimas 10 notificações.

RN2. O sistema carrega mais notificações ao identificar que o cliente está na décima.

UC007 – Cadastrar Atendimento

Descrição

Neste caso de uso o cliente poderá selecionar o tipo de serviço que deseja contratar.

DV01 – Tela de contratação de serviço.

FIGURA 80 - DV01 - TELA DE CONTRATAR SERVIÇO RESERVA HORÁRIO



FONTE: Os autores (2020).

DV02 – Tela de seleção de serviços da empresa

FIGURA 81 - DV02 - TELA DE SELEÇÃO DE SERVIÇOS DA EMPRESA



FONTE: Os autores (2020).

Ator Primário

Cliente

Fluxo de Eventos Principal

1. O sistema apresenta DV01 – Tela de contratação de serviços.
2. O cliente seleciona a data desejada. (RN1)
3. O cliente preenche o campo horário.
4. O cliente seleciona o tipo de solicitação (Retirada e entrega do veículo).
5. O cliente preenche o campo observações.
6. O cliente seleciona a opção usar endereço cadastrado (A1)
7. O cliente clica no botão continuar.
8. O sistema chama a DV02 – Tela de seleção de serviços da empresa.
9. O cliente seleciona os serviços que deseja contratar. (RN2)
10. O cliente clica no botão continuar.
11. O sistema chama o caso de uso UC008 – Selecionar opção de pagamento.
12. O caso de uso é encerrado.

Fluxos alternativos

A1 – O cliente seleciona a opção enviar localização atual.

1. O sistema pega a localização atual com base no GPS do celular do cliente.
2. O sistema retorna para o início do fluxo principal.

Regras de negócio

RN1. Data não pode ser menor que a atual.

RN2. O preço total e o tempo de serviço devem ser calculados com base na seleção dos usuários.

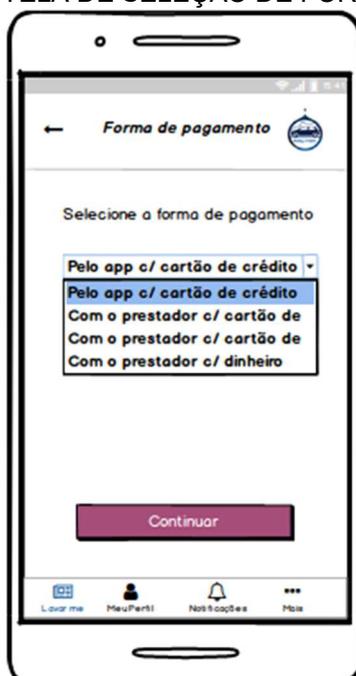
UC008 – Realizar pagamento

Descrição

Neste caso de uso o cliente poderá selecionar a forma de pagamento e contratar o serviço.

DV01 – Realizar pagamento

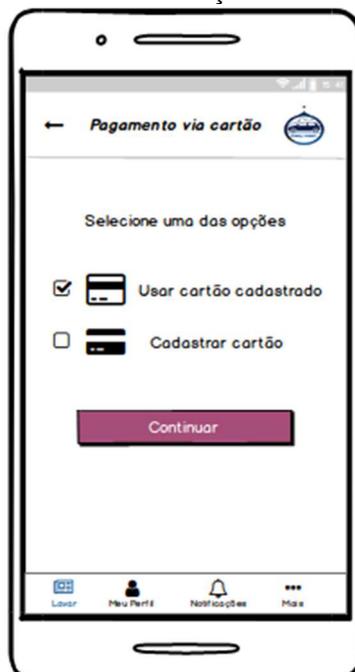
FIGURA 82 - DV01 - TELA DE SELEÇÃO DE FORMA DE PAGAMENTO



FONTE: Os autores (2019).

DV02 – Tela de seleção de cartão de crédito.

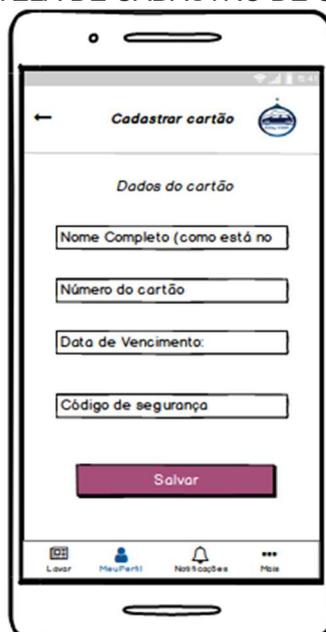
FIGURA 83 - DV02 – TELA DE SELEÇÃO DE CARTÃO DE CRÉDITO



FONTE: Os autores (2019).

DV03 – Tela de cadastro de cartão de crédito

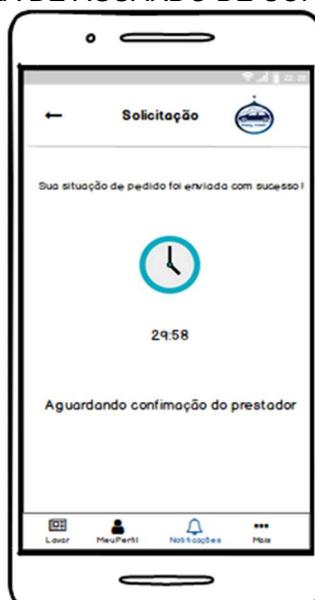
FIGURA 84 - DV03 - TELA DE CADASTRO DE CARTÃO DE CRÉDITO



FONTE: Os autores (2020).

DV04 – Tela de aguardo de confirmação de serviço

FIGURA 85 - DV04 - TELA DE AGUARDO DE CONFIRMAÇÃO DE SERVIÇO



FONTE: Os autores (2020).

Ator Primário

Cliente

Fluxo de Eventos Principal

1. O sistema apresenta DV01 - Tela de Seleção de Cartão de Crédito.
2. O cliente seleciona a seguinte opção de pagamento: “pelo app c/ cartão de crédito”.
3. O sistema chama a DV02 - Pelo app c/ cartão de crédito.
4. O cliente clica no ícone usar cartão cadastrado.
5. O sistema mostra o campo código de segurança do cartão.
6. O cliente informa o código de verificação do cartão.
7. O cliente clica no botão confirmar (A1)(A2)(A3) (A4)(E2)
8. O sistema chama a tela DV04 – Tela de aguardo de confirmação de serviço.
(E3)(RN4)
9. O caso de uso é encerrado.

Fluxos alternativos

A1 – O cliente seleciona a seguinte opção de pagamento: “com o prestador c/ cartão de crédito”.

1. O sistema chama a tela DV04 - Tela de aguardo de confirmação de serviço.
2. O sistema retorna ao fluxo principal.

A2 – O cliente seleciona a seguinte opção de pagamento: “com o prestador c/ cartão de débito”.

1. O sistema chama a tela DV04 - Tela de aguardo de confirmação de serviço.
(E3)(RN4)
2. O sistema retorna ao fluxo principal.

A3 – O cliente seleciona a seguinte opção de pagamento: “com o prestador c/ dinheiro”.

1. O sistema chama a tela DV04 - Tela de aguardo de confirmação de serviço.
2. O sistema retorna ao fluxo principal.

A4 – O cliente clica na opção cadastrar cartão.

1. O sistema chama DV03 – Cadastrar cartão.
2. O cliente preenche o campo nome. (RN1)
3. O cliente preenche o número do cartão. (E1)
4. O cliente preenche o campo data. (RN2)
5. O cliente preenche o campo código de segurança. (RN3)
6. O cliente clica no botão salvar.
7. O sistema retorna ao fluxo principal

Fluxos de Exceção

E1 – Número do cartão inválido.

1. O sistema persiste o número do cartão de crédito.
2. O sistema emite mensagem “Número do cartão é inválido “.

E2– Código de segurança do cartão inválido.

1. O sistema persiste o código de verificação do cartão de crédito.
2. O sistema emite a mensagem “O código de segurança do cartão é inválido “.

E3 – Prestador não pode atender no horário informado pelo cliente.

1. O prestador envia uma mensagem com possíveis horários disponíveis para atendimento.
2. O sistema envia uma notificação da mensagem para o cliente.

3. O cliente clica na notificação.
4. O sistema redireciona o cliente para a página de contratação do serviço.
5. O cliente seleciona uma nova data e horário.
6. O caso de uso é encerrado.

Regras de negócio

RN1. Nome deve ser igual o nome que está no cartão de crédito.

RN2. Data de agendamento deve ser maior que a data atual.

RN3. O código de segurança não é armazenado no sistema e deve ser solicitado toda vez no ato de contratação de serviço.

RN4. O pagamento só é finalizado caso o prestador venha a confirmar a solicitação

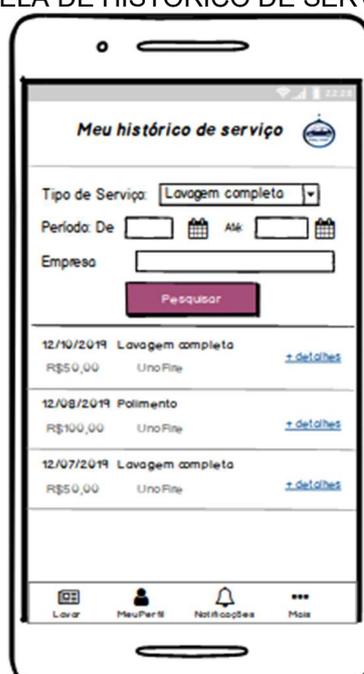
UC009 – Consultar Histórico de serviços contratados

Descrição

Neste caso de uso o cliente poderá visualizar o seu histórico de serviços contratados.

DV01 – Tela de Histórico de serviços contratados

FIGURA 86 - DV01 - TELA DE HISTÓRICO DE SERVIÇOS CONTRATADOS



FONTE: Os autores (2020).

DV02 – Tela Histórico detalhado de serviço contratado.

FIGURA 87 - DV02 - TELA HISTÓRICO DETALHADO DE SERVIÇO CONTRATADO



FONTE: Os autores (2020).

Ator Primário

Cliente

Fluxo de Eventos Principal

1. O sistema apresenta a tela DV01 - Tela de Histórico de Serviços contratados.
2. O cliente preenche os campos com as opções de filtro que deseja aplicar.
3. O cliente clica no botão pesquisar.
4. O sistema realiza uma busca no banco de dados com base nos filtros informados pelo cliente.
5. O cliente clica no link "+ detalhes".
6. O sistema realiza uma busca no banco de dados referente ao serviço selecionado pelo cliente.
7. O sistema chama a tela DV02 – Histórico detalhado de serviço prestado.
8. O caso de uso é encerrado.

UC010 – Visualizar informações e serviços das empresas.

Descrição

Neste caso de uso o cliente pode visualizar as informações das empresas no aplicativo, como horários, avaliações, e seus respectivos serviços ofertados.

DV01 – Tela de perfil do prestador de serviços

FIGURA 88 - DV01 - TELA DE PERFIL DO PRESTADOR DE SERVIÇOS



FONTE: Os autores (2020).

Ator Primário

Cliente

Fluxo de Eventos Principal

1. O sistema apresenta DV01 – Tela de perfil do prestador de serviços.
2. O sistema realiza uma busca no banco de dados com informações.
3. O cliente clica no botão contratar.
4. O sistema chama o UC007 – Cadastrar Atendimento.
5. O caso de uso é encerrado.

UC011 – Avaliar prestador de serviço.

Descrição

Neste caso de uso, após a realização de uma lavagem ou outro serviço ser o cliente poderá avaliar o prestador.

Data View

DV01 – Tela de Avaliação de Serviço

FIGURA 89 - DV01 - TELA DE AVALIAÇÃO DE SERVIÇO



FONTE: Os autores (2020).

Ator Primário

Cliente

Fluxo de Eventos Principal

1. O sistema apresenta a tela DV01 – Tela de Avaliação de serviço.
2. O cliente seleciona uma nota de avaliação de 1 a 5 estrelas para o prestador de

serviço. (A1)(R1)

3. O cliente escreve um comentário referente ao serviço prestado.
4. O cliente clica no botão Enviar. (E1)
5. O sistema envia as informações da avaliação para o banco de dados.
6. O sistema exibe o toast com a seguinte mensagem: “Avaliação enviada com sucesso.
7. O sistema faz um redirecionamento para a página inicial do sistema.
8. O caso de uso é encerrado.

Fluxos Alternativos

A1 – O cliente clica no botão Voltar.

1. O sistema faz um redirecionamento para a página inicial do sistema.
2. O caso de uso é encerrado.

Fluxos de Exceção

E1 – Erro com operação no banco de dados.

1. O sistema emite a seguinte mensagem: ” Erro ao inserir avaliação no banco de dados. Por favor, tente novamente! ”.

Regras de negócio

RN1. O campo nota terá valor entre 0 e 5 estrelas.

UC012 – Enviar mensagem ao suporte

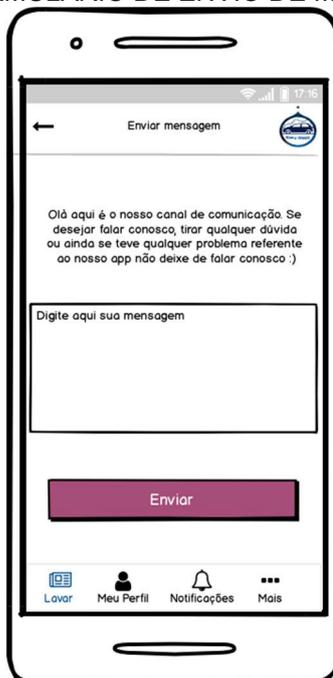
Descrição

Neste caso de uso o cliente poderá enviar uma mensagem ao suporte da plataforma Easy Wash.

Data View

DV01 – Formulário de Envio de mensagem ao suporte.

FIGURA 90 - DV01 - FORMULÁRIO DE ENVIO DE MENSAGEM AO SUPORTE



FONTE: Os autores (2020).

Ator Primário

Cliente

Fluxo de Eventos Principal

1. O sistema apresenta a tela DV01. - Tela de Envio de mensagem ao suporte. (A2)(A3)(A4)(A5)
2. O cliente preenche o input com a mensagem a ser enviada. (RN1)
3. O sistema habilita os campos Enviar e Cancelar.
4. O cliente clica no botão Enviar. (A1)(E1)
5. O sistema envia as informações da mensagem para o banco de dados.
6. O sistema emite um toast com a seguinte mensagem: "Mensagem enviada com sucesso".
7. O sistema faz um redirecionamento para a página inicial do aplicativo.
8. O caso de uso é encerrado.

Fluxos Alternativos

A1 – Cliente clica sobre o botão Voltar.

1. O sistema redireciona para a tela inicial do sistema.

2. O caso de uso é encerrado.

A2 – Cliente clica sobre a opção do menu Lavar.

1. O sistema mostra a aba com as opções do menu lavar.

A3 – Cliente clica sobre a opção do menu Meu Perfil.

1. O sistema mostra a aba com as opções do menu Meu Perfil.

A4 – Cliente clica sobre a opção do menu Notificações.

1. O sistema mostra a aba com as opções do menu Notificações.

A5 – Cliente clica sobre a opção do menu Mais.

1. O sistema mostra a aba com as opções do menu Mais.

Fluxos de Exceção

E1 – Erro ao salvar mensagem no banco de dados.

1. O sistema mostra a mensagem: “Erro ao enviar mensagem para o suporte técnico. Por favor, tente novamente! ”.

Regras de negócio

RN1. O campo mensagem não pode ser vazio. Deve conter entre 8 e 255 caracteres.

APÊNDICE H - ESPECIFICAÇÃO APLICATIVO PRESTADOR

UC001 – Cadastrar-se no sistema

Descrição

O prestador de serviços poderá se cadastrar no sistema para ter acesso plataforma de lavagens de carro. Após o cadastro um e-mail irá para o sistema web para aprovação.

Data View

DV01 – Tela de cadastro (Dados da empresa)

FIGURA 91 - DV01 - FORMULÁRIO DE CADASTRAR EMPRESA

A imagem mostra a tela de cadastro de uma empresa em um aplicativo móvel. O título da tela é "Cadastro" com um ícone de seta para trás e um logotipo de uma empresa. Abaixo do título, há o subtítulo "Dados da empresa" e duas opções de seleção: "Empresa" e "MEI". O formulário contém campos de entrada para: Nome, Razão social, CNPJ, Telefone, Email, Senha e Confirmar senha. Um botão "Continuar" em cor marrom está na base do formulário, acompanhado do indicador "1/5".

FONTE: Os autores (2020).

DV02 – Tela de cadastro endereço

FIGURA 92 - DV02 - FORMULÁRIO DE CADASTRAR ENDEREÇO DA EMPRESA

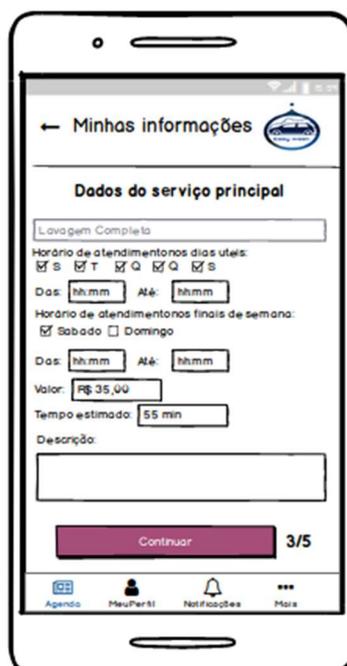


The screenshot shows a mobile application interface for registering a company address. At the top, there is a back arrow and the title 'Cadastro' next to a logo. Below this is a section titled 'Dados de endereço' containing several input fields: 'CEP', 'Rua', 'Número', 'Complemento', 'Estado' (a dropdown menu), and 'Cidade' (a dropdown menu). At the bottom of the form is a purple button labeled 'Continuar' with a progress indicator '2/5'.

FONTE: Os autores (2020).

DV03 – Tela de cadastro (Lista de Serviços)

FIGURA 93 - DV03 - FORMULÁRIO DE CADASTRAR SERVIÇO PRINCIPAL

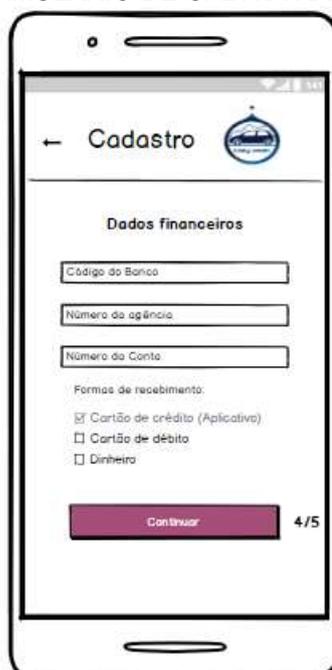


The screenshot shows a mobile application interface for registering a main service. At the top, there is a back arrow and the title 'Minhas informações' next to a logo. Below this is a section titled 'Dados do serviço principal' containing several input fields and checkboxes: a text field for 'Lavagem Completa', a section for 'Horário de atendimentos nos dias úteis' with checkboxes for S, T, Q, Q, S and time pickers 'Das: hh:mm Até: hh:mm', a section for 'Horário de atendimentos finais de semana' with checkboxes for Sabado and Domingo and time pickers 'Das: hh:mm Até: hh:mm', a 'Valor' field with 'R\$ 35,00', a 'Tempo estimado' field with '55 min', and a 'Descrição' text area. At the bottom of the form is a purple button labeled 'Continuar' with a progress indicator '3/5'. At the very bottom of the screen is a navigation bar with icons for 'Agenda', 'Meu Perfil', 'Notificações', and 'Mais'.

FONTE: Os autores (2020).

DV04– Tela de cadastro (Dados financeiros)

FIGURA 94 - DV04 - FORMULÁRIO DE CADASTRAR DADOS FINANCEIROS



The image shows a smartphone screen with the 'Cadastro' (Registration) app. The title bar at the top says 'Cadastro' with a back arrow on the left and a logo on the right. Below the title bar, the section is titled 'Dados financeiros' (Financial Data). There are three input fields: 'Código do Banco' (Bank Code), 'Número da agência' (Agency Number), and 'Número da Conta' (Account Number). Below these fields, there is a section for 'Formas de recebimento' (Payment Methods) with three checkboxes: 'Cartão de crédito (Aplicativo)' (Credit Card (App)) which is checked, 'Cartão de débito' (Debit Card), and 'Dinheiro' (Cash). At the bottom of the form is a purple 'Continuar' (Continue) button with '4/5' next to it.

FONTE: Os autores (2020)

DV05 – Tela de cadastro (FOTO)

FIGURA 95 - DV05 - FORMULÁRIO DE CADASTRAR FOTO

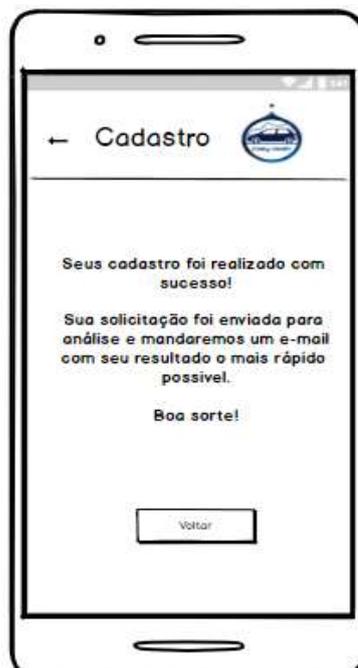


The image shows a smartphone screen with the 'Cadastro' (Registration) app. The title bar at the top says 'Cadastro' with a back arrow on the left and a logo on the right. Below the title bar, the text says 'Estamos finalizando... agora só falta sua foto de perfil' (We are finishing... now only your profile photo is missing). In the center, there is a large grey circle with the text 'Insira foto aqui' (Insert photo here). Below the circle are two buttons: 'Fotografar' (Take Photo) with a camera icon and 'Carregar' (Upload) with an upload icon. At the bottom of the form is a purple 'Concluir' (Finish) button with '5/5' next to it. At the very bottom, there is a small line of text: 'Ao concluir, você está concordando com os Termos de Serviço e Política de Privacidade' (By completing, you are agreeing with the Terms of Service and Privacy Policy).

FONTE: Os autores (2020)

DV06 – Tela de cadastro (Conclusão)

FIGURA 96 - DV06 - TELA DE FINALIZAÇÃO



FONTE: Os autores (2020).

Fluxo de Eventos Principal

1. O sistema apresenta a tela DV01.
2. O usuário preenche seus dados e clica no botão “continuar”. (E1)
3. O sistema carrega a tela DV02.
4. O usuário preenche seus dados e clica no botão “continuar”. (E1)
5. O sistema chama tela DV03.
6. O usuário preenche dados.
7. O usuário aperta o botão “continuar”.
8. O sistema chama tela DV04. (E1)
9. O usuário preenche seus dados e aperta o botão “continuar”.
10. O sistema chama a tela DV05.
11. O usuário tira uma foto e clica no botão “continuar”. (A1)
12. O sistema carrega a tela DV06.
13. O usuário clica no botão “voltar”.
14. O caso de uso está finalizado.

Fluxos Alternativos

- A1 – O usuário clicar no botão “carregar imagem”.

1. O sistema irá abrir um pop-up para o usuário selecionar a imagem desejada no seu celular.

Fluxos de Exceção

E1 – Campos vazios.

1. O sistema mostra mensagem “Existem campos vazios, favor preenchê-los para continuar”.

UC002 – Realizar Login no sistema

Descrição

Neste caso de uso o prestador poderá realizar acesso ao sistema utilizando suas credenciais de e-mail e senha previamente cadastrados.

Data View

DV01 - Tela de login

FIGURA 97 - DV01 - TELA DE LOGIN



FONTE: Os autores (2020).

DV02 – Tela de recuperação de senha

FIGURA 98 - DV01 - TELA RECUPERAR SENHA



FONTE: Os autores (2020)

Fluxo de Eventos Principal

1. O sistema apresenta a tela DV01.
2. Usuário preenche seus dados e clica no botão “entrar”. (A1) (A2) (E1) (E2)
3. O caso de uso está finalizado.

Fluxos Alternativos

A1 – O usuário seleciona a opção “Esqueci a senha!”

1. O sistema carrega a DV02.
2. O usuário preenche seus dados. (E1) (E3)
3. O usuário clica no botão “recuperar”.
4. O sistema enviará um e-mail com o passo a passo para recuperar a senha.
5. O caso de uso está finalizado.

A2- O usuário seleciona a opção “Não possui uma conta? Clique aqui!”

1. O sistema chama o UC015.

Fluxos de Exceção

E1 – Campos vazios.

1. O sistema emite a seguinte mensagem: “Existem campos vazios, favor preenche-los para continuar”.

E2 – Login ou senha incorretos.

1. O sistema emite a seguinte mensagem: “Login ou senha incorretos”.

E3 – E-mail inválido.

1. O sistema emite a seguinte mensagem: “E-mail inválido”.

Regras de negócio

Não há regras de negócio.

UC003 – Gerenciar dados cadastrais

Descrição

Neste caso de uso o prestador de serviço pode gerenciar seus dados cadastrais. Deve ser possível editar e incluir novos dados.

Data View

DV01 – Tela de minhas informações (Dados da empresa)

FIGURA 99 - DV01 - TELA DE MINHAS INFORMAÇÕES – DADOS DA EMPRESA

A imagem mostra a interface de usuário de um aplicativo móvel. No topo, há uma barra de navegação com o texto 'Minhas informações' e um ícone de perfil. Abaixo, o formulário 'Dados da empresa' contém campos para: Nome, Razão social, CNPJ, Telefone, E-mail, Senha e Confirmar senha. Um botão 'Continuar' está visível na base do formulário, com o indicador '1/3'. Na barra inferior do aplicativo, há ícones para 'Agenda', 'Meu Perfil', 'Notificações' e 'Mais'.

FONTE: Os autores (2020).

DV02 – Tela de minhas informações (Dados de endereço)

FIGURA 100 - DV02 - TELA DE MINHAS INFORMAÇÕES – DADOS DO ENDEREÇO



The screenshot shows a mobile application interface for editing profile information. At the top, there is a back arrow and the text 'Minhas informações' next to a circular profile picture icon. Below this, the section is titled 'Dados de endereço'. It contains several input fields: 'CEP', 'Rua', 'Número', 'Complemento', 'Estado' (a dropdown menu), and 'Cidade' (a dropdown menu). At the bottom of the form is a purple 'Continuar' button with a '2/3' progress indicator. The bottom navigation bar includes icons for 'Agenda', 'Meu Perfil', 'Notificações', and 'Mais'.

FONTE: Os autores (2019).

DV03 - Tela de minhas informações (Foto)

FIGURA 101 - DV03 - TELA DE MINHAS INFORMAÇÕES - FOTO



The screenshot shows the same 'Minhas informações' screen, but at a different step. The text reads 'Estamos finalizando... agora só falta sua foto de perfil!'. In the center, there is a large grey circle with the text 'Insira foto aqui'. Below the circle are two buttons: 'Fotografar' with a camera icon and 'Carregar' with a cloud upload icon. At the bottom is a purple 'Salvar' button with a '3/3' progress indicator. The bottom navigation bar remains the same as in the previous screenshot.

FONTE: Os autores (2020).

Fluxo de Eventos Principal

1. O sistema apresenta a tela DV01.
2. O prestador preenche seus dados e clica no botão “Continuar”. (E1)
3. O sistema carrega a tela DV02.
4. O prestador preenche seus dados e clica no botão “Continuar”. (E1)
5. O sistema carrega a tela DV03.
6. O prestador tira uma foto ou adiciona uma de sua galeria de imagens e clica no botão “Continuar”. (A1)
7. O caso de uso está finalizado.

Fluxos Alternativos

A1 – O usuário clica no botão “carregar imagem”.

1. O sistema irá abrir um pop-up para o usuário selecionar a imagem desejada no seu celular.

Fluxos de Exceção

E1 – Campos vazios.

1. O sistema mostra mensagem “Existem campos vazios, favor preenchê-los para continuar”.

Regras de negócio

Não há regras de negócio.

UC004 – Gerenciar serviços**Descrição**

Neste caso de uso o prestador poderá gerenciar seus serviços. Ele será capaz de visualizar, inserir, editar e deletar seus serviços oferecidos.

Data View

DV01 – Lista de serviços

FIGURA 102 - DV01 - TELA DE LISTAGEM DE SERVIÇOS



FONTE: Os autores (2020).

DV02 – Edição/Inclusão de serviços

FIGURA 103 - DV01 - TELA DE INCLUSÃO/EDIÇÃO DE SERVIÇOS



FONTE: Os autores (2020)

Fluxo de Eventos Principal

1. O sistema apresenta a tela DV01.
2. Usuário clica no botão “incluir”. (A1) (A2)
3. O sistema carrega a tela DV02. (RN1)
4. Usuário preenche seus dados e clica no botão “confirmar”. (E1)
5. O caso de uso está finalizado.

Fluxos Alternativos

A1 – O usuário clica no botão com ícone de lápis.

1. O sistema irá carregar a tela DV02 com os campos já preenchidos.
2. O usuário irá alterar seus dados e clicar no botão “confirmar”. (E1)
3. O caso de uso está finalizado.

A2 – O usuário clica no botão com ícone de lixeira.

1. O sistema irá mostrar um pop-up perguntando se realmente deseja excluir esse serviço.
2. O usuário irá aceitar a mensagem.
3. O sistema irá excluir o serviço.
4. O caso de uso está finalizado.

Fluxos de Exceção

E1 – Campos vazios.

1. O sistema mostra mensagem “Existem campos vazios, favor preenchê-los para continuar”.

Regras de negócio

RN1 – A tela DV02, quando for carregada a partir do botão “incluir”, irá estar escrito inclusão no topo da tela. Já quando for carregada a partir do ícone de lápis, estará escrito edição no topo da tela.

UC005 – Gerenciar formas de pagamento

Descrição

O usuário poderá editar suas formas de pagamento.

Data View

DV01 – Tela de dados financeiros

FIGURA 104 - DV01 - TELA DE DADOS FINANCEIROS

← Financeiro

Código do Banco

Número da agência

Número da Conta

Formas de recebimento:

Cartão de crédito (Aplicativo)

Cartão de débito

Dinheiro

Cancelar Salvar

Agência Meu Perfil Notificações Mais

FONTE: Os autores (2020).

Fluxo de Eventos Principal

1. O sistema irá carregar a tela DV01 com os campos já preenchidos.
2. O usuário irá alterar seus dados e clicar no botão “confirmar”. (E1) (A1)
3. O caso de uso está finalizado.

Fluxos Alternativos

A1 – O usuário clica no botão com “cancelar”.

1. O sistema não irá salvar as alterações e retorna para o menu principal.
2. O caso de uso está finalizado.

Fluxos de Exceção

E1 – Campos vazios.

1. O sistema emite a seguinte mensagem: “Existem campos vazios, favor preenchê-los para continuar”.

Regras de negócio

Não há regras de negócio.

UC006 – Consultar histórico de serviços realizados

Descrição

Neste caso de uso o prestador poderá consultar seu histórico referentes aos seus serviços prestados.

Data View

DV01 – Tela de lista de histórico de serviços prestados

FIGURA 105 - DV01 - TELA DE HISTÓRICO DE SERVIÇOS PRESTADOS



FONTE: Os autores (2020).

DV02 – Detalhe do serviço

FIGURA 106 - DV02 - TELA DE HISTÓRICO DE SERVIÇO DETALHADO



FONTE: Os autores (2020).

Fluxo de Eventos Principal

1. O sistema irá carregar a tela DV01.
2. O usuário irá escolher o tipo de serviço e clicar no botão “buscar”. (A1)
3. O sistema irá carregar uma lista com os serviços já efetuados.
4. O usuário irá clicar no ícone de folha para mais informações.
5. O sistema irá carregar a tela DV02.
6. O caso de uso está finalizado.

Fluxos Alternativos

A1 – O usuário poderá escolher um range de datas para a pesquisa e clicar no botão “buscar”.

1. O sistema irá carregar uma lista com os serviços já efetuados.
2. O usuário irá clicar no ícone de folha para mais informações.
3. O sistema irá carregar a tela DV02.
4. O caso de uso está finalizado.

Fluxos de Exceção

Não há fluxo de exceção.

UC007 – Visualizar notificações

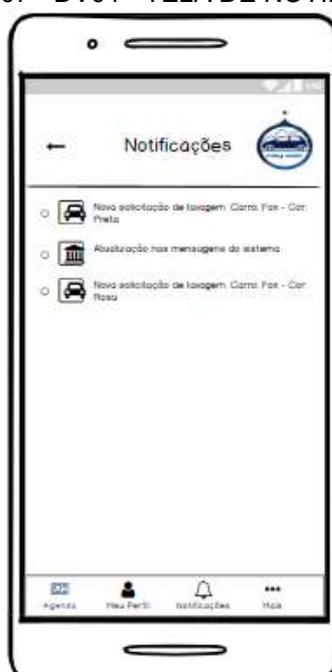
Descrição

O usuário terá uma aba exclusiva com suas notificações do sistema e de novos serviços.

Data View

DV01 – Tela de notificações

FIGURA 107 - DV01 - TELA DE NOTIFICAÇÕES



FONTE: Os autores (2020).

Fluxo de Eventos Principal

1. O sistema realiza uma busca no banco de dados referente a todas as notificações do prestador.
2. O sistema apresenta a tela DV01.
3. O prestador clica em uma notificação de novo serviço. (A1)
4. O sistema chama o UC009 – Solicitações Recebidas.
5. O caso de uso está finalizado.

Fluxos Alternativos

A1 – O usuário clica em uma notificação de sistema.

1. O sistema irá mostrar um pop-up com a mensagem.
2. O usuário irá aceitar a mensagem.
3. O caso de uso está finalizado.

Fluxos de Exceção

Não há fluxo de exceção.

Regras de negócio

Não há regras de negócio.

UC008 – Realizar atendimento

Descrição

Neste caso de uso o usuário poderá visualizar e realizar o atendimento do serviço solicitado.

Data View

DV01 – Tela de atendimento (Lista de atendimentos)

FIGURA 108 - DV01 - TELA DE ATENDIMENTO



FONTE: Os autores (2020).

DV02 – Tela de início do atendimento

FIGURA 109 - DV02 - TELA DE INICIO DE ATENDIMENTO



FONTE: Os autores (2020).

DV03 – Tela de atendimento (Atendimento em andamento)

FIGURA 110 - DV03 - TELA DE ATENDIMENTO EM ANDAMENTO



FONTE: Os autores (2020).

DV04 – Tela de atendimento (Concluído)

FIGURA 111 - DV04 - TELA DE ATENDIMENTO CONCLUÍDO



FONTE: Os autores (2020).

Fluxo de Eventos Principal

1. O sistema irá carregar a tela DV01- Tela de Listagem de atendimentos.
2. O usuário clica no botão de ícone play para iniciar o atendimento. (A1)
3. O sistema irá carregar a tela DV02 – Tela de atendimento inicial.
4. O usuário irá clicar no botão “Iniciar”
5. O sistema irá carregar a tela DV03 – Tela de atendimento em andamento.
6. O usuário irá clicar no botão “Encerrar”.
7. O sistema irá carregar a tela DV04 – Tela de atendimento concluído.
8. O usuário irá tirar uma foto do carro lavado, atualizar o status e preencher uma observação. (E1)
9. O usuário irá clicar no botão “Concluir”
10. O caso de uso está finalizado.

Fluxos Alternativos

A1 – O usuário clica no ícone X.

1. O sistema irá mostrar um pop-up com a mensagem se realmente deseja cancelar o serviço.

2. O usuário irá aceitar a mensagem.
3. O caso de uso está finalizado.

Fluxos de Exceção

E1 – Campos vazios.

1. O sistema mostra mensagem “Existem campos vazios, favor preenche-los para continuar”.

Regras de negócio

Não há regras de negócio.

UC009 – Solicitações Recebidas

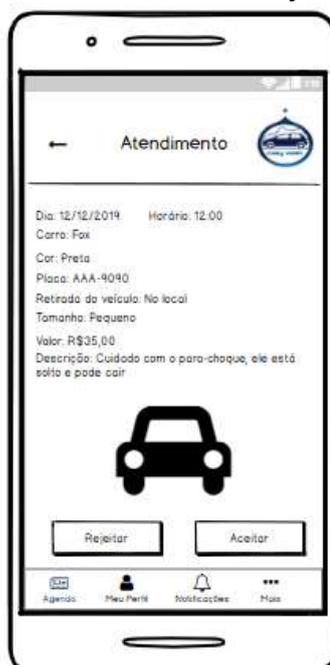
Descrição

O prestador poderá aceitar ou recusar um atendimento de serviço solicitado pelo cliente.

Data View

DV01 – Tela de confirmação de serviço

FIGURA 112 - DV01 - TELA DE CONFIRMAÇÃO DE ATENDIMENTO



FONTE: Os autores (2020).

Fluxo de Eventos Principal

1. O sistema realiza uma busca no banco de dados para recuperar os dados da solicitação.
2. O sistema irá carregar a tela DV01 – Tela de Confirmação de atendimento.
3. O usuário clica no botão “Aceitar”. (A1)
4. O sistema salva os dados do atendimento no banco de dados.
5. O sistema chama o UC021.
6. O caso de uso está finalizado.

Fluxos Alternativos

A1 – O usuário clica no botão “Rejeitar”.

1. O sistema irá mostrar um pop-up perguntando se deseja realmente rejeitar o serviço.
2. O usuário irá clicar no botão OK.
3. O sistema chama o UC021.
4. O caso de uso é finalizado.

Fluxos de Exceção

Não há fluxo de exceção.

Regras de negócio

Não há regras de negócio.

APÊNDICE I - ESPECIFICAÇÃO SISTEMA WEB MODERADOR

UC001 – Moderador poderá realizar login no sistema

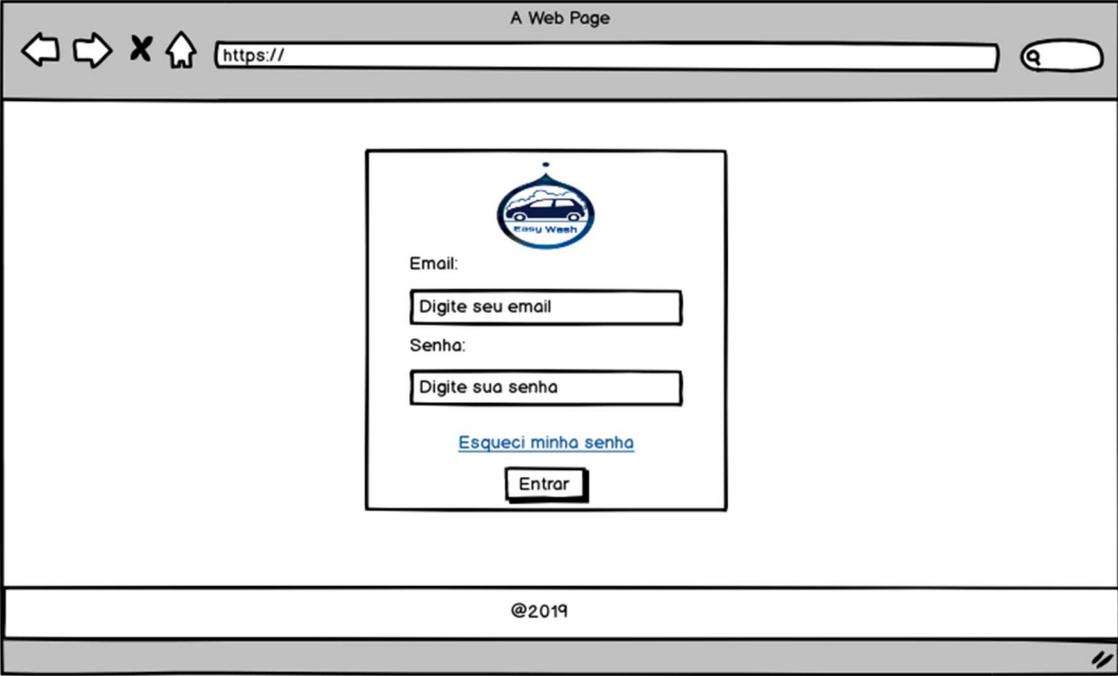
Descrição

Neste caso de uso o moderador poderá realizar login utilizando seu e-mail e senha para ter acesso ao sistema da Easy Wash.

Data View

DV01 – Formulário de Login do sistema

FIGURA 113 - DV01 - FORMULÁRIO DE LOGIN DO SISTEMA



The image shows a web browser window titled "A Web Page". The address bar contains "https://". The main content area displays a login form for "Easy Wash". The form includes the following elements:

- Logo of "Easy Wash" (a car being washed).
- Label "Email:" followed by a text input field containing the placeholder text "Digite seu email".
- Label "Senha:" followed by a text input field containing the placeholder text "Digite sua senha".
- A blue hyperlink labeled "Esqueci minha senha".
- A button labeled "Entrar".

At the bottom of the browser window, the text "@2019" is visible.

FONTE: Os autores (2020).

DV02 – Formulário de Recuperação de senha

FIGURA 114 - DV02 - FORMULÁRIO DE RECUPERAÇÃO DE SENHA

The image shows a web browser window titled "A Web Page". The address bar contains "https://". The main content area displays a form for password recovery. At the top of the form is a logo for "Easy Wash" featuring a car and a water drop. Below the logo, the text reads: "Informe aqui seu email, que reenviamos um link para recuperação de seu acesso." There is a text input field with the placeholder "Digite aqui seu email". Below the input field is a button labeled "Enviar". At the bottom of the form is a blue hyperlink that says "Retornar para a página de login". The footer of the browser window shows "@2019".

FONTE: Os autores (2020).

Ator Primário

Moderador

Fluxo de Eventos Principal

1. O sistema apresenta a tela DV01. (A1)
2. O sistema desabilita o botão Entrar.
3. O moderador preenche os campos e-mail e senha do formulário. (RN2)
4. O sistema habilita o botão Entrar.
5. O moderador clica no botão Entrar.
6. O sistema realiza uma busca no banco de dados com base nas informações inseridas. (E1)
7. O sistema redireciona o usuário para a página home.
8. O caso de uso é encerrado.

Fluxos Alternativos

A1 – O moderador clica sobre o link “Esqueci minha senha”.

1. O sistema redireciona o usuário para a página de recuperação de senha.

2. O sistema apresenta a tela DV02.
3. O moderador preenche o campo e-mail do formulário. (RN1)
4. O moderador clica no botão Enviar.
5. O sistema emite a mensagem “Nova senha enviada com sucesso! ”.(RN3)

Fluxos de Exceção

E1 – Usuário não encontrado.

1. O sistema emite a mensagem “Dados de login ou senha inválidos! ”.
2. O sistema retorna para o início do fluxo principal.

Regras de negócio

RN1. O campo e-mail deverá possuir uma máscara para e-mail.

RN2. O campo senha deverá ter seu conteúdo mascarado.

RN3. O corpo do e-mail deve conter uma nova senha para o moderador poder acessar o sistema novamente contendo 8 caracteres.

UC002 – Gerenciar Moderadores

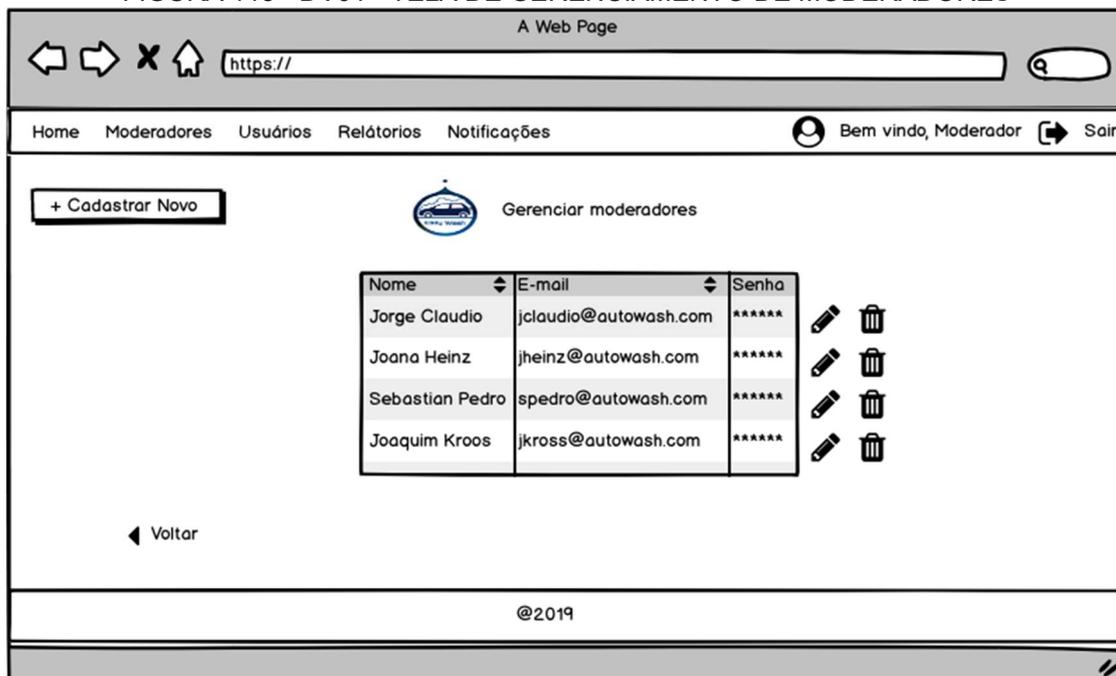
Descrição

Neste caso de uso o moderador logado poderá gerenciar outros moderadores cadastrados no sistema, podendo incluir, atualizar ou deletá-los do sistema.

Data View

DV01 – Tela de Gerenciamento de moderadores

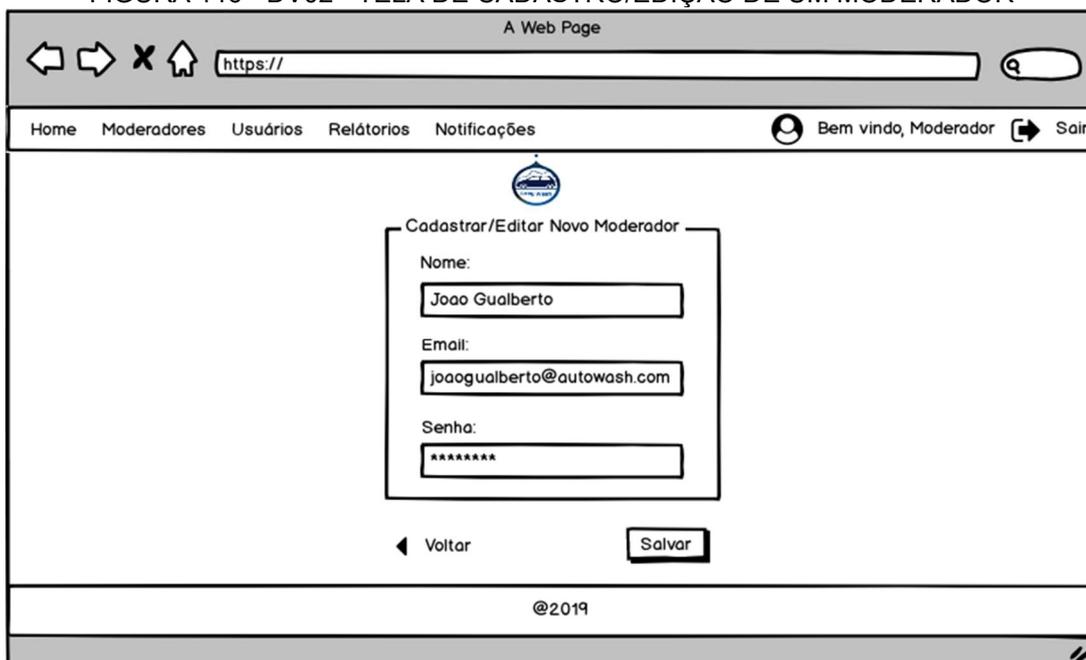
FIGURA 115 - DV01 - TELA DE GERENCIAMENTO DE MODERADORES



FONTE: Os autores (2020).

DV02 – Tela de Cadastro/Edição de um moderador

FIGURA 116 - DV02 - TELA DE CADASTRO/EDIÇÃO DE UM MODERADOR



FONTE: Os autores (2020).

Ator primário

Moderador

Fluxo de Eventos Principal

1. O sistema apresenta a tela DV01 – Tela de gerenciamento de moderadores. (A1)
2. O sistema carrega a lista de moderadores cadastrados no banco de dados.
(A1)(A2)(A3)(A4)
3. O caso de uso é encerrado.

Fluxos Alternativos

A1 – O moderador clica no botão Sair.

1. O sistema encerra a sessão e realiza um redirecionamento para a página home.

A2 – O moderador clica no botão Voltar.

10. O sistema realiza um redirecionamento para a página home.

A3 – O moderador clica no botão Cadastrar Novo.

1. O sistema redireciona para a tela DV02 - Cadastro/Edição de um moderador.
2. O moderador preenche os campos nome, e-mail e senha do formulário.
(R1)(R2)
3. O moderador clica no botão Salvar.
4. O sistema persiste as informações e as envia para o banco de dados. (E2)
5. O sistema emite a mensagem: “Moderador cadastrado com sucesso!”.
6. O caso de uso retorna para o fluxo principal.

A3 – O moderador clica no botão Editar.

1. O sistema redireciona para a tela DV02 - Cadastro/Edição de um moderador.
2. O sistema preenche os campos do formulário com os dados atuais do moderador.
3. O moderador atualiza os campos do formulário. (RN1)(RN2)
4. O moderador clica no botão Salvar.
5. O sistema envia as informações para o banco de dados. (E2)
6. O sistema emite a mensagem: “Moderador atualizado com sucesso!”.
7. O caso de uso retorna para o fluxo principal.

A4 – O moderador clica no botão Excluir.

1. O sistema mostra um modal com a seguinte mensagem: “Tem certeza que deseja excluir o moderador XXXX?”.
2. O moderador clica no botão Ok.
3. O sistema exclui o moderador do sistema. (E2)
4. O sistema emite a mensagem: “Moderador excluído com sucesso! ”.
5. O caso de uso retorna para o fluxo principal.

Fluxo de Exceção

E1 – E-mail já cadastrado

1. O sistema deve emitir a seguinte mensagem: “E-mail já cadastrado no sistema.”

E2 – Erro com operação no banco de dados.

10. O sistema emite a seguinte mensagem:” Erro ao inserir/atualizar ou excluir moderador. Por favor, tente novamente!”.

Regras de negócio

- RN1. O campo nome do moderador deve conter entre 5 e 30 caracteres.
- RN2. O campo senha deve conter 8 caracteres.

UC003 – Moderador poderá gerenciar clientes do sistema.

Descrição

Neste caso de uso o cliente poderá gerenciar os clientes cadastrados no sistema, com as operações de visualizar, editar e deletar dados de um cliente.

Data View

DV01 – Tela de Gerenciamento de clientes

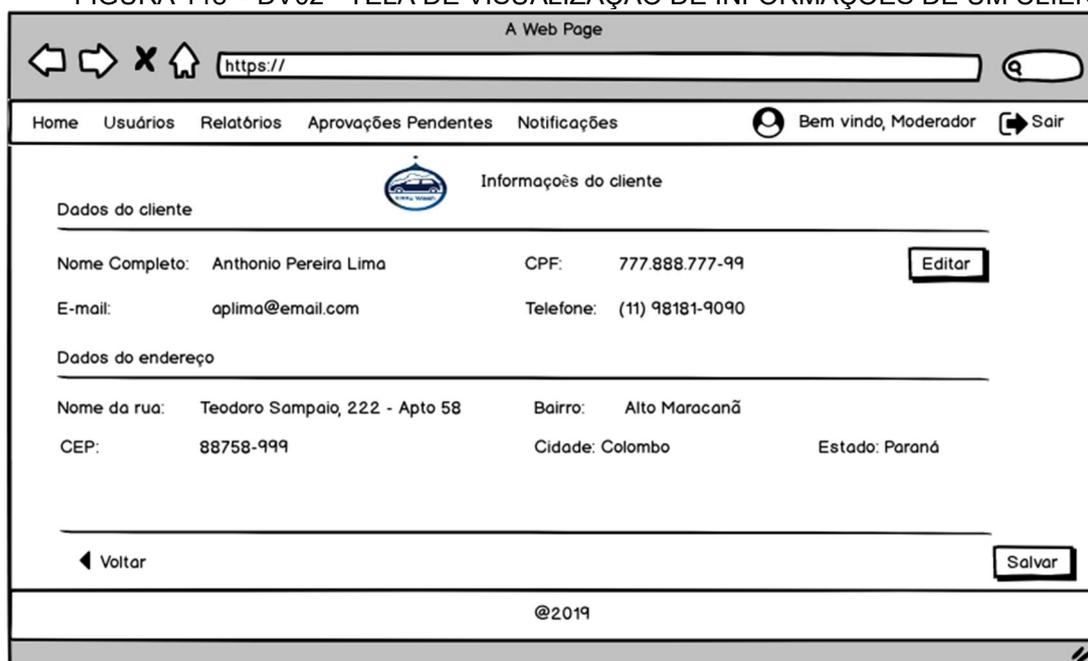
FIGURA 117 - DV01 - TELA DE GERENCIAMENTO DE CLIENTES



FONTE: Os autores (2020).

DV02 – Tela Visualização de informações de um cliente.

FIGURA 118 – DV02 - TELA DE VISUALIZAÇÃO DE INFORMAÇÕES DE UM CLIENTE



FONTE: Os autores (2020).

Ator primário

Moderador

Fluxo de Eventos Principal

1. O sistema apresenta a tela DV01 – Tela de gerenciamento de clientes. (A1)(A2)(A3)(A4).
2. O moderador preenche o campo do formulário de busca com o nome do cliente. (RN1)
3. O cliente clica no ícone de pesquisa.
4. O sistema realiza uma busca no banco de dados com base nos dados preenchidos.
5. O caso de uso é encerrado.

Fluxo Alternativos

A1 – O moderador clica no botão Sair.

1. O sistema encerra a sessão e realiza um redirecionamento para a página home.

A2 – O moderador clica no botão Voltar.

1. O sistema realiza um redirecionamento para a página home.

A3 – O moderador clica no botão Mais informações.

1. O sistema realiza um redirecionamento para a tela DV02 – Visualizar informações de um cliente. (A5)
2. O sistema realiza uma busca no banco de dados para recuperar os dados do cliente.
3. O sistema apresenta as informações do cliente na tela.
4. O caso de uso retorna para o fluxo principal.

A4 – O moderador clica no botão Excluir.

1. O sistema mostra um modal com a seguinte mensagem: “Tem certeza que deseja excluir o cliente XXXX?”.
2. O moderador clica no botão Ok.
3. O sistema exclui o cliente do sistema. (E1)
4. O sistema emite a mensagem: “Cliente excluído com sucesso!”.
5. O caso de uso retorna para o fluxo principal.

A5 – O moderador clica no botão Editar.

1. O sistema habilita os seguintes campos do formulário para edição: E-mail, telefone e todos os campos referentes a dados de endereço.

Fluxos de exceção

E1 - Erro com operação no banco de dados.

1. O sistema emite a seguinte mensagem: " Erro ao excluir cliente. Por favor, tente novamente!".

Regras de Negócio

RN1. A busca deverá ser realizada com base no nome do cliente.

UC004 - Gerenciar prestadores de serviço.

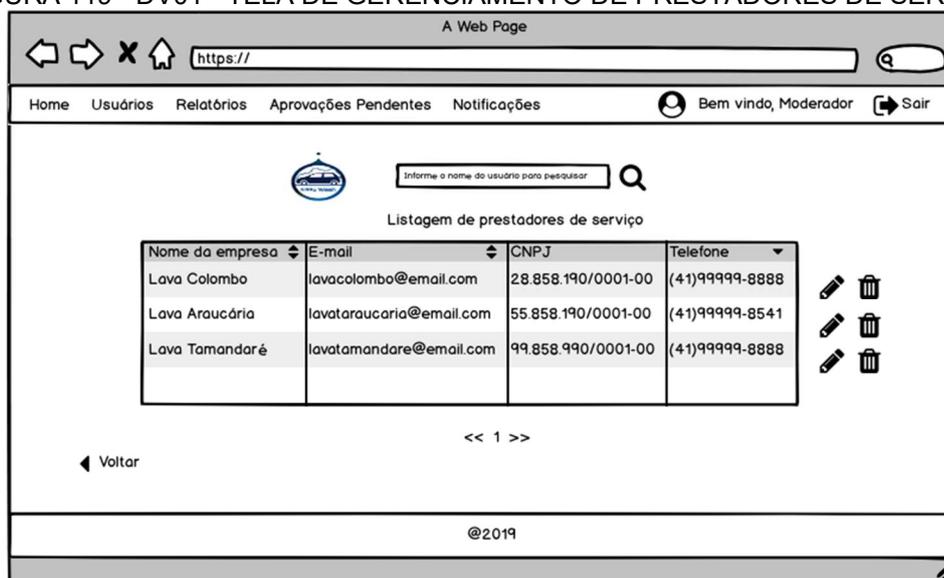
Descrição

Neste caso de uso o moderador poderá gerenciar os prestadores de serviço cadastrados no sistema, com as operações de visualizar, editar e deletar dados de um cliente.

Data View

DV01 – Tela de Gerenciamento de prestadores de serviço

FIGURA 119 - DV01 - TELA DE GERENCIAMENTO DE PRESTADORES DE SERVIÇO



FONTE: Os autores (2020)

DV02 – Tela Visualização de informações de um prestador.

FIGURA 120 - DV02 - TELA DE VISUALIZAÇÃO DE INFORMAÇÕES DE UM PRESTADOR DE SERVIÇO

The screenshot shows a web browser window with the following elements:

- Browser Address Bar:** "A Web Page" and "https://".
- Navigation Menu:** Home, Usuários, Relatórios, Aprovações Pendentes, Notificações.
- User Information:** Bem vindo, Moderador, Sair.
- Section Header:** Informações do prestador de serviço.
- Dados da empresa:**
 - Razão Social: Anthony Serviços LTDA
 - CNPJ: 145145214521
 - Telefone: (41)3029-0901
 - Endereço: Rua Samanta Vieira, 222 - Tatuquara - Curitiba - PR
 - Buttons: Editar, Salvar.
- Lista de serviços da empresa:**

Nome do Serviço	Descrição	Média Duração Prevista	Valor
Lavagem Básica	Lavagem simples externa + cera líquida	30 min	R\$ 20
Lavagem	externa + cera em pasta	120 min	R\$40
Higienização	Desinfecção interna estofamento +	60 min	R\$70
- Footer:** Voltar, @2019.

FONTE: Os autores (2020)

Ator primário

Moderador

Fluxos de Evento Principal

1. O sistema apresenta a tela DV01 – Tela de gerenciamento de prestadores de serviço (A1)(A2)(A3)(A4).
2. O sistema carrega a lista de prestadores de serviço cadastrados no banco de dados.
3. O caso de uso é encerrado.

Fluxos Alternativos

A1 – O moderador clica no botão Sair.

1. O sistema encerra a sessão e realiza um redirecionamento para a página home.

A2 – O moderador clica no botão Voltar.

1. O sistema realiza um redirecionamento para a página home.

A3 – O moderador clica no botão Mais informações.

1. O sistema realiza um redirecionamento para a tela DV02 – Visualizar informações de um prestador de serviço.
2. O sistema realiza uma busca no banco de dados para recuperar os dados de um prestador de serviço.
3. O sistema apresenta as informações do prestador de serviço na tela.
4. O caso de uso retorna para o fluxo principal.

A4 – O moderador clica no botão Excluir.

1. O sistema mostra um modal com a seguinte mensagem: “Tem certeza que deseja excluir o prestador XXXX?”.
2. O moderador clica no botão Ok.
3. O sistema exclui o prestador do sistema. (E1)
4. O sistema emite a mensagem: “Prestador de serviço excluído com sucesso! ”.
5. O caso de uso retorna para o fluxo principal.

A5 – O moderador clica no botão Editar.

1. O sistema habilitará os seguintes campos do formulário para edição:

Fluxos de exceção

E1 - Erro com operação no banco de dados.

1. O sistema emite a seguinte mensagem: ” Erro ao excluir prestador de serviço. Por favor, tente novamente! ”.

UC005 – Gerenciar solicitações de prestadores de serviço.

Descrição

Neste caso de uso o moderador poderá analisar as solicitações enviadas pelos prestadores de serviço que desejam oferecer seus serviços na plataforma. O moderador poderá visualizar todas as informações referentes a um pedido e aceitar ou rejeitar a solicitação.

Data View

DV01 – Tela de Listagem de solicitações de pedido.

FIGURA 121 - DV01 - TELA DE LISTAGEM DE SOLICITAÇÕES DE PEDIDO PARA ACESSO A PLATAFORMA

Razão Social	Nome do proprietário	E-mail	CNPJ	Telefone
Giacomo Serviços	Giacomo Serviços	giacomog@email.com	111.111.111-11	(41)99999-8888
Mariah Serviços	Mariah Maclachlan	maclachlan@email.com	222.222.222-22	(41)99999-8541
Valerie Serviços	Valerie Liberty	vliberty@email.com	333.333.333-30	(41)99999-8888

FONTE: Os autores (2020).

DV02 – Tela Visualização de informações de um pedido.

FIGURA 122 - DV02 - TELA DE VISUALIZAÇÃO DE SOLICITAÇÃO DE PEDIDO PARA ACESSO A PLATAFORMA

Dados do prestador de serviço

Solicitação de cadastro de prestador de serviço Editar

Nome Completo: Anthony Pereira LTDA CPF: 745.541.222-99 E-mail: anthony@outlook.com Telefone: (21)98199-9090

Dados da empresa

Razão Social: Anthony Serviços LTDA CNPJ: 145145214521 Telefone: (41)3029-0901

Endereço: Rua Samanta Vieira, 222 - Tatuquara - Curitiba - PR Horário de atendimento: Seg a Dom das 08:00 às 17:00 hrs

Dados da conta bancária

Código do banco: 7070 Numero de agência: 8511 Numero da conta: 12458-3

Lista de serviços da empresa: Polimento, lavagem à seco, lavagem interna e vitrificação

Inserir justificativa: Aprovado Não aprovado

Voltar Enviar

FONTE: Os autores (2020).

Ator primário

Moderador

Fluxos de Evento Principal

1. O sistema apresenta a tela DV01 – Tela de Listagem de solicitações de pedido.
(A1)(A2)(A3)
2. O sistema carrega a lista de pedidos pendentes de resposta a serem analisados no banco de dados.
3. O caso de uso é encerrado.

Fluxos Alternativos

A1 – O moderador clica no botão Sair.

1. O sistema encerra a sessão e realiza um redirecionamento para a página home.

A2 – O moderador clica no botão Voltar.

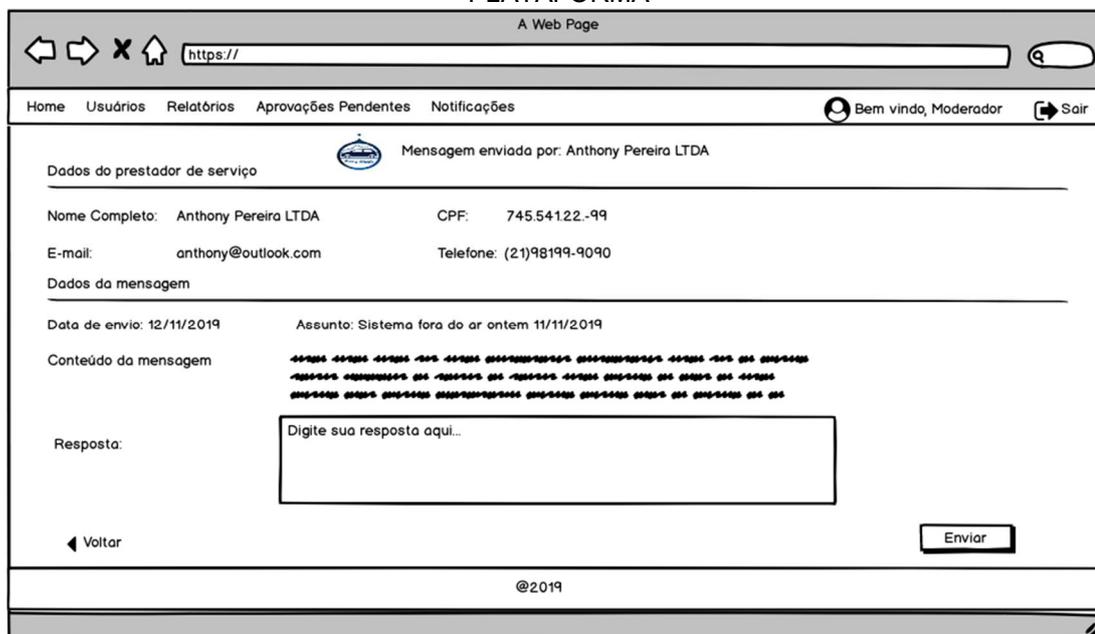
1. O sistema realiza um redirecionamento para a página home.

A3 – O moderador clica no botão Mais informações.

1. O sistema realiza um redirecionamento para a tela DV02 – Visualizar informações de solicitação de pedido.
2. O sistema realiza uma busca no banco de dados para recuperar os dados da solicitação.
3. O sistema apresenta as informações referentes a solicitação de acesso a plataforma na tela.
4. Após análise das informações da empresa e seu respectivo dono, o moderador deve selecionar uma das opções dos campos status do pedido. Sendo aprovado ou não aprovado. (RN01)
5. O moderador deve preencher o campo justificativa, com sua análise referente ao pedido. (RN02)
6. O moderador clica no botão Enviar.
7. O sistema envia as informações para o banco de dados do sistema.
8. O sistema envia um e-mail para o prestador de serviço com o resultado da análise do pedido. (E2)
9. O sistema emite uma mensagem: “E-mail enviado com sucesso para o prestador XXXX”.

DV02 – Tela de Visualização de mensagem

FIGURA 124 - DV02 - TELA DE VISUALIZAÇÃO DE MENSAGEM ENVIADA A MODERAÇÃO À PLATAFORMA



FONTE: Os autores (2020).

Ator primário

Moderador

Fluxos de Evento Principal

1. O sistema apresenta a tela DV01 – Tela de Listagem de mensagens enviadas pelos usuários do sistema. (A1)(A2)(A3)
2. O sistema carrega a lista de mensagens de usuários no banco de dados.
3. O caso de uso é encerrado.

Fluxos Alternativos

A1 – O moderador clica no botão Sair.

1. O sistema encerra a sessão e realiza um redirecionamento para a página home.

A2 – O moderador clica no botão Voltar.

1. O sistema realiza um redirecionamento para a página home.

A3 – O moderador clica no botão Mais informações.

1. O sistema realiza um redirecionamento para a tela DV02 – Visualizar mensagem do usuário.
2. O sistema realiza uma busca no banco de dados para recuperar os dados da mensagem.
3. O sistema apresenta as informações referentes a mensagem na tela.
4. O moderador deve enviar uma resposta ao usuário. (RN01)
5. O moderador clica no botão Enviar.
6. O sistema envia as informações para o banco de dados do sistema.
7. O sistema emite uma mensagem: “Resposta enviada com sucesso para o usuário XXXX”.
8. O caso de uso é encerrado.

Regras de negócio

RN01. O campo resposta é obrigatório.

UC007 – Gerar relatórios

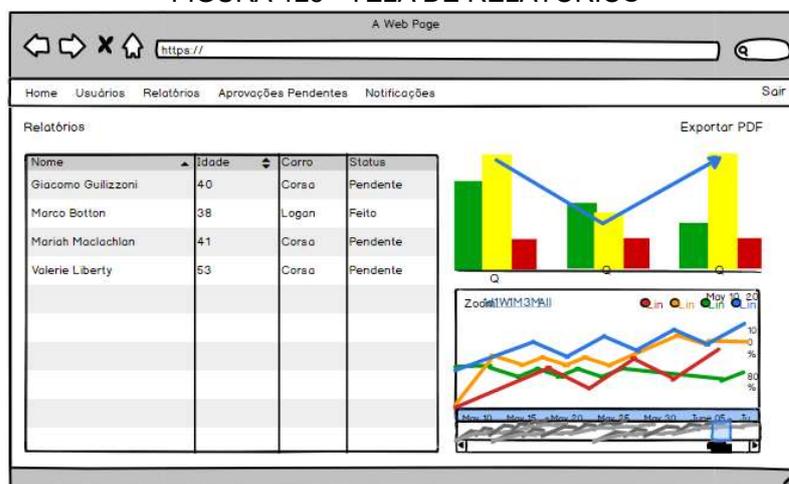
Descrição

Neste caso de uso o moderador poderá gerar relatórios com informações relevantes para o melhor controle do sistema e dos usuários.

Data View

DV01 – Tela de Listagem de mensagens enviadas pelos usuários.

FIGURA 125 - TELA DE RELATÓRIOS



FONTE: Os autores (2020).

Ator primário

Moderador

Fluxos de Evento Principal

1. O sistema apresenta a tela DV01 – Tela de Relatórios. (A1)
2. O sistema carrega as informações do banco de dados.
3. O usuário exporta para PDF os relatórios.
4. O caso de uso é encerrado.

Fluxos Alternativos

A1 – O moderador clica no botão Sair.

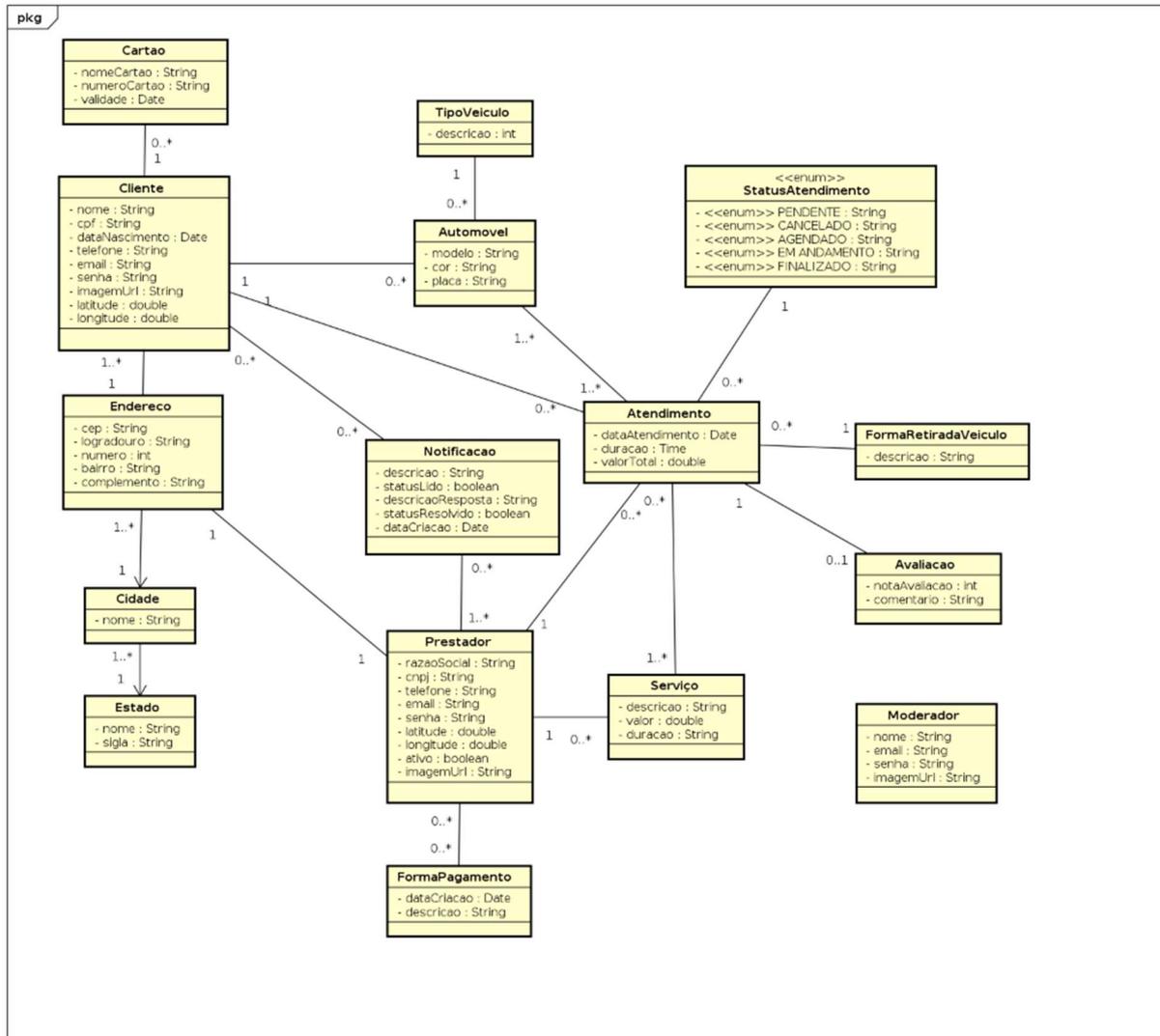
2. O sistema encerra a sessão e realiza um redirecionamento para a página home.

Regras de negócio

RN01. Todos os relatórios são exportados juntos em um único arquivo.

APÊNDICE J - DIAGRAMA DE CLASSES DE DOMÍNIO

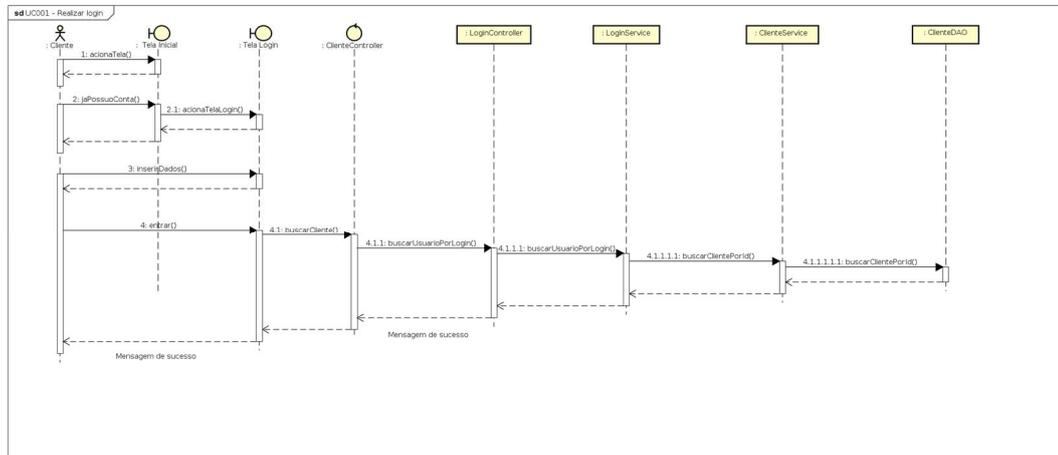
FIGURA 126 - DIAGRAMA DE CLASSES DE DOMÍNIO



FONTE: Os autores (2020).

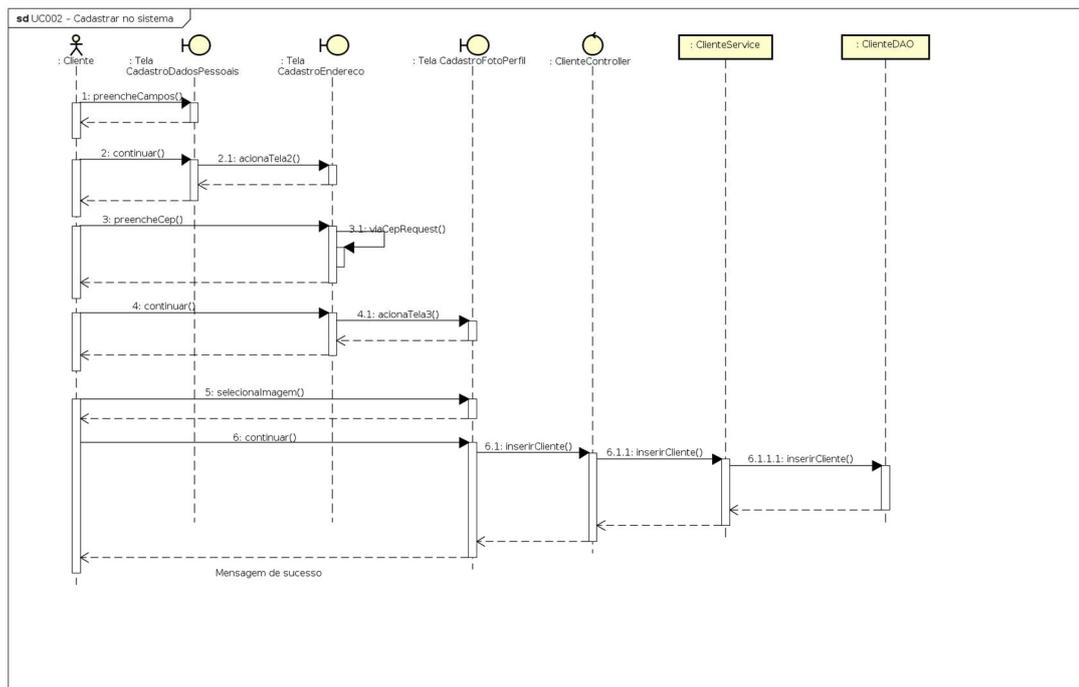
APÊNDICE K - DIAGRAMAS DE SEQUÊNCIA CLIENTE

FIGURA 127 - REALIZAR LOGIN



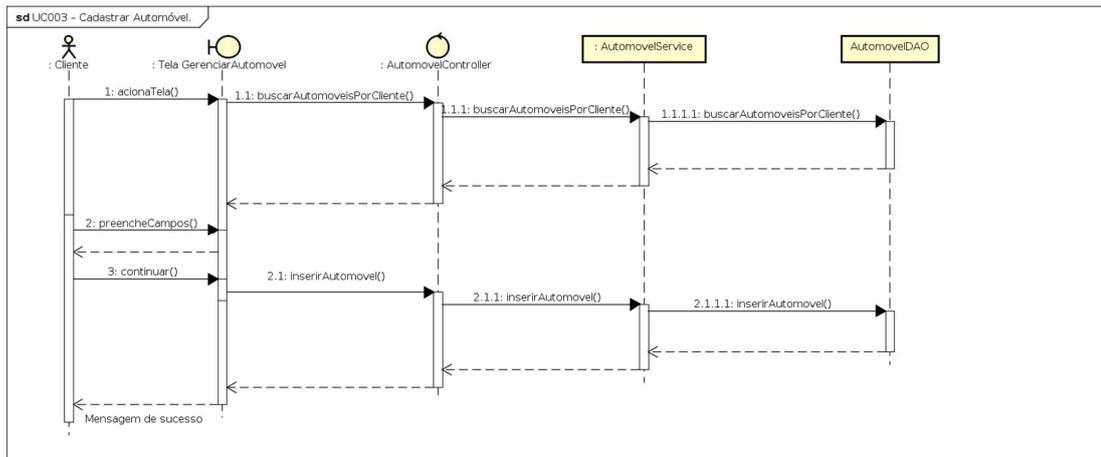
FONTE: Os autores (2019).

FIGURA 128 - CADASTRAR-SE NO SISTEMA



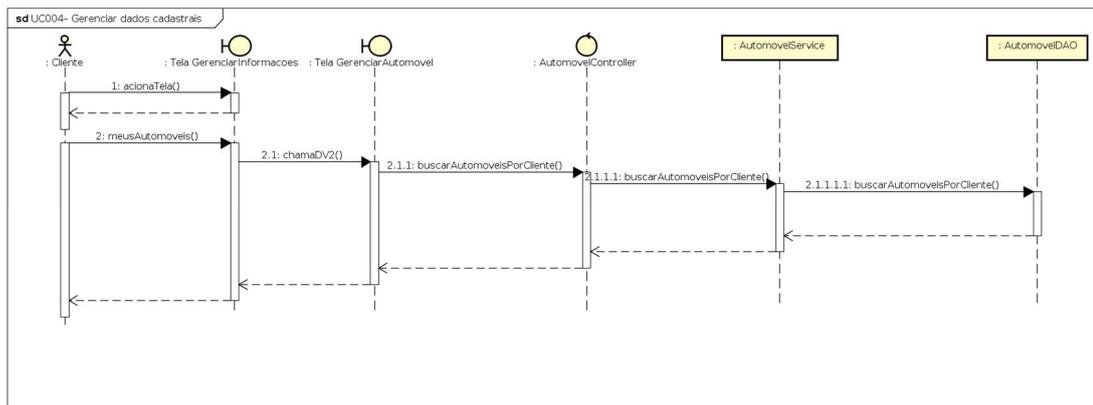
FONTE: Os autores (2020).

FIGURA 129 - CADASTRAR AUTOMÓVEL



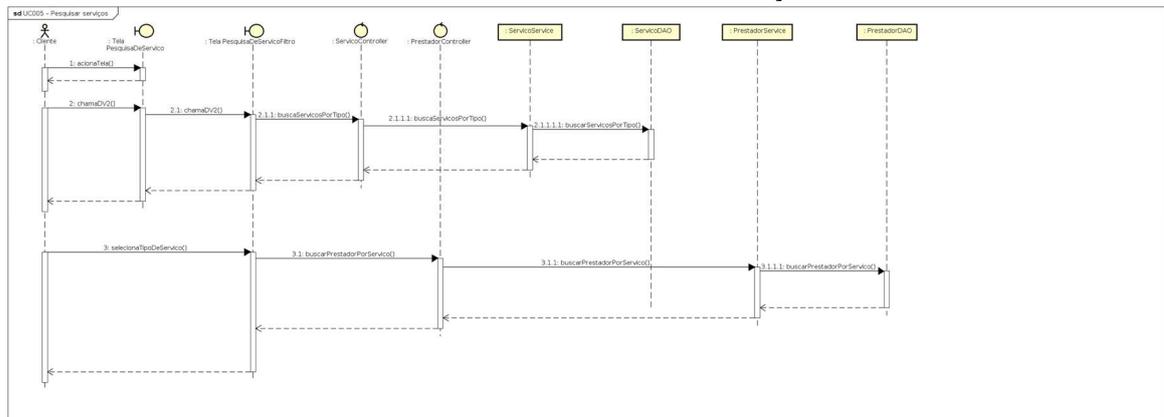
FONTE: Os autores (2020).

FIGURA 130 - GERENCIAR DADOS CADASTRAIS



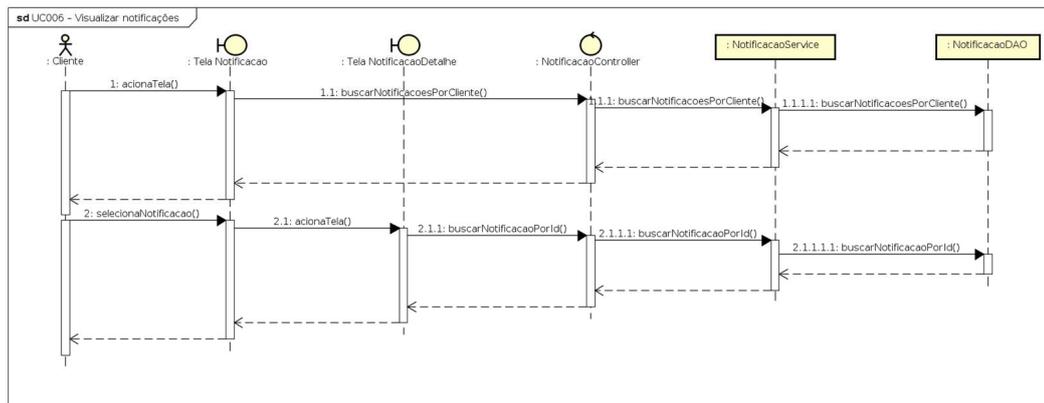
FONTE: Os autores (2020).

FIGURA 131 - PESQUISAR SERVIÇOS



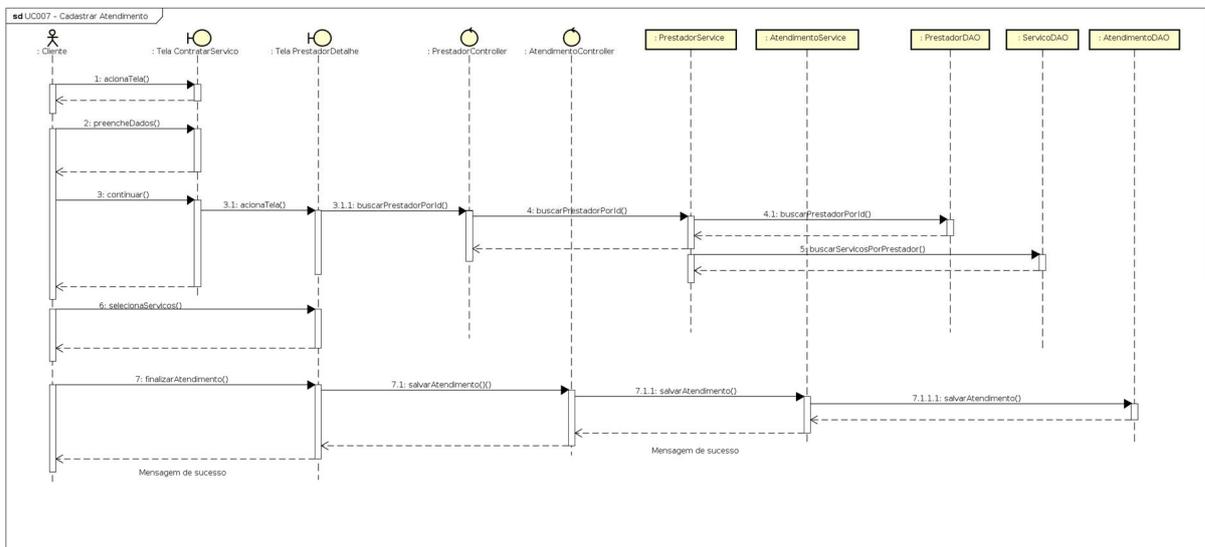
FONTE: Os autores (2020).

FIGURA 132 - VISUALIZAR NOTIFICAÇÕES



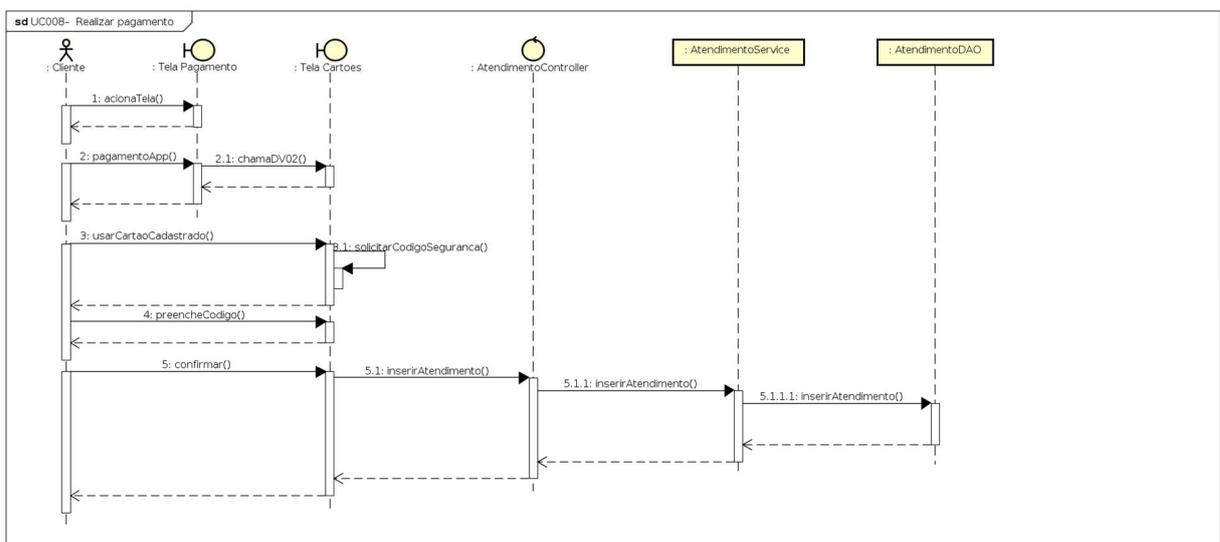
FONTE: Os autores (2020).

FIGURA 133 - CADASTRAR ATENDIMENTO



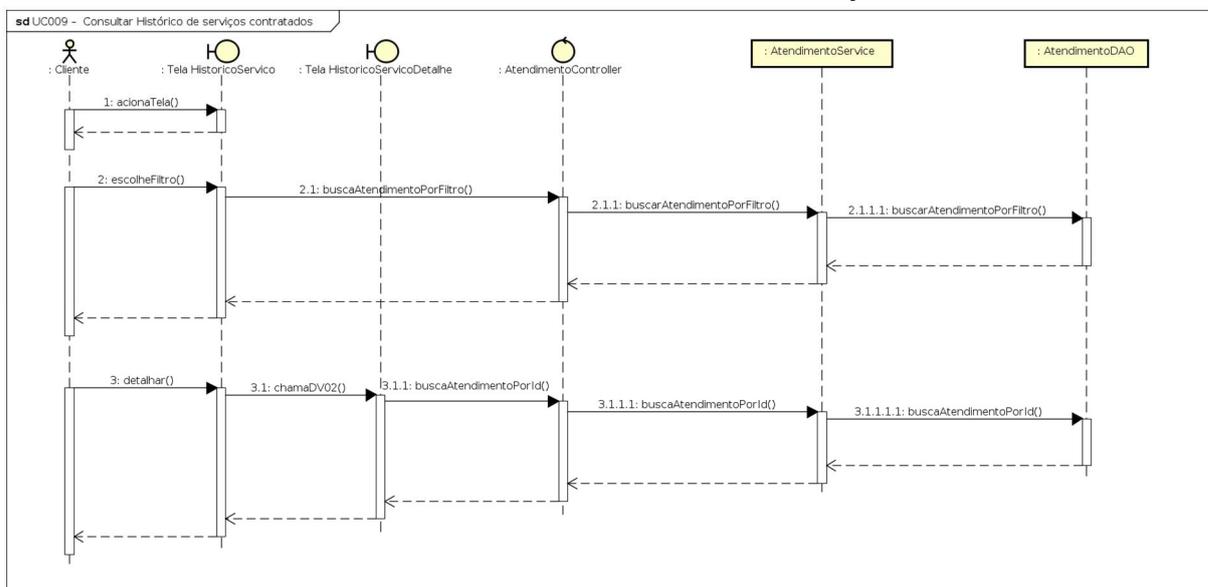
FONTE: Os autores (2020).

FIGURA 134 - REALIZAR PAGAMENTO



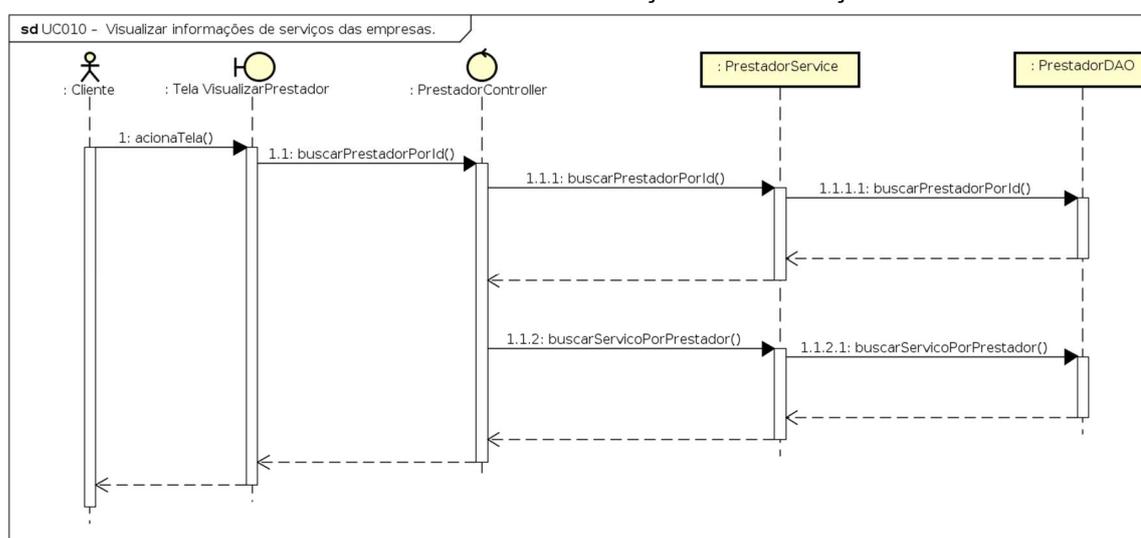
FONTE: Os autores (2020).

FIGURA 135 - CONSULTAR HISTÓRICO DE SERVIÇOS COTRATADOS



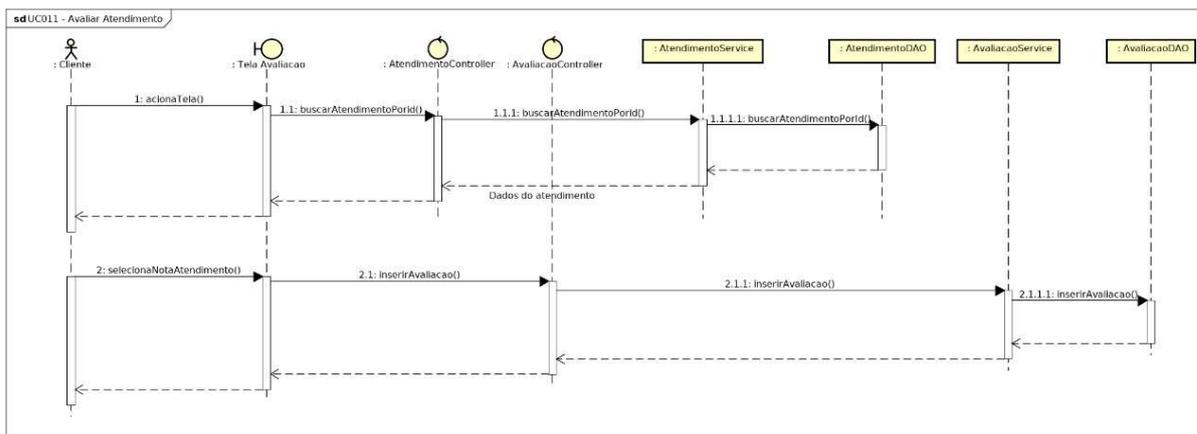
FONTE: Os autores (2020).

FIGURA 136 - VISUALIZAR INFORMAÇÕES E SERVIÇOS DA EMPRESA



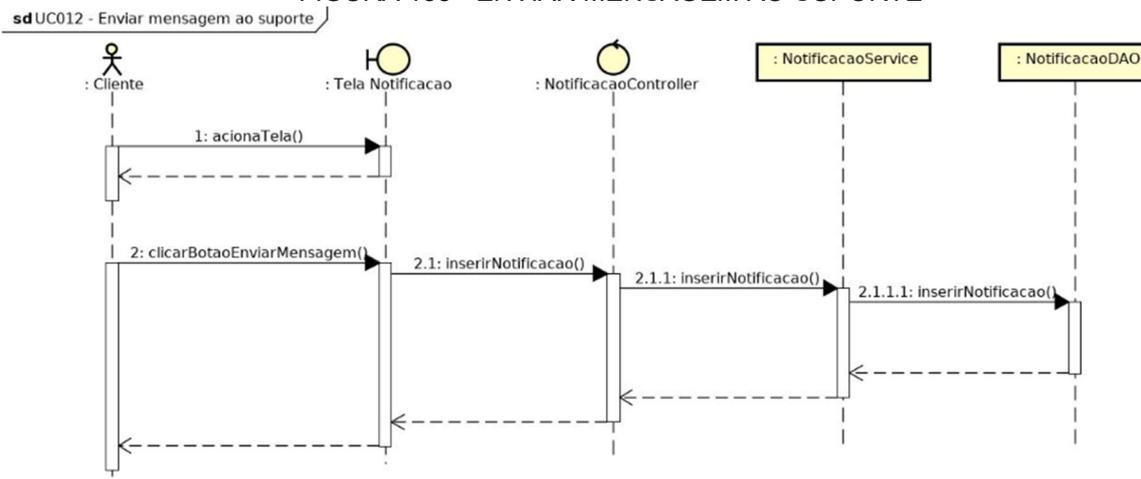
FONTE: Os autores (2019).

FIGURA 137 - AVALIAR ATENDIMENTO



FONTE: Os autores (2020).

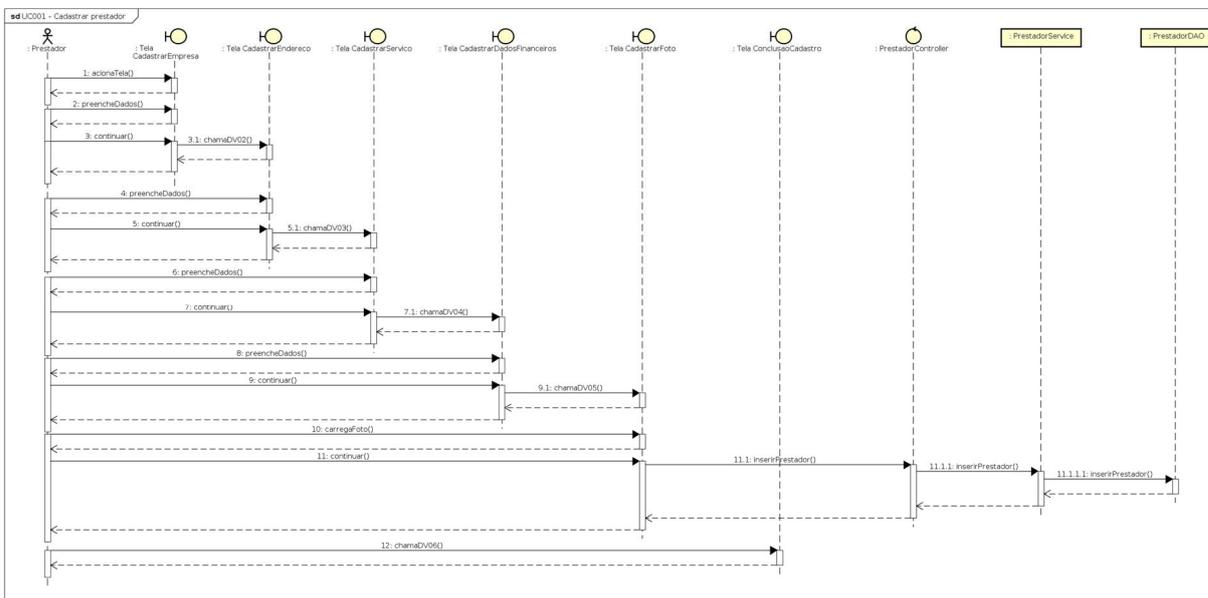
FIGURA 138 - ENVIAR MENSAGEM AO SUPORTE



FONTE: Os autores (20120).

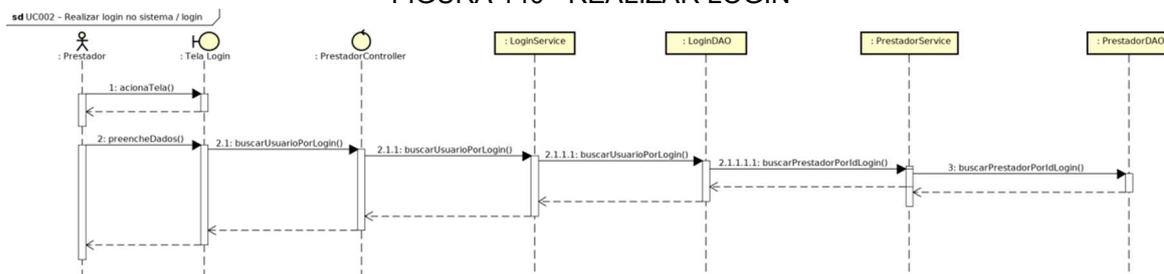
APÊNDICE L - DIAGRAMAS DE SEQUÊNCIA PRESTADOR

FIGURA 139 - CADASTRAR PRESTADOR



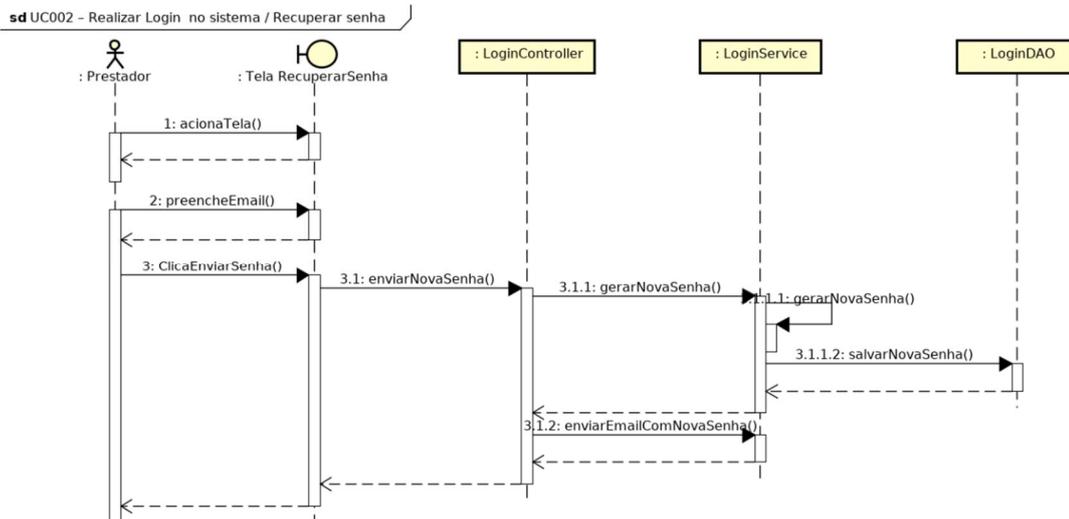
FONTE: Os autores (2020).

FIGURA 140 - REALIZAR LOGIN



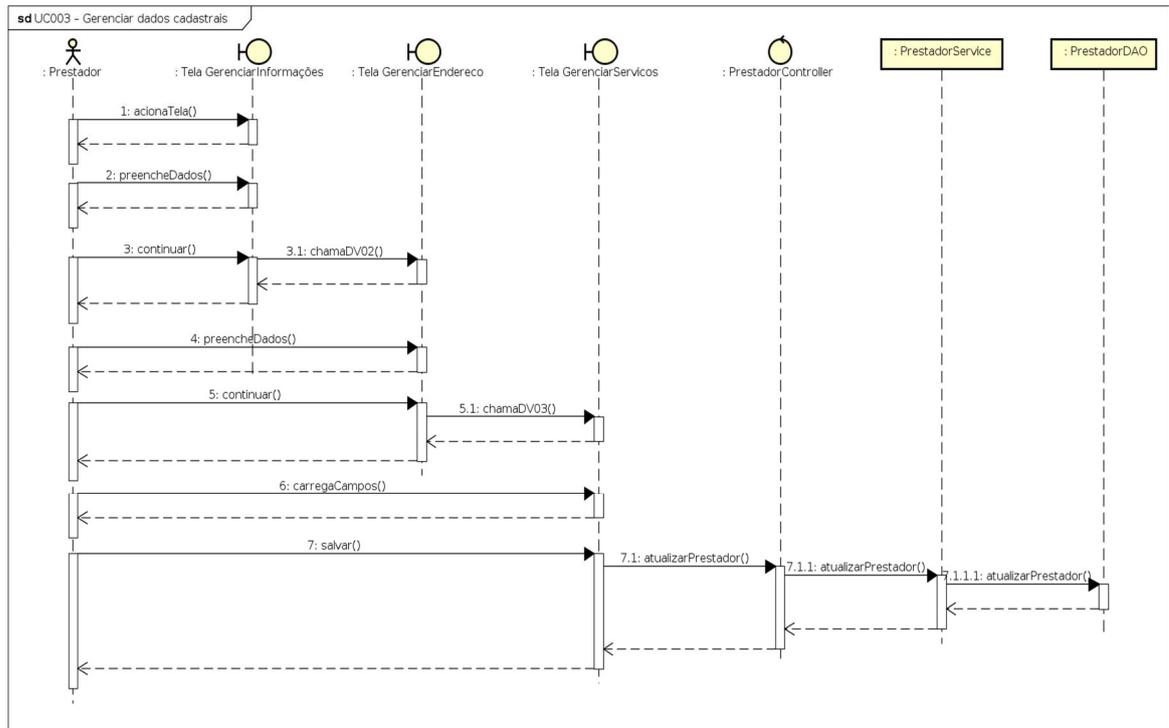
FONTE: Os autores (2020).

FIGURA 141 - REALIZAR LOGIN/ RECUPERAR SENHA



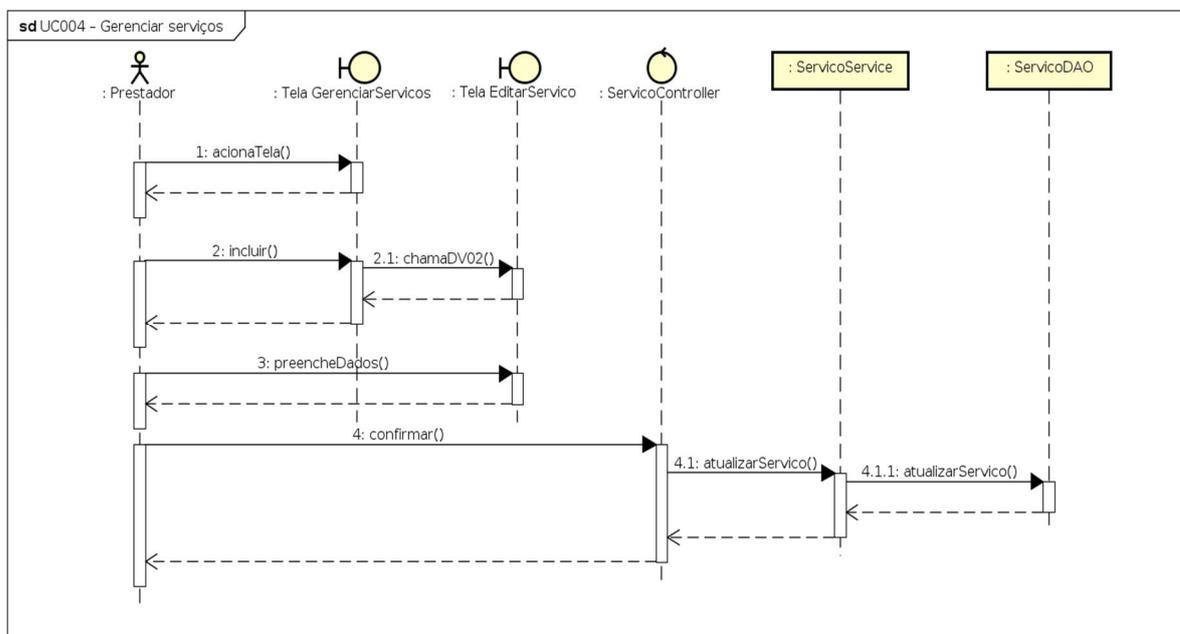
FONTE: Os autores (2020).

FIGURA 142 - GERENCIAR DADOS CADASTRAIS



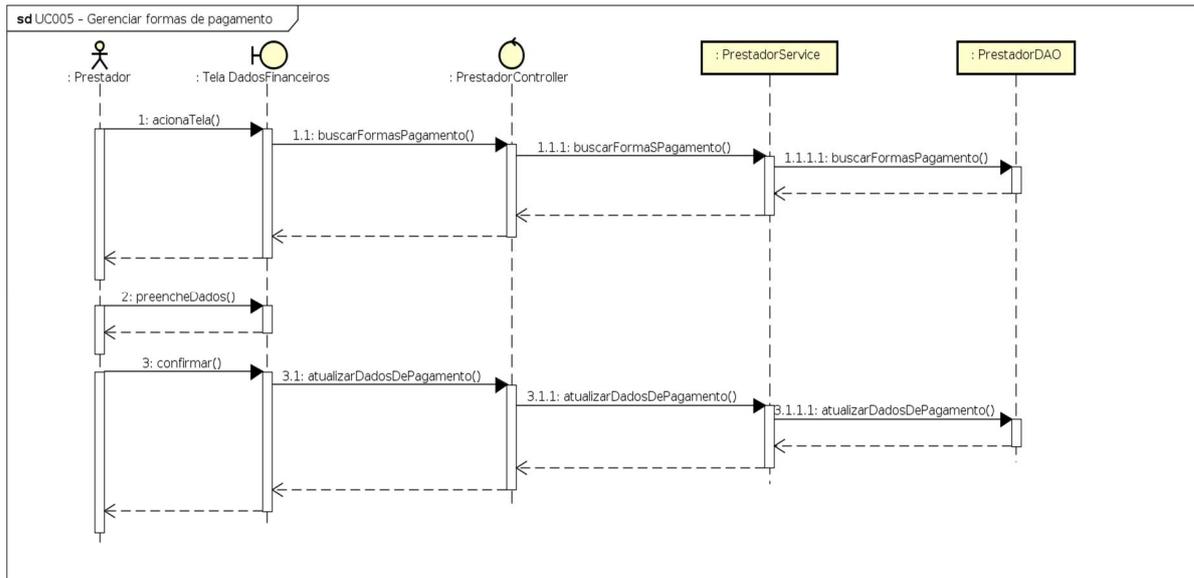
FONTE: Os autores (2020).

FIGURA 143 - GERENCIAR SERVIÇOS



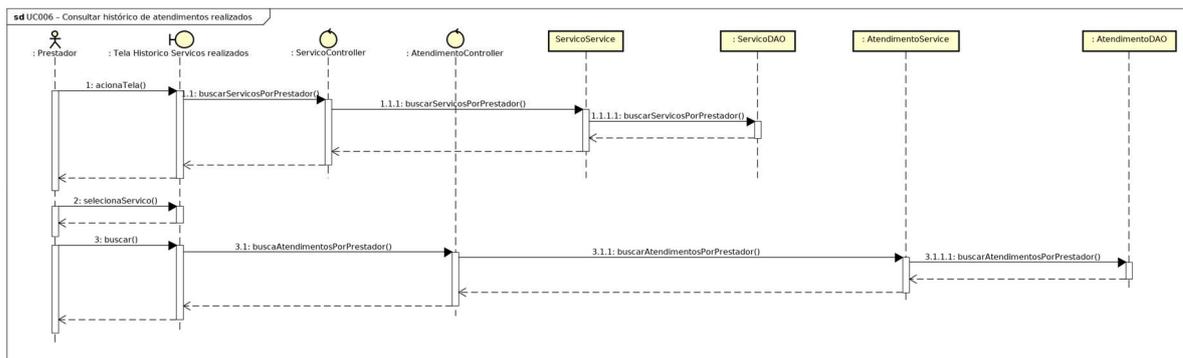
FONTE: Os autores (2020).

FIGURA 144 - GERENCIAR FORMAS DE PAGAMENTO



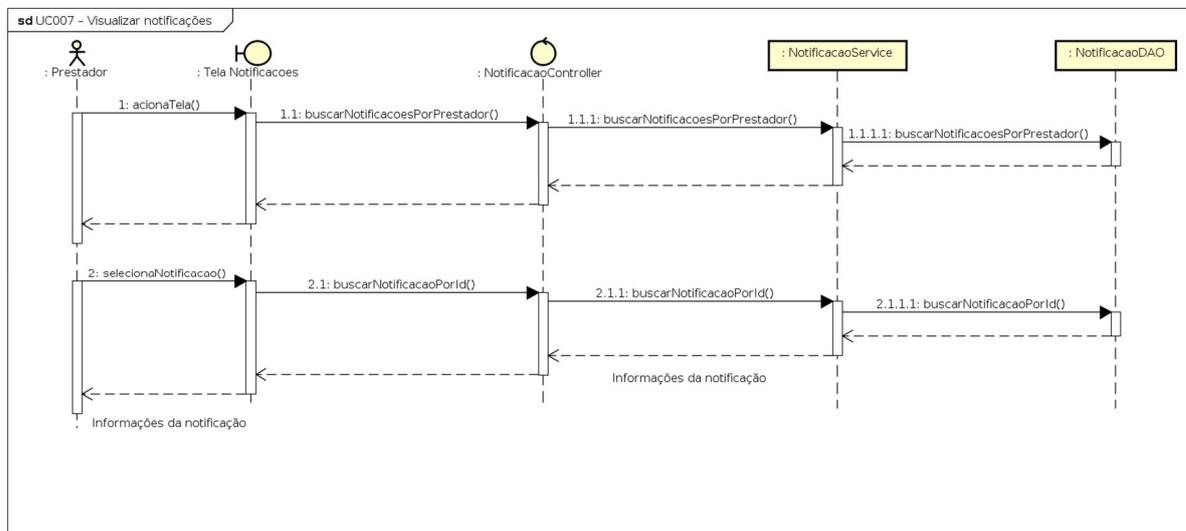
FONTE: Os autores (2019).

FIGURA 145 - CONSULTAR HISTÓRICO DE ATENDIMENTOS REALIZADOS



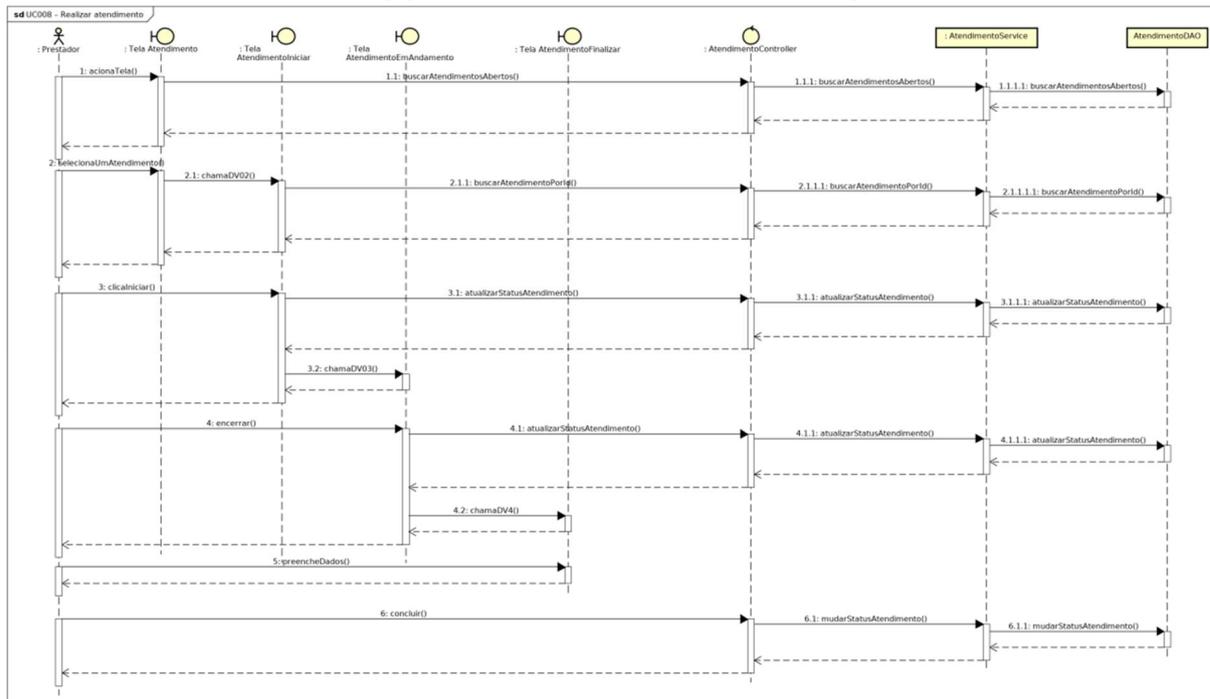
FONTE: Os autores (2020).

FIGURA 146 - VISUALIZAR NOTIFICAÇÕES



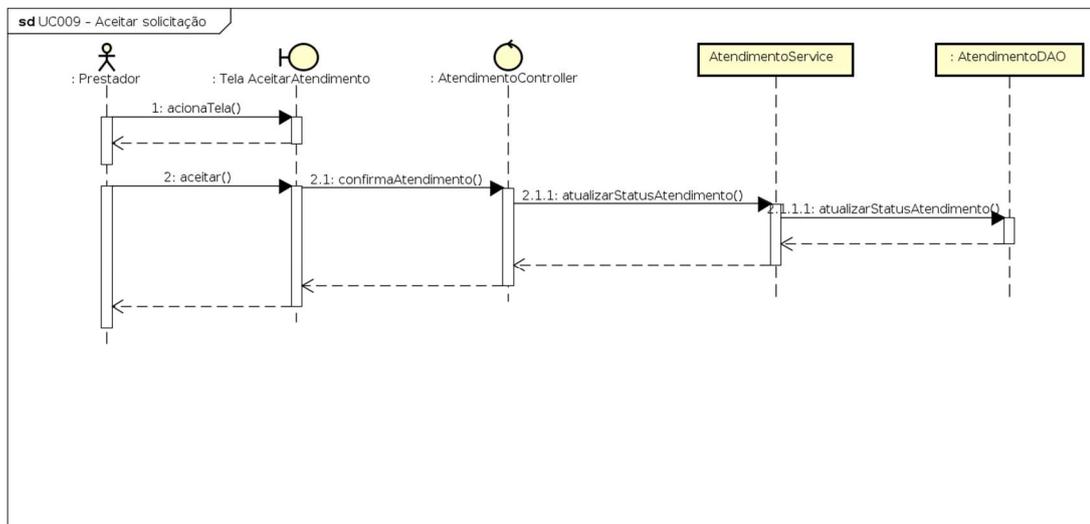
FONTE: Os autores (2020).

FIGURA 147 - REALIZAR ATENDIMENTO



FONTE: Os autores (2020).

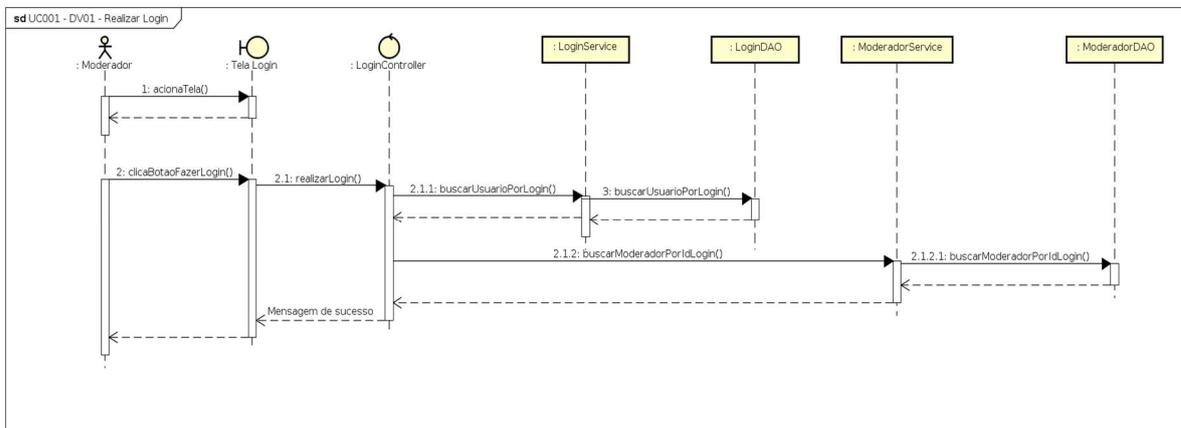
FIGURA 148 - ACEITAR SOLICITAÇÃO DE ATENDIMENTO



FONTE: Os autores (2020).

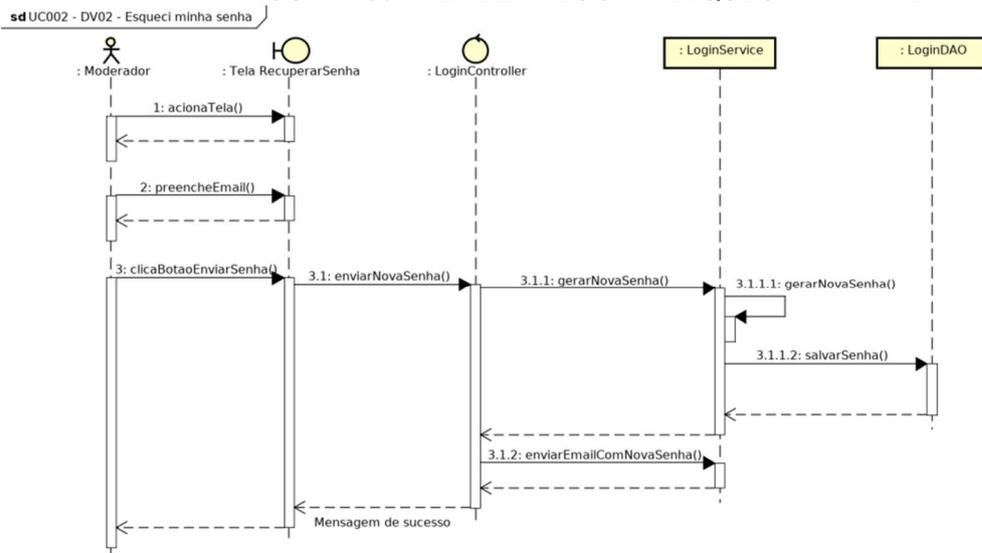
APÊNDICE M – DIAGRAMAS DE SEQUÊNCIA MODERADOR

FIGURA 149 - REALIZAR LOGIN



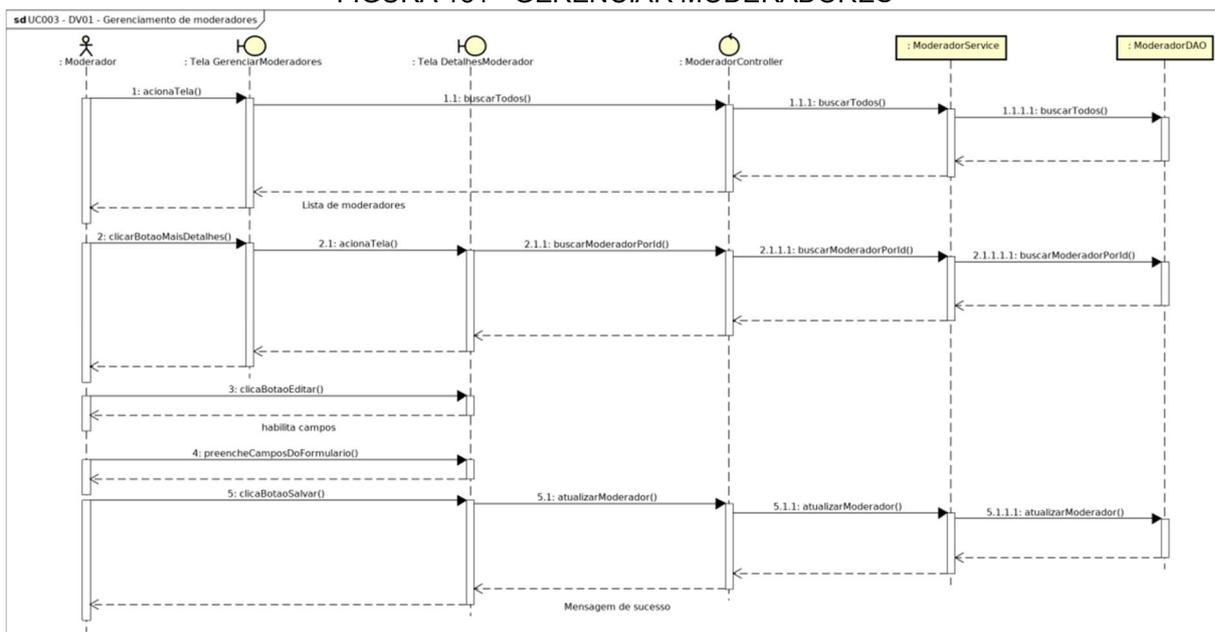
FONTE: Os autores (2020).

FIGURA 150 - REALIZAR LOGIN - ESQUECI MINHA SENHA



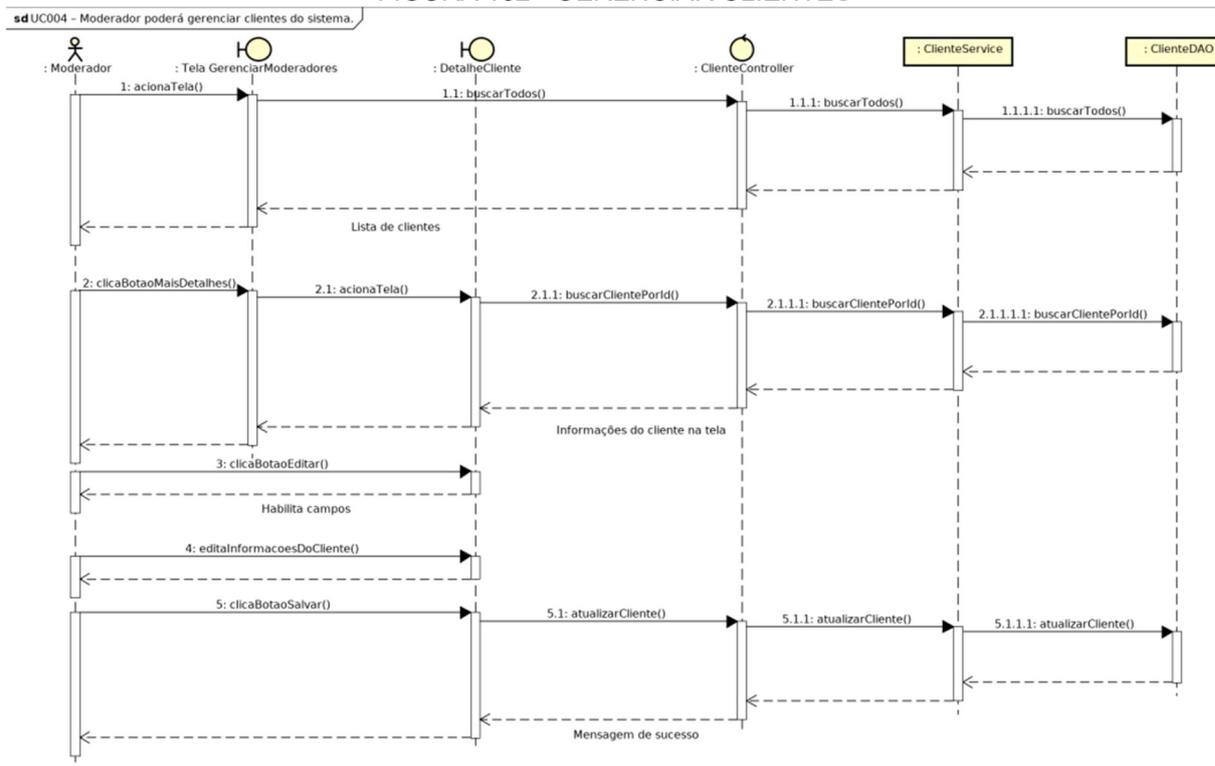
FONTE: Os autores (2020).

FIGURA 151 - GERENCIAR MODERADORES



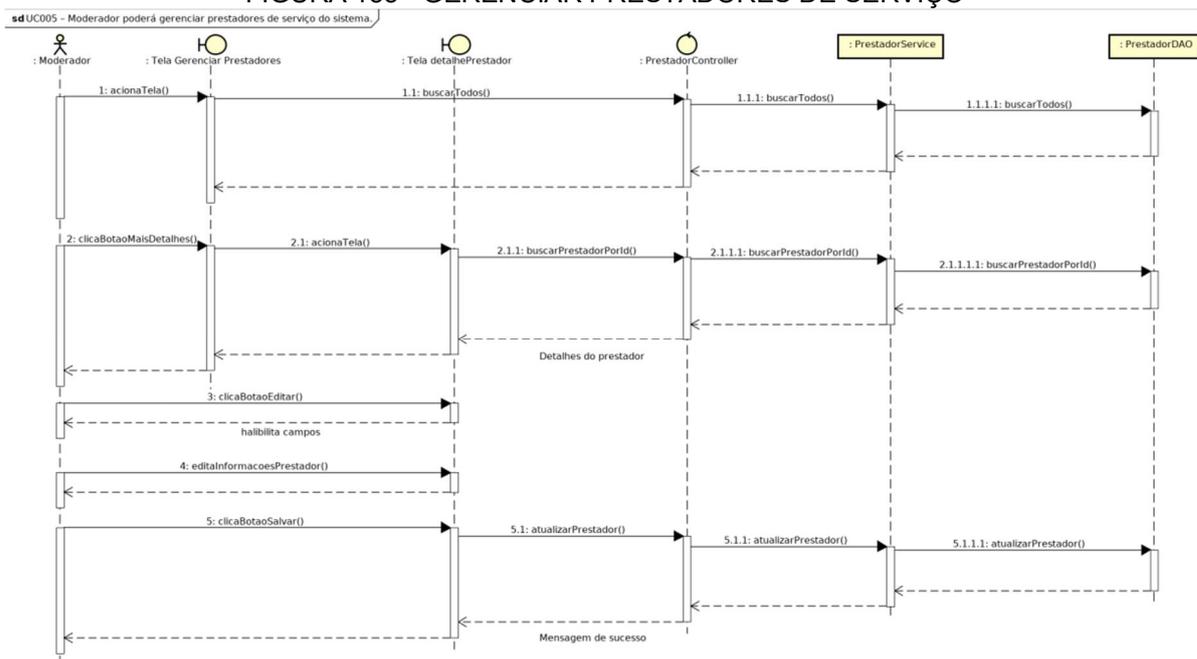
FONTE: Os autores (2020).

FIGURA 152 - GERENCIAR CLIENTES



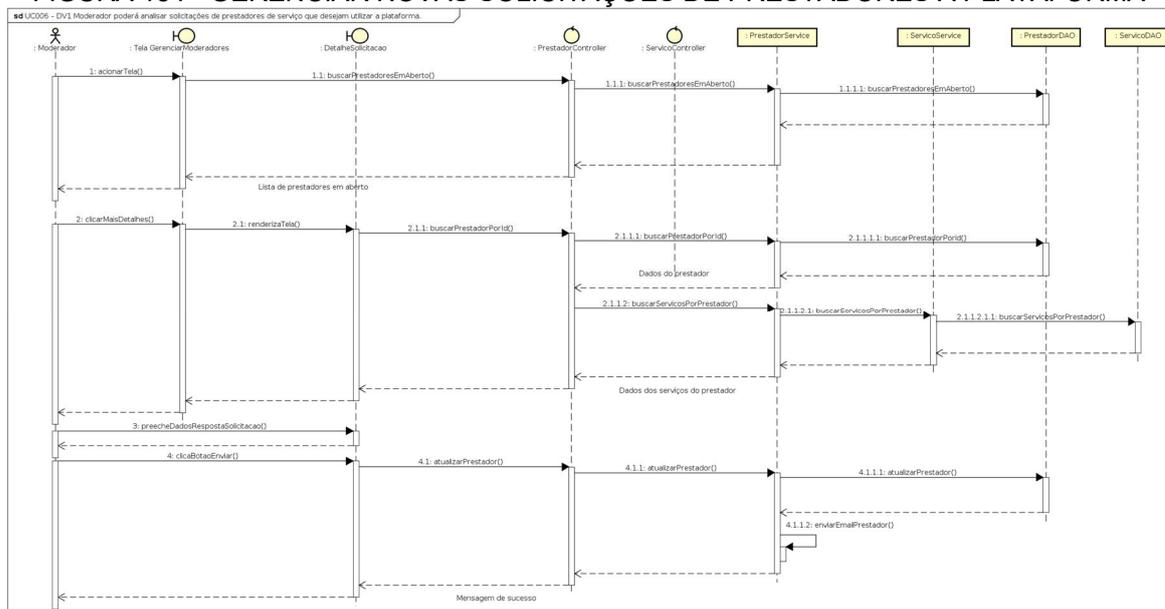
FONTE: Os autores (2020).

FIGURA 153 - GERENCIAR PRESTADORES DE SERVIÇO



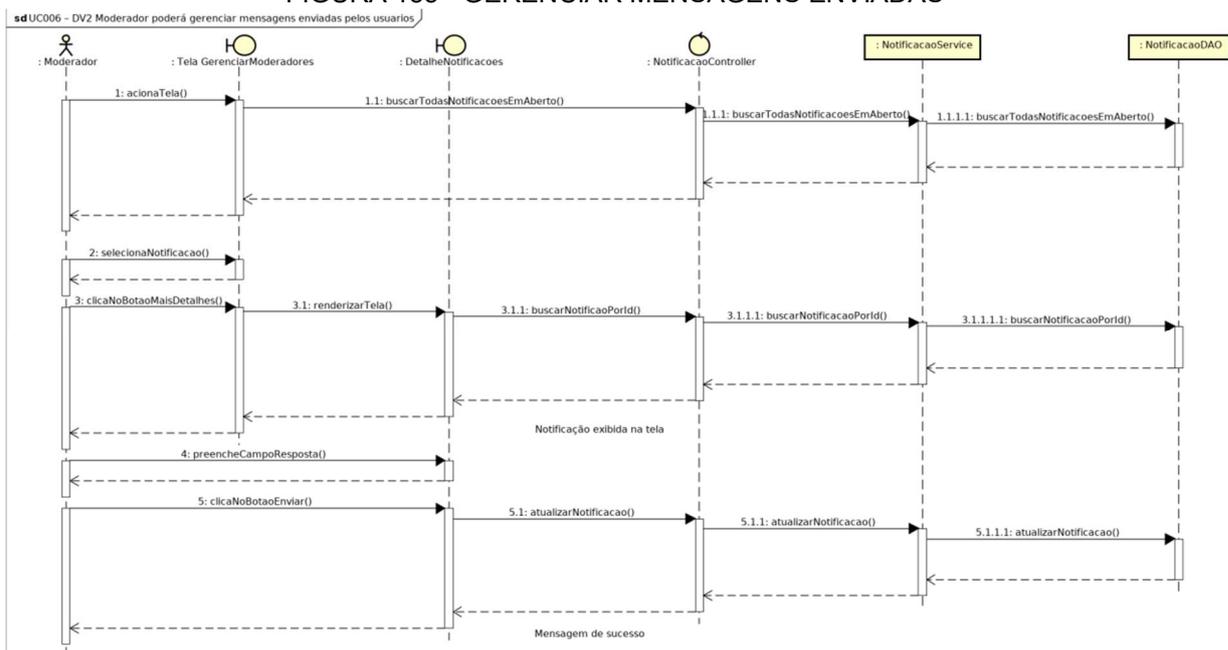
FONTE: Os autores (2020).

FIGURA 154 - GERENCIAR NOVAS SOLICITAÇÕES DE PRESTADORES À PLATAFORMA



FONTE: Os autores (2020).

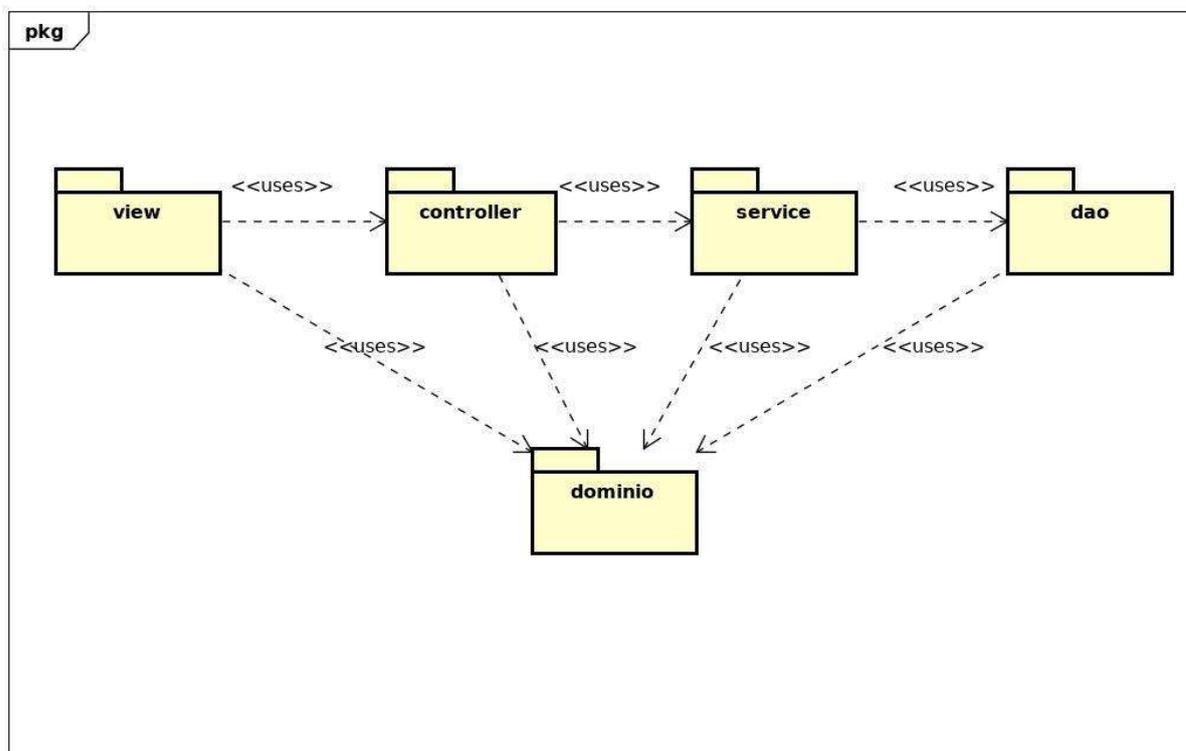
FIGURA 155 - GERENCIAR MENSAGENS ENVIADAS



FONTE: Os autores (2020).

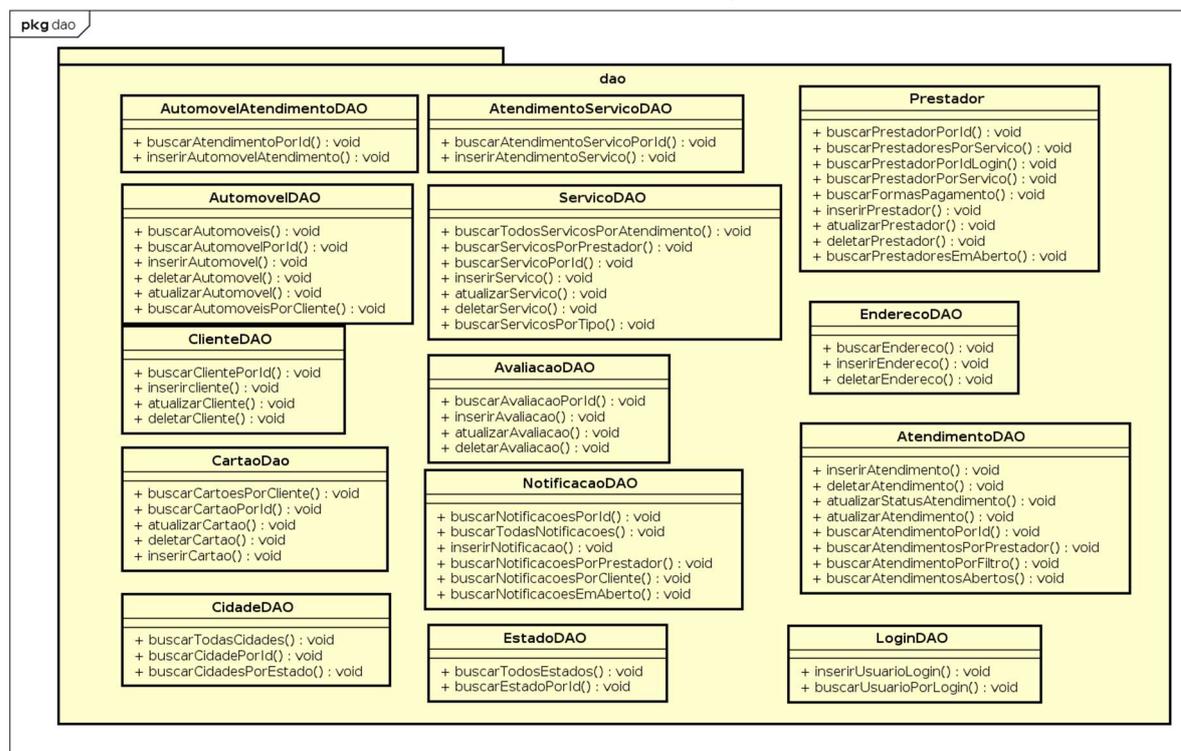
APÊNDICE N - DIAGRAMAS DE PACOTES

FIGURA 156 - DIAGRAMA DE PACOTES



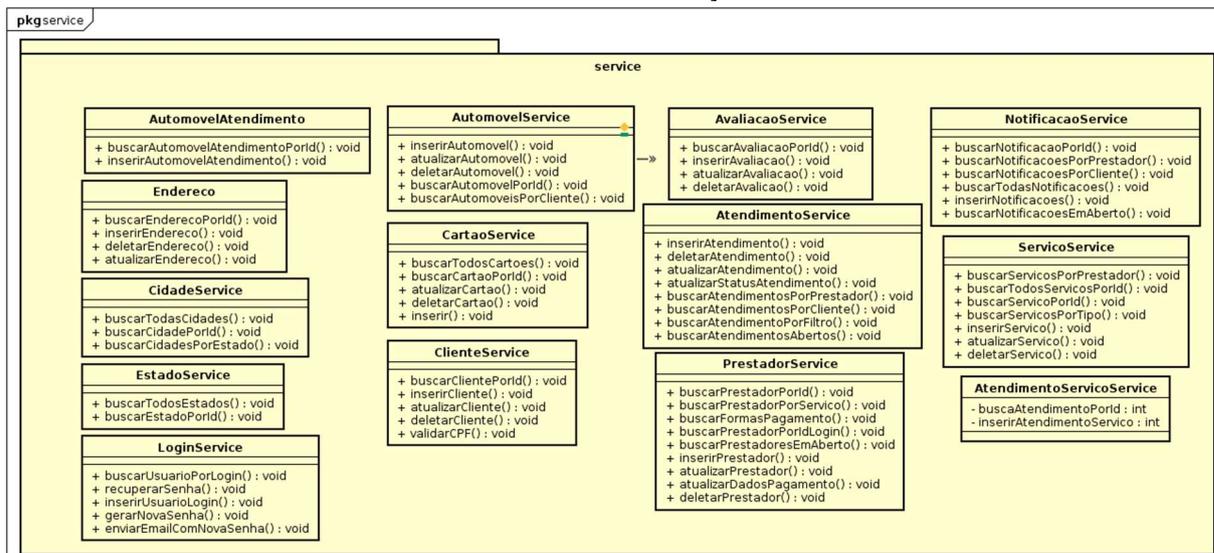
FONTE: Os autores (2020).

FIGURA 157 - DIAGRAMA DE IMPLEMENTAÇÃO DE PACOTES DAO



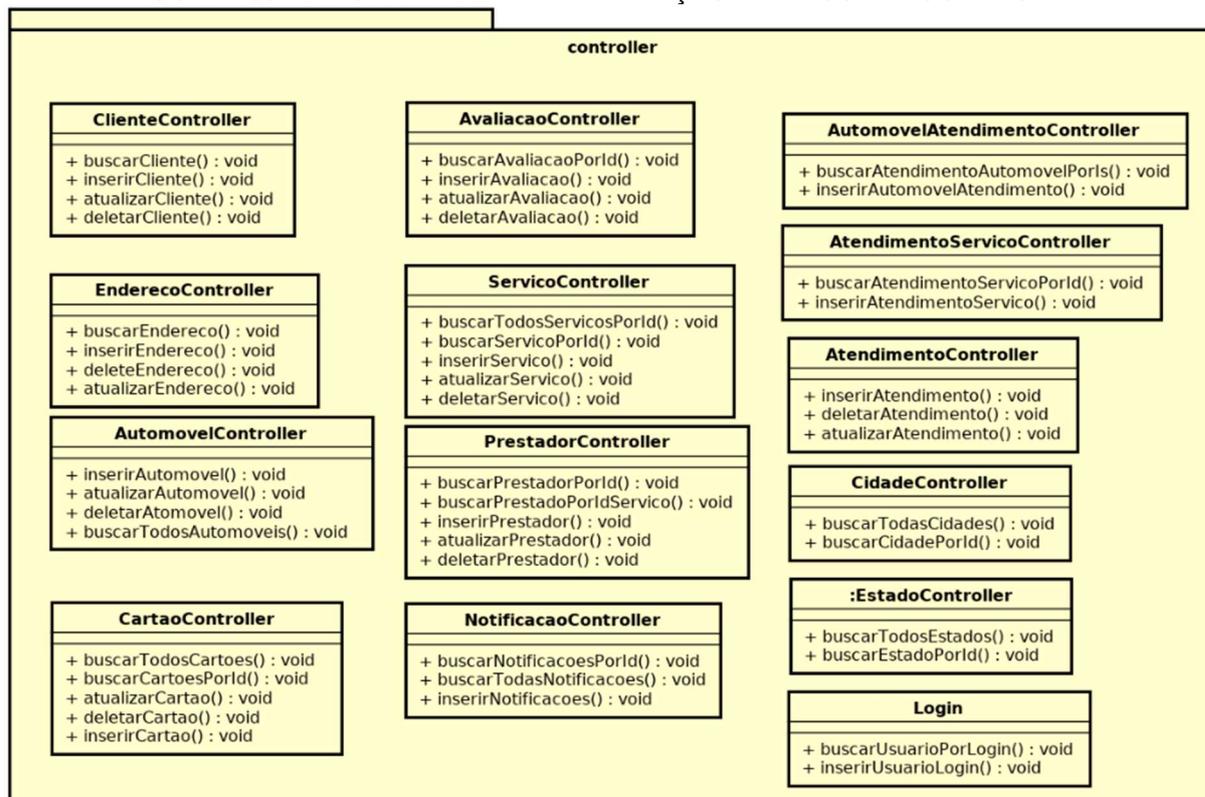
FONTE: Os autores (2020).

FIGURA 158 - DIAGRAMA DE IMPLEMENTAÇÃO DE PACOTES SERVICE



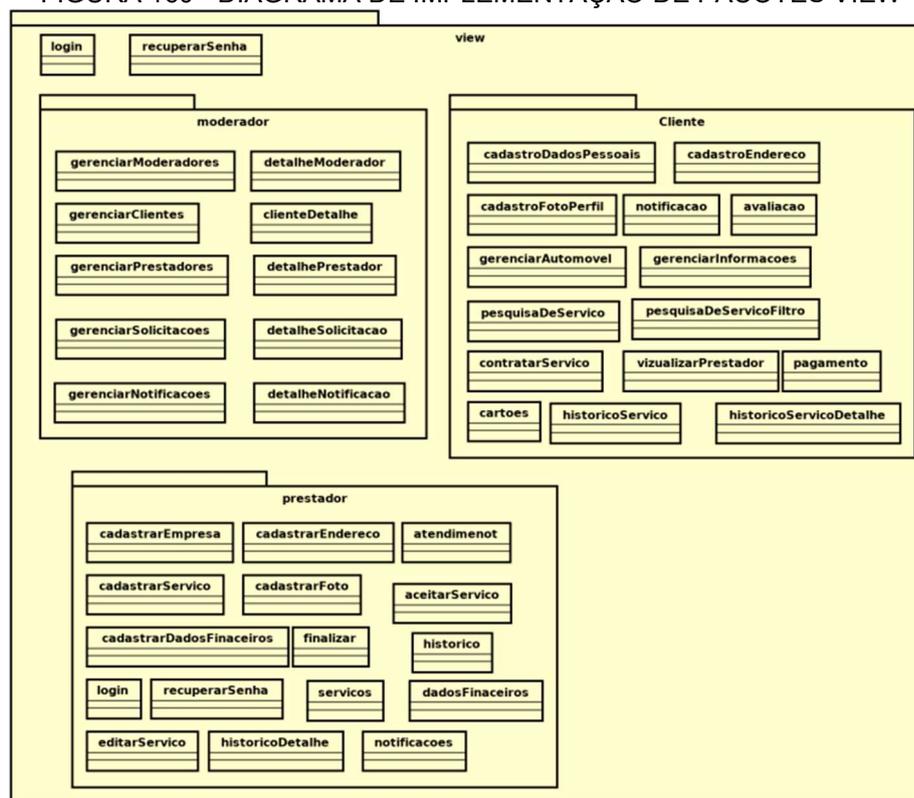
FONTE: Os autores (2020).

FIGURA 159 - DIAGRAMA DE IMPLEMENTAÇÃO DE PACOTES CONTROLLER



FONTE: Os autores (2020).

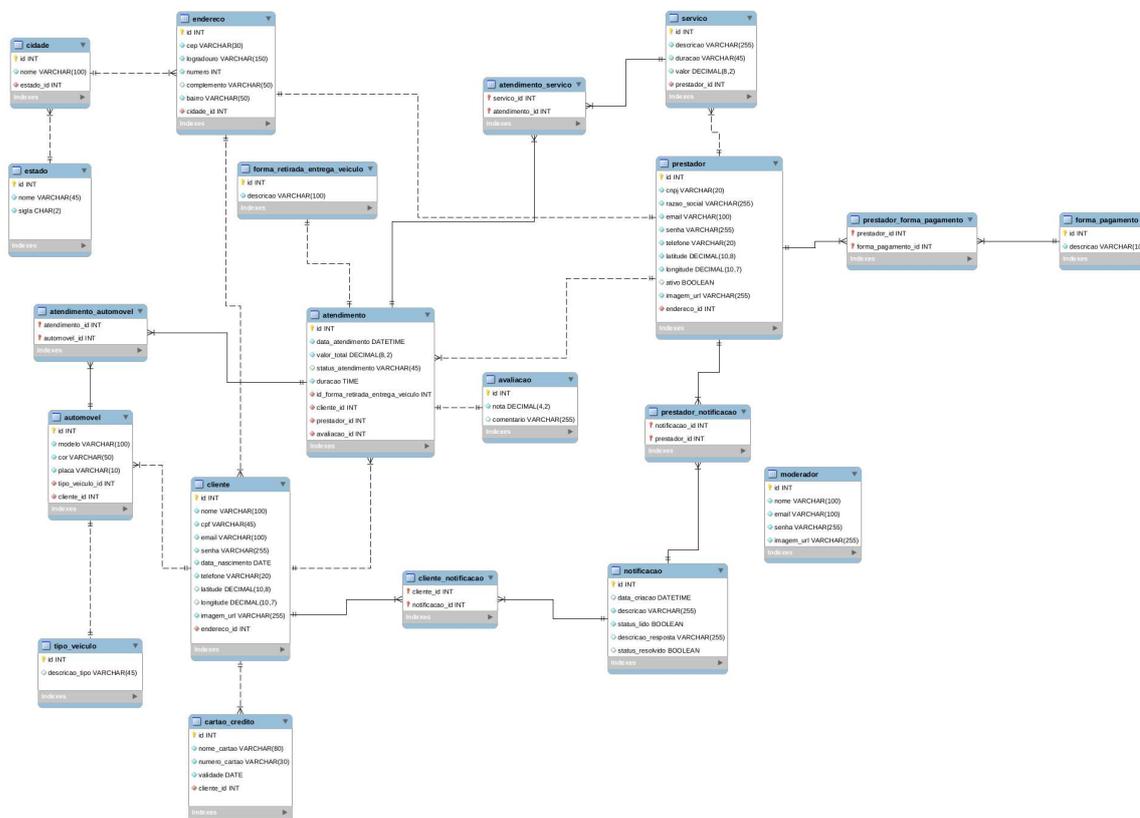
FIGURA 160 - DIAGRAMA DE IMPLEMENTAÇÃO DE PACOTES VIEW



FONTE: Os autores (2020).

APÊNDICE O – MODELO FÍSICO DO BANCO DE DADOS

FIGURA 161 - MODELO FÍSICO DO BANCO DE DADOS



FONTE: Os autores (2020).