

UNIVERSIDADE FEDERAL DO PARANÁ

ALEXANDRE BENATTO CADAN

FELIPE DE ALMEIDA

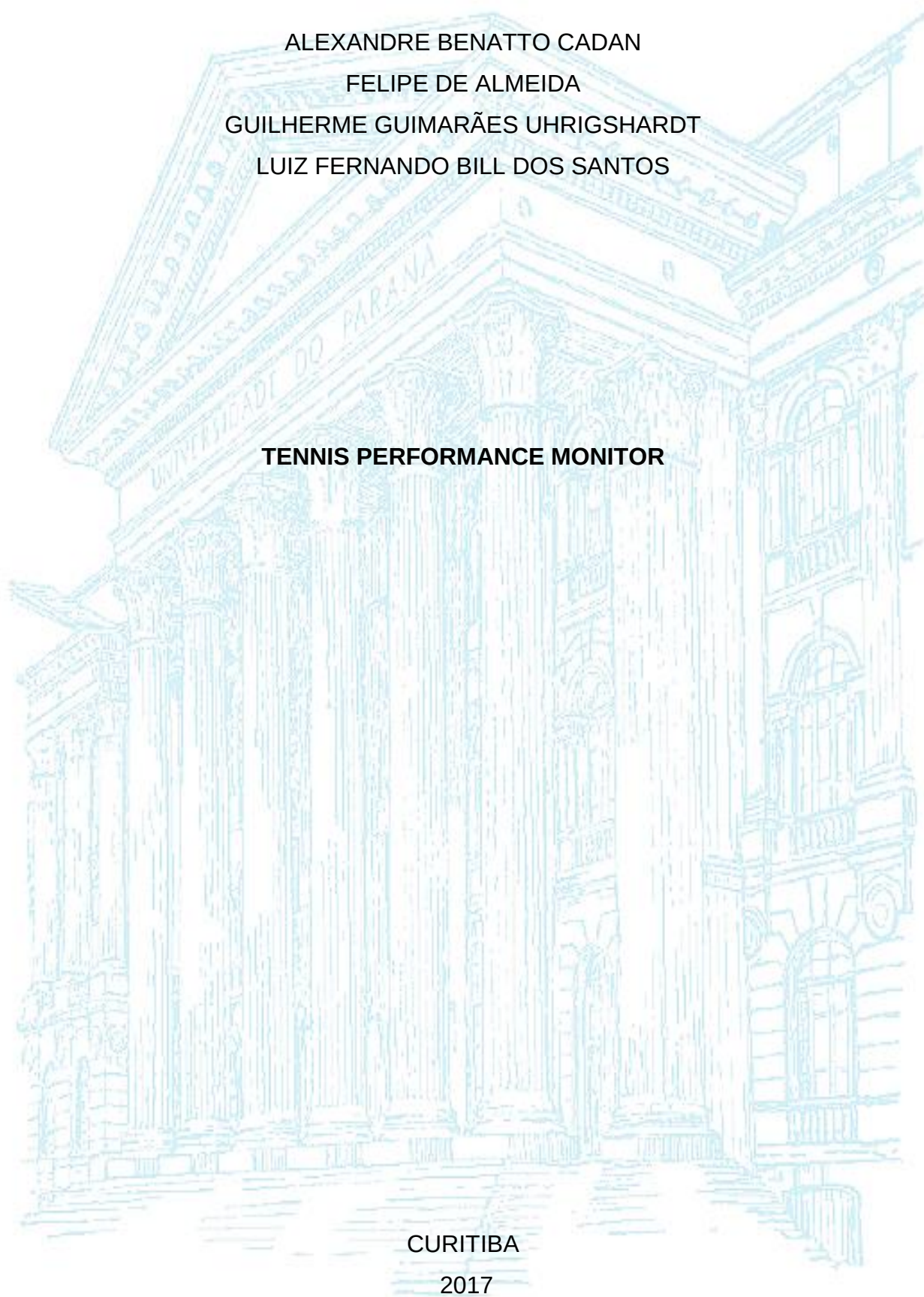
GUILHERME GUIMARÃES UHRIGSHARDT

LUIZ FERNANDO BILL DOS SANTOS

TENNIS PERFORMANCE MONITOR

CURITIBA

2017



UNIVERSIDADE FEDERAL DO PARANÁ

ALEXANDRE BENATTO CADAN

FELIPE DE ALMEIDA

GUILHERME GUIMARÃES UHRIGSHARDT

LUIZ FERNANDO BILL DOS SANTOS

TENNIS PERFORMANCE MONITOR

Trabalho de conclusão de curso apresentado como requisito para obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas, do Setor de Educação Profissional e Tecnológica, da Universidade Federal do Paraná.

Orientador: Dr. Razer Anthom Nizer Rojas Montañó.

CURITIBA

2017

TERMO DE APROVAÇÃO

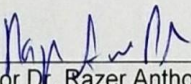
Guilherme Uhrigshardt
Alexandre Benatto Cadan
Felipe de Almeida
Luiz Bill dos Santos

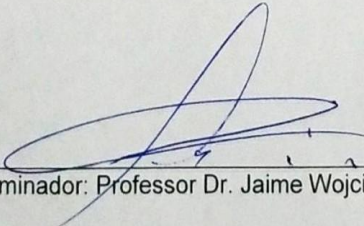
TENNIS PERFORMANCE MONITOR

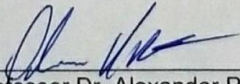
Trabalho apresentado como
requisito parcial para a obtenção do
título de Tecnólogo em Análise e
Desenvolvimento de Sistemas da
Universidade Federal do Paraná.

Curitiba, 08 de Dezembro de 2017.

BANCA EXAMINADORA


Orientador: Professor Dr. Razer Anthom Nizer Rojas Montañó
SEPT/UFPR


Examinador: Professor Dr. Jaime Wojciechowski
SEPT/UFPR


Examinador: Professor Dr. Alexander Robert Kutzke
SEPT/UFPR

RESUMO

A melhor performance em atividades físicas é cada vez mais procurada. Estatísticas, métodos e conceitos são utilizados para aumentar o número de acertos, corrigir possíveis falhas e acrescentar maior qualidade na saúde humana. Com o conceito tecnológico em constante crescimento é possível ajustar essas metodologias para alcançar o melhor nível possível, seja em agilidade, precisão, impulsão ou apenas bem-estar. Visando proporcionar essa qualidade, foi desenvolvido o aplicativo TPM – *Tennis Performance Monitor*, que utiliza concepções para monitorar a precisão em jogadas de partidas de Tênis, estatísticas de cada rodada (apresentando ao usuário o placar e característica de cada ponto realizado ou perdido), inscrições em campeonatos e comparação de desempenho com seus adversários. Tudo isso de uma maneira descomplicada para quem utiliza o sistema, apenas assinalando cada ponto o tipo final de seu golpe. Conjuntamente foi desenvolvido o sistema web, que utiliza os conceitos do aplicativo, para o usuário *premium* que quer gerenciar sua academia, torneio ou evento, podendo obter relatórios com os dados e estatísticas de cada participante.

ABSTRACT

Athletes evermore often seek their physical limits and performance tuning. Statistics, methods and concepts are used to refine the benefits of physical activities, refine movements and increase the health quality of humans. As the technological concept grows constantly, it is possible to adjust these methodologies to achieve a highest level possible in many aspects like agility, precision, impulse or even just welfare. Aiming to provide these kinds of resources the TPM - Tennis Performance Monitor application was developed, using concepts to monitor the players accuracy during the match, stats of all matches in a tournament (including score and stats charts) and also near tournaments organized by category. All of it can be achieved in a very simple way, with just two touches on the screen, informing which player won the point and how respectively. In order to manage matches and tournaments information, a web system was developed in parallel, and can be used by premium users that wishes to promote their tennis facilities and tournaments. These kinds of users will benefit from reports to better understand their target audience and increase their profit.

LISTA DE FIGURAS

FIGURA 1 – QUADRA DE TÊNIS.....	16
FIGURA 2 – BOLA.....	17
FIGURA 3 – RAQUETE.....	17
FIGURA 4 – JUÍZES.....	19
FIGURA 5 – FASES DO SCRUM.....	23
FIGURA 6 – PADRÃO MVC.....	32
FIGURA 7 – EXEMPLO TRELLO.....	33
FIGURA 8 – SISTEMA DE CONTROLE DE VERSÃO LOCAL.....	34
FIGURA 9 – SISTEMA DE CONTROLE DE VERSÃO CENTRALIZADA.....	34
FIGURA 10 – SISTEMA DE CONTROLE DE VERSÃO DISTRIBUÍDO.....	35
FIGURA 11 – WBS SISTEMA TPM.....	48
FIGURA 12 – GRÁFICO DE GANTT.....	49
FIGURA 13 – FLUXOGRAMA DO PROJETO.....	63
FIGURA 14 – BACKLOG DA SPRINT 1.....	65
FIGURA 15 – BACKLOG DA SPRINT 2.....	66
FIGURA 16 – BACKLOG DA SPRINT 3.....	67
FIGURA 17 – BACKLOG DA SPRINT 4.....	68
FIGURA 18 – BACKLOG DA SPRINT 5.....	69
FIGURA 19 – BACKLOG DA SPRINT 6.....	70
FIGURA 20 – FLUXO DE ACESSO AO SISTEMA.....	71
FIGURA 21 – TELA DE LOGIN DO APLICATIVO TPM.....	72
FIGURA 22 – TELA DE CADASTRO DO APLICATIVO TPM.....	73
FIGURA 23 – TELA DE ARBITRAR TORNEIO DO APLICATIVO TPM.....	73
FIGURA 24 – TELA DE ARBITRAR PARTIDA DO APLICATIVO TPM.....	74
FIGURA 25 – TELA DE MONITORAR PARTIDA DO APLICATIVO TPM.....	74
FIGURA 26 – TELA DE MONITORAR ACERTOS DO APLICATIVO TPM.....	75
FIGURA 27 – TELA DE MONITORAR ERROS DO APLICATIVO TPM.....	75
FIGURA 28 – TELA DE VISUALIZAR TORNEIOS DO APLICATIVO TPM.....	76
FIGURA 29 – TELA DE VISUALIZAR CONQUISTAS DO APLICATIVO TPM.....	76
FIGURA 30 – TELA DE VISUALIZAR ESTATÍSTICAS DO APLICATIVO TPM.....	77
FIGURA 31 – TELA DE LOGIN DO SISTEMA WEB.....	78
FIGURA 32 – TELA DE CADASTRO DO SISTEMA WEB.....	78
FIGURA 33 – TELA DE LOGIN DO SISTEMA WEB.....	78
FIGURA 34 – TELA DE INICIAL DO ADMINISTRADOR DO SISTEMA WEB.....	79
FIGURA 35 – TELA DE GERENCIAMENTO DE USUÁRIOS DO SISTEMA WEB.....	80
FIGURA 36 – TELA DE RELATÓRIOS ADMINISTRATIVO DO SISTEMA WEB.....	80
FIGURA 37 – TELA DE CADASTRO DE LOCAIS DO SISTEMA WEB.....	81
FIGURA 38 – TELA DE VISUALIZAR TORNEIOS DO SISTEMA WEB.....	81
FIGURA 39 – TELA DE EDITAR TORNEIOS DO SISTEMA WEB.....	82
FIGURA 40 – TELA DE CONTATO DO SISTEMA WEB.....	82
FIGURA 41 – TELA DE VISUALIZAÇÃO DA CHAVE DO SISTEMA WEB.....	83

LISTA DE QUADROS

Quadro 1 – VISÃO GERAL DO CRONOGRAMA DO PROJETO.....	39
Quadro 2 – CRONOGRAMA DE GERENCIAMENTO DO PROJETO.....	39
Quadro 3 – CRONOGRAMA DA DOCUMENTAÇÃO.....	40
Quadro 4 – CRONOGRAMA DA SPRINT 1.....	40
Quadro 5 – CRONOGRAMA DA SPRINT 2.....	41
Quadro 6 – CRONOGRAMA DA SPRINT 3.....	41
Quadro 7 – CRONOGRAMA DA SPRINT 4.....	42
Quadro 8 – CRONOGRAMA DA SPRINT 5.....	42
Quadro 9 – CRONOGRAMA DA SPRINT 6.....	43
Quadro 10 – VISÃO GERAL DO CRONOGRAMA DO PROJETO.....	43
Quadro 11 – CRONOGRAMA DE GERENCIAMENTO DO PROJETO.....	44
Quadro 12 – CRONOGRAMA DA DOCUMENTAÇÃO.....	44
Quadro 13 – CRONOGRAMA DA SPRINT 1.....	44
Quadro 14 – CRONOGRAMA DA SPRINT 2.....	45
Quadro 15 – CRONOGRAMA DA SPRINT 3.....	45
Quadro 16 – CRONOGRAMA DA SPRINT 4.....	46
Quadro 17 – CRONOGRAMA DA SPRINT 5.....	46
Quadro 18 – CRONOGRAMA DA SPRINT 6.....	47
Quadro 19 – TABELA DE PRECEDÊNCIA.....	47
Quadro 20 – REQUISITOS FUNCIONAIS DE USUÁRIO – SISTEMA WEB.....	50
Quadro 21 – REQUISITOS FUNCIONAIS DE USUÁRIO PREMIUM – WEB.....	51
Quadro 22 – REQUISITOS FUNCIONAIS DE ADMINISTRADOR – WEB.....	52
Quadro 23 – REQUISITOS FUNCIONAIS DE USUÁRIO – APP.....	53
Quadro 24 – REQUISITOS NÃO FUNCIONAIS.....	54
Quadro 25 – PLANO DE RISCOS.....	55
Quadro 26 – TABELA DE RESPONSABILIDADES.....	56

SUMÁRIO

1 INTRODUÇÃO	11
1.1 Objetivos Gerais.....	12
1.1.1 Demanda Encontrada	12
1.1.2 Softwares Existentes	13
1.2 Objetivos Específicos	13
2. FUNDAMENTAÇÃO TEÓRICA.....	14
2.1 História do Tênis	14
2.1.1 Torneios	14
2.2 Regras.....	16
2.3 Objetivos do Tênis.....	19
2.4 Termos do Tênis	20
2.5 Metodologias de Desenvolvimento.....	22
2.5.1 Scrum.....	22
2.5.1.1 Equipe Scrum.....	23
2.5.1.2 Product Owner	24
2.5.1.3 Scrum Master	24
2.5.1.4 Sprint.....	24
2.5.1.4.1 Planejamento de Sprint	25
2.5.1.4.2 Reuniões Diárias	25
2.5.1.4.3 Revisão da Sprint	25
2.5.1.4.4 Retrospectiva da Sprint	26
2.5.1.4.5 Backlog da Sprint	26
2.5.2 WBS	26
2.5.6 Gantt	27
2.6 Interface	27
2.6.1 Design Responsivo	27
2.6.2 Ergonomia.....	27
2.6.3 Usabilidade	28
2.7 Modelagem	28
2.7.1 UML – Unified Modeling Language	29
2.7.1.1 Diagramas Comportamentais.....	29
2.7.1.2 Diagramas Estruturais	29

2.8	Armazenamento de Dados.....	30
2.8.1	SQL Server.....	30
2.8.2	Data Access Object (DAO).....	30
2.8.3	Web Services	31
2.9	Implementação.....	31
2.9.1	MVC (Model, View, Controller).....	31
2.9.2	Trello	32
2.9.3	Versionamento	33
2.9.3.1	Sistemas de Controle de Versões Locais.....	33
2.9.3.2	Sistemas de Controle de Versões Centralizadas	34
2.9.3.3	Sistemas de Controle de Versões Distribuídas	35
2.10	Entity Framework.....	35
3	METODOLOGIA.....	36
3.1	Tecnologias	36
3.2	Modelo de Processo de Engenharia de Software	37
3.3	Plano de Atividade	38
3.3.1	Cronograma	38
3.3.1.1	Cronograma Planejado	38
3.3.1.2	Cronograma Realizado	43
3.3.2	Tabela de Precedência	47
3.3.3	WBS.....	48
3.3.4	Gráfico de Gantt.....	49
3.3.5	Requisitos Funcionais e Requisitos Não Funcionais.....	50
3.4	Plano de Riscos	55
3.5	Responsabilidades.....	56
3.6	Materiais.....	57
3.6.1	Recursos de Hardware.....	57
3.6.2	Recursos de Software	59
3.6.2.1	Softwares de Desenvolvimento.....	59
3.6.2.2	Softwares de Documentação	62
3.7	Desenvolvimento do Projeto	63
3.7.1	Definição do backlog do produto.....	64
3.7.2	Divisão do backlog do produto em sprints.....	64

3.7.3	Sprints realizadas.....	64
3.7.3.1	Sprint 1.....	65
3.7.3.2	Sprint 2.....	66
3.7.3.3	Sprint 3.....	67
3.7.3.4	Sprint 4.....	68
3.7.3.4	Sprint 5.....	69
3.7.3.5	Sprint 6.....	70
4	APRESENTAÇÃO DO SOFTWARE	71
4.1	Interfaces e Funcionalidades.....	71
4.1.1	Aplicativo	72
4.1.2	Portal Web.....	78
4.2	Perfis	83
4.2.1	Perfil Usuário Comum	83
4.2.2	Perfil de Usuário Premium.....	84
4.2.3	Perfil de Administrador	84
5	CONSIDERAÇÕES FINAIS	85
5.1	Conclusão	85
5.2	Implementações Futuras	86
5.2.1	Notificações.....	87
5.2.2	Adaptações para novas modalidades esportivas	87
5.2.3	Desenvolver em outras plataformas.....	87
5.2.4	Criação de ranking de usuários.....	87
5.3	Agradecimentos	88

1 INTRODUÇÃO

De acordo com uma pesquisa realizada pela Fundação Getúlio Vargas e publicada pelo canal de notícias Estadão, até o fim de 2017 o Brasil terá um *smartphone* por habitante e há hoje no país quatro computadores a cada cinco cidadãos. Isto é, a tecnologia está cada vez mais abrangente em nosso cotidiano. Na rua, em casa, nas instituições acadêmicas e principalmente no trabalho, estamos sempre nos conectando ao mundo por meio aplicativos, redes sociais ou portais de notícias.

Em um mundo onde as pessoas estão periodicamente ligadas à celulares, *notebooks*, *tablets*, *smart watches* e aparelhos musicais inteligentes, não é difícil que a tecnologia seja utilizada para diferentes propósitos como solução para algumas situações. No esporte, por exemplo, ela está cada vez mais presente desempenhando um papel importante, ajudando os mais diversos atletas, seja na preparação, na divulgação de torneios e resultados ou nos equipamentos.

Exemplos de modalidades em que isso já é realidade não faltam. O canal de informações Raia Oito, popular entre os praticantes de natação, apresenta a evolução dos trajes, mostrando as roupas são feitas sob medida para o atleta, que passa por um escaneamento 3D do corpo para que não haja material excessivo o que poderia causar um maior atrito diminuindo sua performance. O material utilizado também é especial e tenta reproduzir as mesmas características da pele de animais aquáticos.

No *Rugby*, por exemplo, o árbitro pode utilizar o auxílio das câmeras durante os jogos para revisar suas decisões, permitindo uma disputa mais justa.

Em competições amadoras, nem sempre podemos contar com a tecnologia ao nosso favor. Para inscrever-se em torneios de tênis em clubes filiados à Federação Paranaense de Tênis por exemplo, o atleta precisa passar por um processo longo e nada agradável. Todo ano a Federação divulga em seu portal uma lista enorme de todos os torneios que serão disputados no ano, divididos por categoria. Nesta lista é possível ver detalhes de onde o torneio será disputado como endereço e telefone. Para solicitar a inscrição o jogador deve ligar no clube ou se locomover até ele. Cada clube então armazena as informações dos inscritos de uma forma diferente, seja em fichas de papel ou planilhas do *Excel*.

Com a notável procura por modernização em modalidades esportivas, o desejo em obter informações precisas e rápidas está cada vez mais presente nos dias atuais. O volume de dados com informações e estatísticas de competições, têm alterado a forma de enxergar qualidades, defeitos, melhorias e declínios em atletas profissionais e amadores em relação à sua performance. Para praticantes de tênis, não poderia ser diferente. Além daqueles que querem aumentar o nível de jogo para participar em torneios de nível mais elevado, ainda existe os que jogam para manter uma boa saúde (combatendo doenças cardiovasculares, buscando a redução de peso ou ampliando a qualidade de vida e bem-estar).

Aplicativos e sites de diversas plataformas têm facilitado as tarefas corriqueiras do ser humano. No mundo do tênis não é diferente, a maioria daqueles que tem essa modalidade como uma atividade regular possui acesso à essas tecnologias e poderiam se beneficiar com sistemas voltados à eles.

1.1 Objetivos Gerais

O Aplicativo TPM – *Tennis Performance Monitor* proporcionará auxílio aos árbitros nas pontuações e estatísticas conforme o andamento do jogo, facilitará o acesso as informações de pontuação e erros durante partidas, e também à torneios e partidas simples. Hoje em dia, tais facilidades estão presentes apenas em eventos de grande porte, em grandes torneios, assim excluindo atletas amadores, que muitas vezes desejam acompanhar seu rendimento, pontos fracos e sua evolução como um todo. Essa escassez de informações pode afetar o crescimento do atleta, podendo levar até mesmo ao abandono do esporte.

1.1.1 Demanda

Durante um torneio de tênis, muitos atletas desejam ver e analisar suas estatísticas de cada partida, assim como nas partidas de jogadores profissionais onde nos intervalos das partidas são exibidos pela televisão. Para atingir tal objetivo, era necessário que todos os pontos de uma partida fossem registrados, assim como a forma que cada um foi finalizado. Outra dificuldade encontrada por muitos tenistas

amadores é encontrar adversários do mesmo nível. Constatou-se então que essas demandas poderiam ser atendidas com a criação de um aplicativo para celulares.

1.1.2 Softwares Existentes

Após a pesquisa nas lojas “Play Store” e “AppStore”, encontramos alguns aplicativos com funcionalidades semelhantes ao TPM. Alguns permitem ao usuário contar a pontuação de partidas, como o “Tennis Score Keeper”, e outros permitiam localizar outros jogadores e torneios como “Smash! Tennis”. Apesar disso, nenhum deles unificava todas as funcionalidades descritas até aqui.

1.2 Objetivos Específicos

Este trabalho é composto pelos objetivos específicos:

- Seleção das tecnologias de desenvolvimento;
- Análise de negócio para criação de banco de dados;
- Definição das regras de segurança da informação;
- Prototipação de um layout agradável ao usuário;
- Desenvolvimento de um portal web para gerenciamento por um administrador;
- Desenvolvimento de um aplicativo para smartphones;

2 FUNDAMENTAÇÃO TEÓRICA

Serão expostas ao longo dos tópicos as tecnologias utilizadas, bem como as técnicas e as pesquisas realizadas para o desenvolvimento do projeto. As obras, artigos e buscas utilizadas para consulta também farão parte dessa documentação, sendo destacada como referências.

2.1 História do Tênis

O jogo de tênis originou-se na Inglaterra, no século XIX. Muitos historiadores, porém, acreditam que a origem do esporte tenha sido na França no século XII, além disso, usava-se as mãos para bater na bola. No final do século XVI começou-se a usar raquetes para bater na bola e o jogo passou a ser chamado de "tênis", do francês antigo "Tenez", que pode ser traduzido como "receba" ou "segure". O jogo era popular nos dois países, onde era praticado em locais cobertos, onde a bola batesse em uma parede. Mais tarde, o jogo seria chamado de "tênis real".

O Tênis Real evoluiu consideravelmente nos últimos três séculos. No século XVI, a luva se torna uma raquete, e o jogo tinha-se movido para uma área fechada, as regras começaram a ser estabilizadas. O Tênis Real começou a ganhar popularidade na realeza europeia, alcançando grande fama ainda no século XVI.

2.1.1 Torneios

O Torneio de tênis de Wimbledon é o mais antigo torneio de tênis do mundo, e é considerado o de maior prestígio. Foi criado em 1877, quando o All England Lawn Tennis and Croquet Club organizou a primeira competição da história do tênis, com um total de 22 participantes. Ao torneio de simples homens (onde um jogador enfrentava outro) vieram se acrescentar as duplas em 1879 e o simples feminino em 1884. May Sutton, dos Estados Unidos, foi a primeira vencedora não-britânica, ganhou o torneio feminino em 1905. Dois anos depois, Norman Brookes, da Austrália, ganhou o campeonato simples masculino.

O Grand Slam é a sequência dos quatro principais torneios do mundo: Roland Garros (França), Wimbledon (Inglaterra), Open dos EUA (Estados Unidos) e Open da Austrália. O termo veio de uma associação com outro esporte, o baseball. Quando uma equipe consegue completar, de uma só vez, as quatro bases, ela provoca um "*grand slam*" (grande barulho) no estádio — ninguém se cansa de aplaudir a incrível jogada. No tênis, quem vence os quatro maiores torneios consegue um fato tão raro quanto completar as quatro bases sucessivas no baseball, daí a comparação. O maior vencedor de Grand Slams é o suíço Roger Federer (ainda em atividade) com 18 títulos. Vencer os quatro Grand Slams em uma única temporada é um fato muito raro. Dois homens e três mulheres conseguiram essa proeza: John Donald Budge (Estados Unidos), em 1938 e Rod Laver (Austrália), em 1962 e 1969 no torneio masculino e Maureen Connolly (Estados Unidos), em 1953, Margaret Smith Court (Austrália), em 1970 e Steffi Graf (Alemanha), em 1988 no torneio feminino.

Em 21 de Maio de 1881, a United States National Lawn Tennis Association (USNLTA) foi fundada para padronizar as regras e organizar as competições. O United States National Men's Singles Championship, hoje o US Open de tênis, foi realizado pela primeira vez em 1881, em Newport, Rhode Island. O torneio feminino se estabeleceu em 1887. O esporte também era popular na França, com o Open da França datando de 1891. Assim, Wimbledon, o US Open, o Open da França e o Open da Austrália (de 1905) tornaram-se os eventos de maior prestígio no tênis, algo que se mantém até hoje. Juntos, esses torneios são chamados de *Majors* ou *Slams*.

As regras mais abrangentes foram divulgadas em 1924, pela Federação Internacional das Regras de Tênis, hoje conhecida como Federação Internacional de Tênis, e se mantém ativa até os dias atuais, com a principal alteração sendo a adição do *tie-break*. No mesmo ano, o esporte foi retirado dos Jogos Olímpicos, tendo retornado somente em 1984, como um evento promocional. O sucesso do evento foi tão grande que o Comitê Olímpico Internacional (COI) decidiu reintroduzir o tênis como um esporte olímpico na edição de 1988, em Seul. A Copa Davis, a principal competição internacional entre as federações nacionais, existe desde 1900. A versão análoga para as mulheres é a Fed Cup, iniciada em 1963.

2.2 Regras

Nessa seção serão apresentadas as principais regras do tênis com algumas figuras ilustrando seus respectivos tópicos:

- A quadra;

Uma quadra oficial de tênis deve medir 23,77 metros de comprimento por 10,97 metros de largura para as partidas de duplas e de 8,23 metros de largura para o jogo de simples. A rede deve ter altura de 1,06 metros e a área de saque onde a bola deve pingar na execução de um saque é de 6,40 metros de comprimento por 4,11 metros de largura. Apresentado na FIGURA 1.

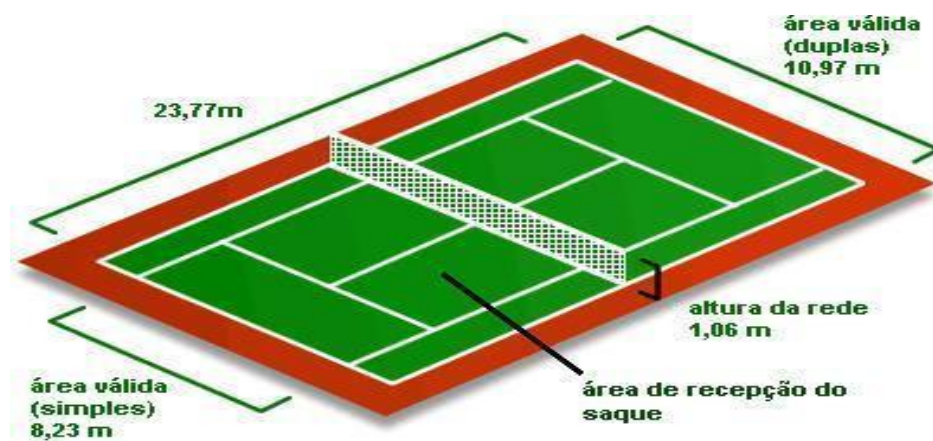


FIGURA 1 - QUADRA DO TÊNIS
LIVRO: Almanaque dos Esportes, Editora Europa, 2009.

Carpete: Piso rápido feito de material sintético.

Saibro: Piso de terra batida, que amortece a bola e torna o jogo mais lento.

- A bola;

A bola de tênis deve ser branca ou amarela. Não podendo apresentar costuras. Diâmetro de 6,35 a 6,67 centímetros e seu peso deve ser maior do que 56,7 e menor do que 58,5 gramas. Apresentado na FIGURA 2.



FIGURA 2 - BOLA
LIVRO: Almanaque dos Esportes, Editora Europa, 2009.

- A raquete;

A raquete de tênis deve ter sua superfície plana sendo permitido somente um padrão de cordas. O padrão deve ser uniforme..

A raquete deve ter o tamanho máximo de 73,66 centímetros no seu comprimento, e também não ultrapassar os 31,75 centímetros de largura total.

A sua superfície das cordas não deve ultrapassar os 39,37 centímetros no seu comprimento e os 29,21 centímetros na sua largura. Apresentado na FIGURA 3.



FIGURA 3 - RAQUETE

LIVRO: Almanaque dos Esportes, Editora Europa, 2009.

- Juízes;

Em uma partida de tênis há no total cerca de 12 juízes divididos em Juiz de cadeira, juiz de rede, juízes de linha laterais e central, juízes de serviço e juízes de saque. Abaixo, você verá quem é cada um deles e quais as suas funções. Observe a figura acima com as explicações abaixo.

Juiz de cadeira: É responsável por comandar a partida. Ele anuncia o placar e tem o poder de reverter as decisões dos demais árbitros. Aplica penalizações, como advertência, perda de pontos e suspensão.

Juiz de rede: Fiscaliza se a bola toca ou não na rede durante um saque. Um dispositivo eletrônico pode auxiliá-lo.

Juiz de saque: São dois em quadra. São responsáveis por cuidar se o saque dado pelo tenista adversário não saiu da linha de saque.

Juiz de serviço: São dois. Observam se o atleta não pisa na linha na hora do saque.

Juízes de linha: São quatro os juízes de linha, dois de cada lado da quadra. Eles são responsáveis por identificar se o saque entrou ou não na área de serviço e orientam o juiz de cadeira em relação as faltas.

Juiz central de linha: São dois também. Observam se a bola atinge o lado correto da quadra na hora do saque.

Juiz de fundo de quadra: Ele pode suspender o jogo por causa do mau tempo, por exemplo. Além disso, pode desclassificar um atleta por conduta antidesportiva.

Pegador de bola: É o responsável por apanhar as bolas na quadra e alcançá-las aos tenistas.



FIGURA 4 - JUÍZES
LIVRO: Almanaque dos Esportes, Editora Europa, 2009.

2.3 Objetivos do tênis

No tênis é preciso fazer com que a bolinha passe por cima da rede e toque dentro dos limites da quadra adversária. Os jogadores devem rebater a bolinha em direção ao adversário antes ou depois dela pingar uma única vez. Se a bolinha quicar duas vezes, parar na rede ou for rebatida para fora da quadra, o ponto será concedido ao adversário. Um sorteio define quem inicia a partida. *Ace*: Um serviço em que o recebedor falha para devolver ou não toca. O sacador vence o ponto imediatamente.

2.4 Termos do Tênis

No tênis é possível rebater a bolinha de maneiras diversas, como por exemplo:

- Ace: Um serviço em que o recebedor falha para devolver ou não toca. O sacador vence o ponto imediatamente.
- Backhand: Golpe de fundo de quadra executado do lado contrário onde o jogador segura a raquete. Para um canhoto, por exemplo, o backhand fica no seu lado direito.
- Bola na linha: Se a bola tocar na linha é considerada dentro da quadra. Portanto é ponto.
- Bye: Passagem direta para a segunda rodada de um torneio. Isto acontece quando não há jogadores suficientes, sendo bye os melhores ranqueados.
- Drop Shot: Ou "Deixadinha". Bola curta, que pinga próxima à rede.
- Dupla Falta: Saque fora da área de serviço duas vezes seguida, o que resulta na perda do ponto pelo sacador.
- Game: Os pontos são contados em 0, 15, 30, 40 e um game. Se houver empate por 40 a 40, vence quem fizer dois pontos seguidos.
- Forehand: Golpe de fundo (depois que a bola quica uma vez na quadra) executado do mesmo lado do corpo em que o jogador segura a raquete. O forehand de um canhoto, por exemplo, é o golpe de fundo no seu lado esquerdo.
- Partida: A partida é dividida em sets, games e pontos. Vence quem ganhar dois sets. Algumas competições realizam partidas com melhor de cinco sets, ou seja, ao invés de vencer dois sets, o jogador precisa vencer três.

- **Set:** O atleta que ganhar seis *games*, com uma diferença mínima de dois games em relação ao adversário, vence o set. Se o placar ficar em 5 a 5, dois novos games são jogados. Com novo empate, joga-se o *tiebreak*.
- **Saque (ou serviço):** Os saques são alternados a cada game. Quem saca tem duas chances para acertar o serviço. Cada serviço, o tenista deve lançar a bola por cima da rede e atingir o solo dentro da área de saque diagonal oposta. Se o adversário não conseguir rebater a bola, o sacador ganha o ponto chamado de “ace”. Já quando um jogador vence o game enquanto o outro está sacando, ocorre a conhecida “quebra de serviço”. Em cada saque, ou serviço, o jogador deve se posicionar alternadamente do lado direito e esquerdo da quadra, começando sempre o game pelo lado direito.
- **Tiebreak:** É o desempate do set. Nele, cada jogada vale um ponto. Após um tenista sacar a primeira bola, os jogadores alternam dois saques. Quem fizer sete pontos ou mais, sendo eles dois de diferença, vence o set.
- **Troca de lado:** Os jogadores devem trocar de lado no fim do primeiro, do terceiro e de cada game ímpar subsequente de cada set e no final de cada set, a menos que o total de games naquele set seja par, caso em que a troca não é feita até o fim do primeiro game do set seguinte.
- **Voleio (Volley):** Golpe executado antes que a bola quique. Se for acima da cabeça, como o saque, trata-se de *smash*.
- **Winner:** Com exceção do saque, golpe ganhador, onde o adversário não consegue tocar a bola. É mais apropriado para bolas batidas do fundo da quadra.

2.5 Metodologias de Desenvolvimento

Na elaboração de um projeto de desenvolvimento, a escolha da metodologia a ser utilizada pode trazer uma economia considerável de tempo, esforço e recursos. Há diversas ferramentas que auxiliam nesse processo.

2.5.1 *Scrum*

Nessa seção será explicada a metodologia que será aplicada para desenvolvimento do projeto

O *Scrum* é uma metodologia para desenvolvimento e manutenção de sistemas complexos e adaptativos. Sendo um modelo de desenvolvimento ágil de software, que segundo Steffen (2012), “É uma nova forma de gestão e desenvolvimento de software que usa uma abordagem de planejamento e execução iterativa e incremental”. De acordo com o Beck et al. (2001) este modelo possui quatro premissas:

- Indivíduos e interações acima de processos e ferramentas;
- Software funcional acima de documentação compreensiva;
- Colaboração do cliente acima de negociação contratual;
- Resposta a mudanças acima de seguir um plano.

Este modelo tem como características aumentar o esforço no desenvolvimento e minimizar na documentação. Com entregas frequentes, que normalmente são de uma a quatro semanas, chamadas *sprints*, o cliente acaba se tornando parte da equipe de desenvolvimento, o que diminui os custos do projeto (CONH, FORD, 2003). A evolução do projeto e da equipe é discutida diariamente, no lugar de reuniões longas e semanais ele adota reuniões rápidas, objetivas e diárias. Essas devem começar sempre no mesmo horário e responder as três perguntas a seguir:

- O que fiz ontem?
- O que estou planejando para fazer amanhã?
- Existe algo me impedindo de realizar minha meta?

Segundo Schwaber e Sutherland (2013), todo o trabalho no *scrum* é realizado em iterações chamadas de *sprints*.

A FIGURA 5 mostra como funciona o *scrum* e onde o *product backlog* (requisitos do sistema obtidos anteriormente) é dividido em pequenas partes. Essas pequenas partes (*sprint backlog*) engloba o conteúdo que será desenvolvido naquela *sprint*, que pode durar de uma a quatro semanas. Dentro da *sprint* acontecem reuniões diárias que são chamadas de *daily meeting*.

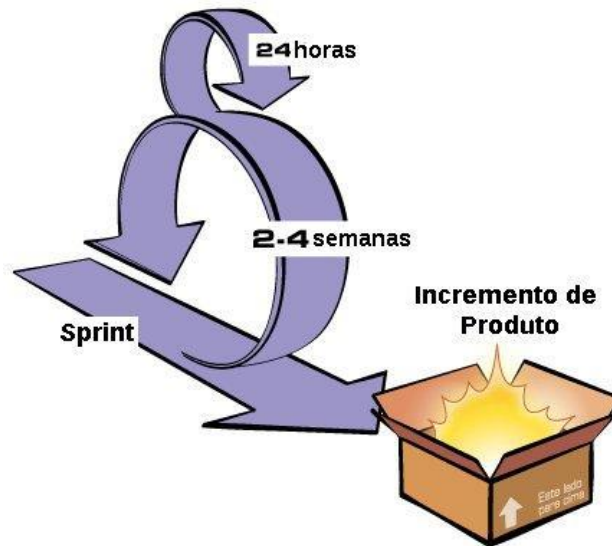


FIGURA 5 – FASES DO SCRUM
FONTE: CISNEIRO, 2009

2.5.1.1 Equipe Scrum

Composto pelo *Project Owner*, o Time de Desenvolvimento e *Scrum Master*. Tem como característica serem multifuncionais, com liberdade para escolher a melhor forma de completar seu trabalho, ao invés de serem dirigidos por outros indivíduos. Os membros do time devem possuir todas as competências necessárias para completar o trabalho, sem depender de pessoas de fora da equipe.

2.5.1.2 *Product Owner*

O *Product Owner* é o cliente de um time *scrum*. Ele deve definir todas as funcionalidades que o software deve possuir (*product backlog*), definir e priorizar o *sprint backlog* que serão incluídos em cada *sprint*, de acordo com o valor de negócio e aceitar ou rejeitar as entregas realizadas pelo (SCHWABER e SUTHERLAND, 2013).

O *product backlog* ou *backlog* do produto é uma única lista que contém todos os itens necessários para o desenvolvimento ou incremento do produto (*software*). Nele estão contidos os requisitos não funcionais e funcionais, mudanças, e bugs encontrados durante os testes. O *product owner* é o responsável pela criação dessa lista. De acordo com Schwaber e Sutherland (2013), este artefato está em constante mudança, uma vez que seu objetivo é manter um registro concreto de todas as funcionalidades, requisitos, melhorias e atributos que envolvem esse novo incremento do sistema (SCHWABER e SUTHERLAND, 2013).

2.5.1.3 *Scrum Master*

O *Scrum Master* objetiva assegurar que a equipe siga as práticas do *scrum* e que não se comprometa excessivamente com relação a aquilo que é capaz de realizar durante uma *sprint*, atuando como um gerente de projeto ou líder técnico. Ele também conduz o *daily meeting* e torna-se responsável por remover obstáculos que sejam apresentados pela equipe durante essas reuniões (LIBARDI, 2010).

2.5.1.4 *Sprint*

Sprint é o espaço de tempo que dura entre duas a quatro semanas em que um produto de valor e usável é criado. Ao final de uma *sprint* outra sempre é iniciada, até que não haja mais *backlogs* no projeto. No decorrer das semanas que englobam uma *sprint*, obrigatoriamente ocorrem quatro eventos (SCHWABER e SUTHERLAND, 2013), são eles:

- Planejamento de sprint (*sprint planning*);

- Reunião diária (*daily meeting*);
- Revisão da *sprint*;
- Retrospectiva da *sprint*.

2.5.1.4.1 Planejamento de *Sprint*

O planejamento de *sprint* é a primeira atividade a ser realizada. Ele é uma reunião onde estão presentes todos os envolvidos no projeto: *scrum team*, *scrum master* e *product owner*. Durante essa reunião o *product owner* descreve as funcionalidades de maior prioridade para a equipe, essa por sua vez, faz perguntas até que não restem mais dúvidas para que possam quebrar essas funcionalidades em tarefas técnicas após a reunião Schwaber e Sutherland (2013).

Após as definições das tarefas técnicas (backlog do produto) o *product owner* e o time de *scrum* priorizam um objetivo para um *sprint*, o sucesso será avaliado ao fim na retrospectiva (SCHWABER e SUTHERLAND, 2013).

2.5.1.4.2 Reuniões Diárias

As reuniões diárias também são conhecidas como *daily meeting*, acontecem todos os dias no mesmo horário e devem ter uma duração máxima de 15 minutos. São realizadas nos primeiros minutos da jornada de trabalho. O objetivo dessa técnica é disseminar o conhecimento sobre o que foi feito no dia anterior, identificar problemas que possam estar acontecendo e priorizar no que vai ser feito no dia que se inicia (SCHWABER e SUTHERLAND, 2013).

2.5.1.4.3 Revisão da *Sprint*

A revisão da *sprint* é realizada sempre que o ciclo da *sprint* termina, ela acontece em uma reunião onde o time *scrum* apresenta o que foi alcançado com seu trabalho para o *product owner*, gerência e clientes. Os itens avaliados durante essa prática são os que foram definidos como requisitos durante o *sprint planning* (SCHWABER e SUTHERLAND, 2013).

2.5.1.4.4 Retrospectiva da *Sprint*

A retrospectiva da *sprint* assim como a revisão, ocorre ao final de um ciclo, ela serve para identificar o que funcionou bem, o que pode ser melhorado e quais as ações necessárias para isso. “Na reunião de retrospectiva da *sprint* é feita a revisão do processo de desenvolvimento, de forma a torná-lo eficaz e gratificante para a próxima iteração” (SCHWABER, SUTHERLAND, 2013).

2.5.1.4.5 *Backlog* da *Sprint*

Os *backlogs* das *sprints* são as tarefas retiradas do *backlog* do produto pelo time de *scrum* para que sejam realizadas em uma *sprint*. Ele é escolhido de acordo com as prioridades definidas pelo *product owner*, assim como a percepção do tempo que será necessário para realizá-las. O *sprint backlog* não pode possuir itens pendentes ao final de uma *sprint*, ele é acompanhado e atualizado diariamente pelo *scrum master* com o intuito de mostrar quais as tarefas que já foram completadas e as que ainda não. Esse acompanhamento é de extrema importância, pois com ele pode-se realizar mudanças do esforço dentro da equipe caso necessário (PEREIRA, 2007).

2.5.2 WBS

A WBS tem em vista diminuir a complexidade do projeto. Quando pensamos em um projeto como a construção de um avião, por exemplo, pensamos em algo monumental, a ideia de WBS é justamente ajudar a quebrar o projeto em partes, facilitando a compreensão do que deve ser feito em cada etapa. Com a WBS podemos prever atrasos do projeto caso o cliente atrase na entrega de algum requisito.

2.5.3 Gantt

O Diagrama de Gantt é um gráfico que visa mostrar os avanços do projeto em todas as etapas. Criado pelo engenheiro mecânico Henry Gantt em 1917, o gráfico é utilizado para controlar a produção de um determinado projeto. (Weaver, PATRICK, 2015)

2.6 Interface

Um dos componentes mais importantes de um software, é uma interface agradável de fácil entendimento do usuário que apresente um design adaptativo.

2.6.1 Design Responsivo

O design responsivo adapta o leiaute da aplicação à página, dependendo do tamanho da tela do dispositivo que renderiza ou a abre. Com o aumento da utilização de dispositivos móveis, houve a necessidade de adaptar as aplicações e web sites para que pudessem ser visualizados e utilizados nesse novo tipo de tecnologia. A partir desta nova premissa, foi criado o design responsivo (FRAN, 2012).

2.6.2 Ergonomia

Ergonomia é uma disciplina científica relacionada ao entendimento das relações entre os seres humanos com outros elementos ou sistemas, e a aplicação dos princípios ergonômicos bem como seus métodos nos projetos têm como objetivo otimizar o bem-estar humano e o desempenho global do sistema (IEA, 2015). Os ergonomistas contribuem para o planejamento, projeto e avaliação de tarefas, postos de trabalho, produtos, ambientes e sistemas de modo a torná-los compatíveis com as necessidades, habilidades e limitações das pessoas.

2.6.3 Usabilidade

Usabilidade é um atributo de qualidade que avalia quão fácil de usar uma interface é. Ela também faz referência para métodos que buscam melhorar e otimizar essa interface na fase de design, deixando-a mais intuitiva possível.

A usabilidade está distribuída à diversos elementos, sendo tradicionalmente associada, segundo Nielsen (1993), aos seguintes fatores:

- Facilidade de aprendizagem: o sistema deve ser fácil de assimilar pelo utilizador, para que este possa começar a trabalhar rapidamente;
- Eficiência: o sistema deve ser eficiente para que o utilizador, depois de aprender a usar, possa atingir uma boa produtividade;
- Facilidade de memorização: o sistema deve ser facilmente memorizado, para que depois de algum tempo sem o utilizar, o usuário lembre-se de como usá-lo;
- Segurança: o sistema deve prever erros, evitar que os utilizadores os cometam e, se o cometerem, permitir fácil recuperação ao estado anterior.
- Satisfação: o sistema deve ser usado de uma forma agradável, para que os utilizadores fiquem satisfeitos com a sua utilização.

2.7 Modelagem

A modelagem de um sistema computacional é de suma importância para o entendimento do problema apresentado em relação à solução proposta. Possui diversas formas de representação, tanto diagramas, quanto modelos gráficos.

2.7.1 - UML – Unified Modeling Language

A linguagem unificada de modelagem, UML ou *Unified Modeling Language* é uma linguagem gráfica de modelagem utilizada para visualizar, especificar, construir e documentar os artefatos de sistemas de objetos (GILLEANES, 2007). A UML não tem como objetivo definir como o trabalho deve ser desenvolvido e sim o que deve ser feito.

Ela é subdividida em duas categorias, a de diagramas estruturais e comportamentais.

2.7.1.1 - Diagramas Comportamentais

Os diagramas comportamentais são diagramas que dizem respeito à interação entre os elementos que compõe o sistema e seus comportamentos, como às regras de negócio envolvidas (GUEDES, 2009). Os diagramas de comportamento são: diagrama de caso de uso, diagrama de máquina de estados e diagrama de atividades. Em sua subcategoria "Interação" estão inclusos os diagramas de sequência, comunicação, visão geral de interação e por último, porém não menos importante o de temporização (GUEDES, 2009).

2.7.1.2 Diagramas Estruturais

Os diagramas estruturais dizem respeito à arquitetura do sistema. Tem como principais objetivos identificar entidades de dados, informações de interesse e componentes englobados pelo sistema (BEZERRA, 2015). Inclui diagrama de classe, diagrama de objeto, diagrama de componentes, diagrama de estrutura composta, diagrama de pacote e diagrama de utilização (FOWLER, 2014).

2.8 Armazenamento de Dados

A maneira como os dados são salvos e o acesso aos mesmos, deve ser feito de maneira rápida e segura, para isso existem diversas tecnologias e maneiras diferentes de fazê-los.

2.8.1 SQL Server

O SQL Server é um SGBD (Software Gerenciador de Banco de Dados) de fácil e simples utilização, essa ferramenta possui alto desempenho tanto em consultas SQL simples, quanto no desenvolvimento de bases de dados complexas.

2.8.2 *Data Access Object* (DAO)

O padrão de projeto DAO surgiu com a necessidade de separar a lógica de negócios da lógica de persistência de dados. Este padrão permite que possa ser mudada a forma de persistência sem que isso influencie em nada na lógica de negócio, além de tornar nossas classes mais legíveis. Classes DAO são responsáveis por trocar informações com o SGBD (Sistema Gerenciador de Banco de Dados) e fornecer operações CRUD (*Create, Read, Update e Delete*) e de pesquisas, elas devem ser capazes de buscar dados no banco e transformar esses em objetos ou lista de objetos, fazendo uso de listas genéricas, também deverão receber os objetos, converter em instruções SQL e mandar para o banco de dados. A interação é sempre feita com as bases de dados, nunca com as classes ou formulários. Se for aplicado o padrão corretamente, o mesmo abstrairá completamente o modo de busca e gravação dos dados, tornando isso transparente para aplicação e facilitando a manutenção na aplicação e migração de banco de dados. Também é possível centralizar a troca de dados com o SGBD, tendo um ponto único de acesso a dados, sendo assim a aplicação terá um ótimo design orientado a objeto (FOWLER, 2003).

2.8.3 Web Services

De acordo com o W3C (*World Wide Web Consortium*) um Web Service define-se como: “um sistema de software projetado para suportar a interoperabilidade entre máquinas sobre rede”. Hoje em dia é muito comum empresas oferecerem serviços para a comunicação entre elas e outras pessoas, sejam físicas ou jurídicas, internas ou externas a empresa. Alguns desses serviços podem ser automatizados com a utilização de Web Services. Por exemplo, em uma empresa que vende produtos, não é necessário existir um representante de vendas se o seu cliente já tem em mãos, o preço e todos os outros dados relevantes para construir um pedido de compra. Este pedido pode ser feito utilizando uma interface computacional. O cliente entra em um site, realiza seu pedido e confirma a compra. Isso é um serviço web, isto é, um serviço que está publicado na web que pode estar disponível para qualquer pessoa, ou a um determinado grupo (W3C, 2017).

2.9 Implementação

Durante o desenvolvimento de um software, a utilização correta de padrões, tecnologias e técnicas garante a qualidade do produto final (MARTIN, 2000), ambos serão descritos nesta seção.

2.9.1 MVC (Model, View, Controller)

MVC é um padrão de arquitetura que visa separar uma aplicação, ou até mesmo um pedaço da interface de uma aplicação em três partes: Modelo (*model*), Visão (*view*) e Controlador (*Controller*) - (LIU, 2010). Tem como objetivo separar dados lógicos de negócio (Modelo) da interface do usuário (Visão) e do fluxo da aplicação (Controlador), permitindo que uma mesma lógica de negócios possa ser acessada e visualizada através de várias interfaces.

- Modelo: é a parte do aplicativo que lida com a lógica para os dados do aplicativo, inserindo, buscando, excluindo e alterando dados;

- Visão: É a interface que interage com o usuário, responsável por tudo o que o que ele visualiza, também é a camada visão que recebe os dados, informações e comandos inseridos pelo usuário e as envia para a controlador;
- Controlador: É a interface do sistema que possui as regras de negócio, ela é responsável por controlar todo o fluxo de informação que passa pelo sistema, realizando a comunicação entre a visão e o controlador.

A FIGURA 6 demonstra como funciona o padrão MVC.

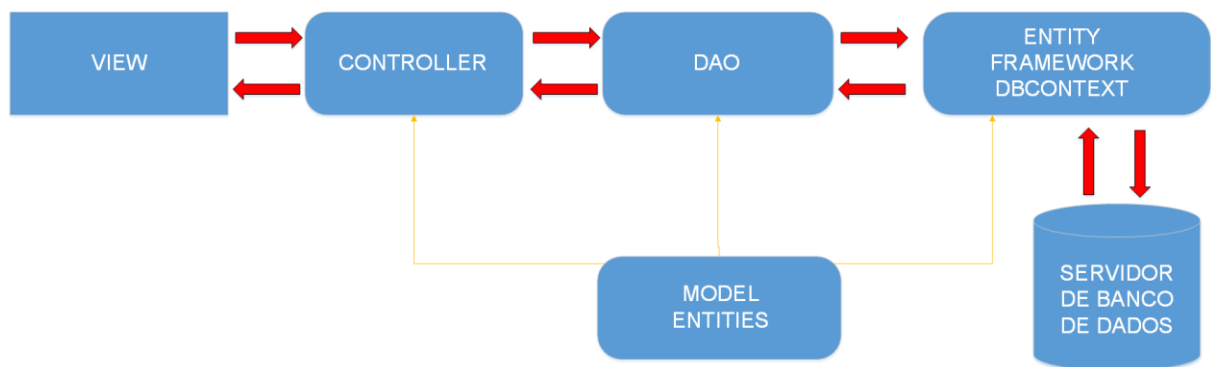


FIGURA 6 – PADRÃO MVC

FONTE OS AUTORES

2.9.2 Trello

O Trello é uma ferramenta de colaboração que organiza seus projetos em quadros, exatamente como a metodologia Kanban (TRELLO, 2015). O sistema Trello informa o que está sendo trabalhado, quem está trabalhando em quê, e onde algo está em um processo. A organização das *sprints* no Trello poder ser vista na FIGURA 7.

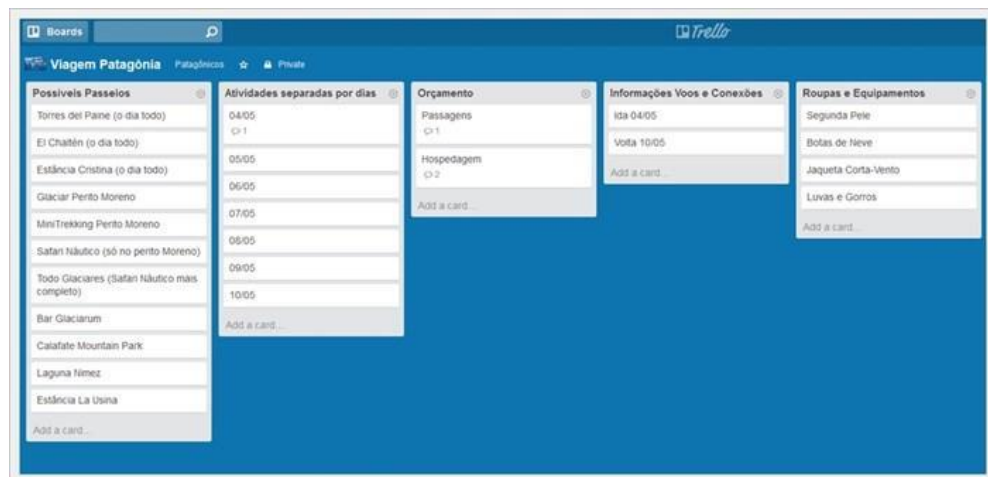


FIGURA 7 – EXEMPLO TRELLO
FONTE: Trello, 2017

2.9.3 Versionamento

O Versionamento ou Sistema de controle de versões, é um *software* capaz de registrar mudanças no contexto do arquivo ou conjunto de arquivos, permitindo que o usuário possa retroceder à uma versão específica (CHACON Straub, 2014).

2.9.3.1 Sistemas de controle de versões locais

O sistema de controle de versão local permite que o usuário consiga gerenciar seu sistema de controle de versões localmente, ou seja em sua própria máquina, podendo voltar a versões específicas sem fazer alterações no servidor. Como mostra a FIGURA 8.

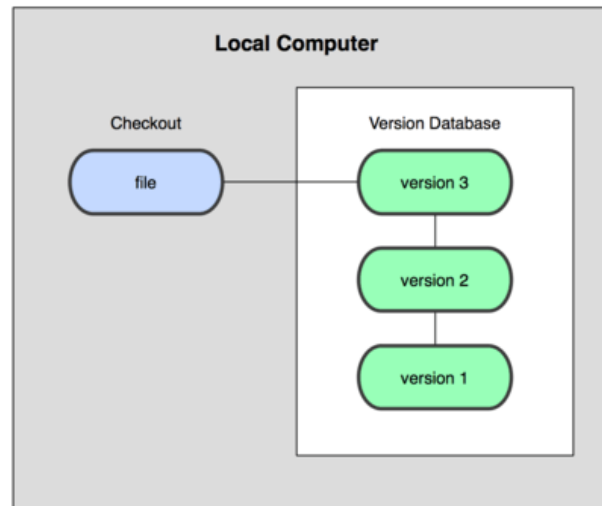


FIGURA 8 – SISTEMA DE CONTROLE DE VERSÃO LOCAL
FONTE: GIT, 2015

2.9.3.2 Sistemas de controle de versões centralizadas

Diferente de um sistema de controle de versão local, o sistema de controle de versão centralizado constitui-se de um sistema de controle de versão, capaz de versionar códigos e scripts de bancos de dados em um servidor, podendo ser centralizado todos os arquivos para que todos os outros usuários que estejam desenvolvendo o produto possam visualizar essas alterações. Como mostra a FIGURA 9.

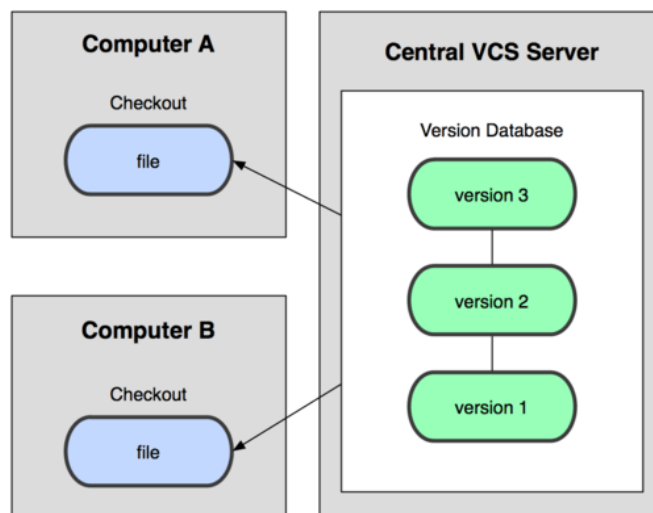


FIGURA 9 – SISTEMA DE CONTROLE DE VERSÃO CENTRALIZADA
FONTE: GIT, 2015

2.9.3.3 Sistemas de controle de versões distribuídas

Em um sistema de controle de versão distribuído, é possível recuperar todo o repositório caso ocorra alguma falha no servidor, pois o repositório onde ficam os arquivos pode ser encontrado em outros servidores. Esse sistema de controle de versão é o mais utilizado atualmente, tais como: Git, Mercurial, SVN, Tortoise e TFS. Como mostra a FIGURA 10.

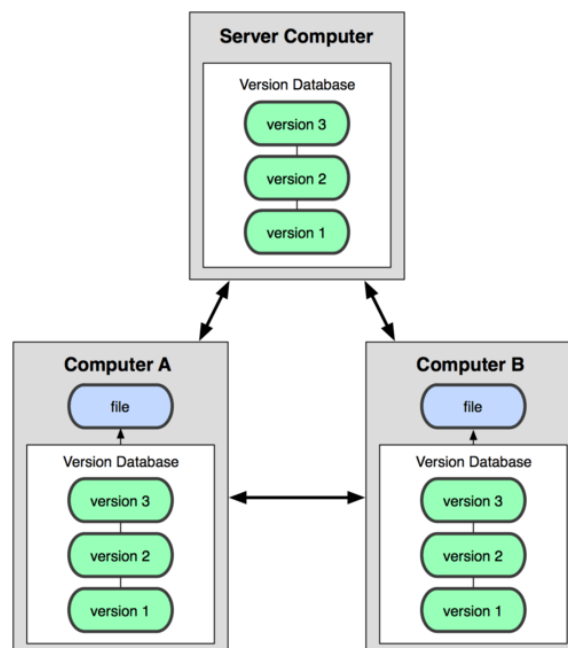


FIGURA 10 – SISTEMA DE CONTROLE DE VERSÃO DISTRIBUÍDO
FONTE: GIT, 2015

2.10 Entity Framework

O *Entity Framework* é um conjunto de tecnologias no ADO.NET que dão suporte ao desenvolvimento de aplicativos de software orientados a dados. Os arquitetos e desenvolvedores de aplicativos orientados a dados lutam com a necessidade de realizar dois objetivos muito diferentes. Precisam modelar as entidades, as relações e a lógica dos problemas de negócios que estão solucionando e também precisam trabalhar com os mecanismos de dados usados para armazenar e recuperar os dados (Microsoft, 2017).

3 METODOLOGIA

Para o desenvolvimento do sistema será utilizado a metodologia ágil *Scrum* com entregas a cada 3 semanas, no começo do desenvolvimento havia sido pensado em 5 *sprints*, porém ao rever a quantidade de atividades das *sprints* foi decidido adicionar mais uma e redistribuir melhor as tarefas, isso tudo logo no início do desenvolvimento. As reuniões diárias foram adaptadas ao Skype e outros programas que facilitem a comunicação quando a reunião não puder ser presencial, sempre que uma atividade não for completada na Sprint planejada ela será movida em caráter de emergência para a próxima Sprint.

3.1 Tecnologias

O aplicativo Android foi desenvolvido utilizando a linguagem de programação Java, com a IDE Android Studio. Para a interação com o banco de dados foi desenvolvido um projeto WEB API, implementando um Web Service, utilizando a arquitetura REST na linguagem C#. Todas as buscas e inserções no banco de dados que são realizadas pelo aplicativo são feitas a partir de chamadas deste serviço. Para administrar as chamadas, foi utilizada a biblioteca Volley.

Utilizando a IDE Visual Studio Community 2017, foi criado um projeto contendo biblioteca de classes que compõem o DAO (Data Access Object), com as entidades e classes de persistência. Também foi desenvolvido um projeto WEB utilizando o padrão de arquitetura MVC (Model, View e Controller), tendo em vista a experiência dos integrantes no desenvolvimento através dessa arquitetura. Para implementação do sistema web foi utilizado a linguagem de programação ASP.NET e C#. Também foram aplicadas algumas bibliotecas JavaScript, como biblioteca de galeria de imagens, biblioteca de calendário e a biblioteca de representação de chave de torneio. Todas elas foram modificadas e personalizadas para atender as necessidades dos sistemas.

Também está sendo utilizado nesse sistema o SAP Crystal Reports, ferramenta integrada a IDE Visual Studio que gera relatórios de maneira simplificada.

O SGBD (Sistema Gerenciador de Banco de Dados) escolhido foi o SQL Server Management Studio 2016. Como repositório do código-fonte no projeto WEB, DAO e WEB API, utilizou-se o TFS (Team Foundation Server) – repositório na nuvem que permite a criação de equipes e versionamento de código. Para o desenvolvimento do aplicativo Android foi utilizado o GIT LAB..

3.2 Modelo de Processo de Engenharia de Software

Levando em consideração a amplitude e complexidade do projeto e suas funcionalidades, o *backlog* do sistema foi criado baseado no documento WBS e separado por módulos. Sendo assim, foi necessário organizá-los em módulos para criação dos *backlogs* de cada *sprint*. Os seguintes procedimentos foram adotados para a divisão e execução das tarefas:

- Levantamento de requisitos e fundamentação teórica em paralelo;
- Desenvolvimento do Diagrama WBS para planejamento das *sprints* do projeto;
- Desenvolvimento dos Diagramas UML após o levantamento de requisitos e discussão do projeto entre os integrantes da equipe e o orientador;
- Desenvolvimento de protótipos funcionais para validação das funcionalidades, a fim de garantir que a tecnologia atenda aos requisitos do sistema;
- Modelagem inicial do banco de dados, sujeito a alterações na estrutura de acordo com as necessidades ao longo do desenvolvimento;
- Desenvolvimento das aplicações (website e aplicativo Android) em paralelo;
- Realização dos testes finais e finalização da documentação.

3.3 Plano de Atividade

As atividades realizadas durante o desenvolvimento do projeto foram planejadas e fundamentadas nos seguintes documentos:

- Cronograma;
- Tabela de Precedência;
- Diagrama de Gantt;
- WBS.

3.3.1 Cronograma

Com o objetivo de demonstrar o tempo de execuções das atividades, foram montados dois tipos de cronogramas: o cronograma planejado, contendo atividades e datas planejadas antes de começar o desenvolvimento do produto e o cronograma realizado, que mostra o que realmente foi feito em cada data, permitindo uma comparação. A utilização de cronograma tem como objetivo o planejamento de todas as atividades necessárias para a confecção do software e da sua documentação, assim como descrever a ordem cronológica em que serão realizadas. Para uma melhor organização das tarefas, foi definido como necessário a utilização das precedências das atividades antes de colocar em prática a distribuição em Kanbans entre a equipe. Essa decisão foi tomada para a realização desse projeto, assim não alterando as metodologias escolhidas mas sim buscando o aperfeiçoamento das mesmas.

3.3.1.1 Cronograma Planejado

Nessa seção estão demonstrados os cronogramas planejados antes do início do desenvolvimento do software. O QUADRO 1 representa a visão geral do cronograma, e os QUADROS 2 á 9 apresentam a visão mais detalhada.

ID	Nome da Tarefa	Trabalho (horas)	Início	Término
1	Sistema TPM	1640	08/09/2017	12/04/2017
2	Gerenciamento do Projeto	130	08/09/2017	30/8/2017
7	Documentação	214	30/8/2017	16/9/2017
18	Sprint 1	168	08/07/2017	27/08/2017
25	Sprint 2	168	28/8/2017	17/9/2017
32	Sprint 3	272	18/09/2017	10/08/2017
43	Sprint 4	284	10/09/2017	29/10/2017
54	Sprint 5	268	30/10/2017	19/11/2017
62	Sprint 6	136	20/11/2017	12/04/2017

QUADRO 1 - VISÃO GERAL DO CRONOGRAMA DO PROJETO

FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos
3	Tabela de precedência	18	08/09/2017	08/11/2017	Alexandre, Felipe
4	Plano de risco	16	08/12/2017	19/8/2017	Alexandre, Felipe
5	Diagrama de Gantt	24	20/8/2017	22/8/2017	Alexandre
6	Formatação da Documentação	8	20/8/2017	22/8/2017	Felipe
7	WBS	16	23/8/2017	24/8/2017	Alexandre, Felipe
8	Backlog Trello	24	25/8/2017	27/8/2017	Alexandre
9	Plano de comunicação	24	28/8/2017	30/8/2017	Alexandre, Felipe, Guilherme e Luiz

QUADRO 2 - CRONOGRAMA DE GERENCIAMENTO DO PROJETO

FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
10	Requisitos Funcionais	40	30/8/2017	09/02/2017	Felipe	
11	Alteração dos Requisitos Funcionais	22	09/03/2017	09/05/2017	Alexandre, Guilherme e Luiz	10
12	Requisitos Não Funcionais	10	09/03/2017	09/05/2017	Felipe	
13	Diagramas de caso de uso	10	09/03/2017	09/05/2017	Alexandre	11
14	Especificação de caso de uso	33	09/06/2017	09/11/2017	Felipe	13
15	Diagrama de Classe	33	09/12/2017	17/9/2017	Felipe	11
16	Diagrama de Entidade Relacionamento	33	09/06/2017	09/10/2017	Guilherme e Luiz	11
17	Diagrama de Sequência	33	09/11/2017	16/9/2017	Alexandre, Felipe, Guilherme e Luiz	14

QUADRO 3 - CRONOGRAMA DA DOCUMENTAÇÃO
FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
18	Cronograma	30	08/07/2017	08/10/2017	Alexandre e Felipe	
19	Prototipação das Telas	24	08/11/2017	13/8/2017	Luiz	18
20	Análise de Requisitos	45	13/8/2017	17/8/2017	Felipe e Guilherme	18
21	Infraestrutura (servidor)	40	18/8/2017	22/8/2017	Luiz e Guilherme	20
22	Infraestrutura IDE, BD	25	23/8/2017	26/8/2017	Luiz e Guilherme	21
23	Trello - Atualizar	4	27/08/2017	27/08/2017	Alexandre e Felipe	22

QUADRO 4 - CRONOGRAMA DA SPRINT 1
FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
25	Alteração dos Requisitos Funcionais	24	28/8/2017	30/8/2017	Felipe	20
24	Plano de Risco	16	28/8/2017	29/8/2017	Alexandre	
25	Protótipo no Servidor	24	28/8/2017	30/8/2017	Luiz e Guilherme	22
26	Tabela de precedência	24	30/8/2017	09/01/2017	Alexandre	
27	Plano de comunicação	8	31/8/2017	09/05/2017	Felipe	
28	Criação diagrama lógico do BD	20	31/8/2017	09/02/2017	Luiz e Guilherme	22
29	Protótipo funcional Web	24	09/03/2017	09/10/2017	Luiz	21
30	Protótipo funcional App	24	09/03/2017	09/10/2017	Guilherme	21
31	Trello - Atualizar	4	16/9/2017	17/9/2017	Alexandre e Felipe	30

QUADRO 5 - CRONOGRAMA DA SPRINT 2
 FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
32	Criação do Script do BD	24	18/09/2017	22/9/2017	Felipe e Guilherme	28
33	Prototipar todas as telas do sistema	40	18/9/2017	27/9/2017	Luiz	30
34	Diagrama de classe	24	23/9/2017	28/9/2017	Felipe e Guilherme	
35	Diagramas de Casos de uso	24	18/9/2017	25/9/2017	Alexandre	
36	Início da Especificações de caso de uso	52	18/9/2017	30/9/2017	Felipe	35
37	UC 01 - Efetuar Login WEB	16	28/9/2017	10/01/2017	Luiz	36
38	UC 02 - Realizar Cadastro WEB	24	10/02/2017	10/06/2017	Luiz	36
39	UC 01 - Efetuar Login APP	16	29/9/2017	10/01/2017	Guilherme	36
40	UC 02 - Realizar Cadastro APP	24	10/02/2017	10/06/2017	Guilherme	36
41	Correção de Bugs	24	10/01/2017	10/04/2017	Alexandre e Felipe	36
42	Trello - Atualizar	4	10/07/2017	10/08/2017	Alexandre	36

QUADRO 6 - CRONOGRAMA DA SPRINT 3
 FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
43	Finalizar especificação de caso de uso	24	10/09/2017	10/12/2017	Felipe	
44	Configurar Servidor	16	10/09/2017	10/11/2017	Luiz e Guilherme	
45	Consolidar diagramas	24	10/12/2017	15/10/2017	Alexandre e Felipe	
46	UC 03 - Solicitar Conta Premium WEB	24	10/09/2017	10/12/2017	Luiz	44
47	UC 04 - Gerenciar Local WEB	36	13/10/2017	19/10/2017	Luiz	44
48	UC 05 - Gerenciar Torneio WEB	36	20/10/2017	25/10/2017	Luiz	44
49	UC 03 - Arbitrar Partida APP	36	10/09/2017	10/12/2017	Guilherme	44
50	UC 04 - Monitorar Partida APP	36	13/10/2017	19/10/2017	Guilherme	44
51	Correção de Bugs	24	16/10/2017	20/10/2017	Alexandre e Felipe	46
52	Alteração de Layout	24	21/10/2017	10/04/2017	Alexandre	46
53	Trello - Atualizar	4	24/10/2017	29/10/2017	Alexandre	52

QUADRO 7 - CRONOGRAMA DA SPRINT 4
FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
54	UC 06 - Gerenciar Árbitro WEB	24	30/10/2017	11/03/2017	Luiz	
55	UC 07 - Gerar Chave WEB	80	11/04/2017	14/11/2017	Luiz	
56	UC 08 - Gerar Relatório WEB	36	15/11/2017	19/11/2017	Luiz	
57	UC 05 - Inscrever-se em Torneio APP	48	30/10/2017	11/05/2017	Guilherme	
58	UC 06 - Visualizar Conquistas APP	36	11/06/2017	11/11/2017	Guilherme	
59	Correção de Bugs	24	11/04/2017	11/10/2017	Felipe	54
60	Alteração de Layout	16	11/12/2017	16/11/2017	Alexandre e Felipe	54
61	Trello - Atualizar	4	18/11/2017	19/11/2017	Alexandre	60

QUADRO 8 - CRONOGRAMA DA SPRINT 5
FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
62	UC 09 - Gerenciar Usuário WEB	24	20/11/2017	25/11/2017	Luiz	
63	UC 10 - Gerenciar Requisição WEB	36	26/11/2017	30/11/2017	Luiz	
64	UC 07 - Visualizar Estatísticas APP	24	20/11/2017	25/11/2017	Guilherme	
65	Correção de Bugs	24	26/11/2017	28/11/2017	Alexandre e Felipe	62
66	Alteração de Layout	24	29/11/2017	12/02/2017	Alexandre	62
67	Trello - Atualizar	4	12/03/2017	12/04/2017	Alexandre	66

QUADRO 9 - CRONOGRAMA DA SPRINT 6

FONTE: OS AUTORES, 2017

3.3.1.2 Cronograma Realizado

Nessa seção estão demonstrados os cronogramas realizados para o desenvolvimento do software. O QUADRO 10 representa a visão geral do cronograma, e os QUADROS 11 a 18 apresentam a visão mais detalhada.

ID	Nome da Tarefa	Trabalho (horas)	Início	Término
1	Sistema TPM	1600	08/09/2017	12/04/2017
2	Gerenciamento do Projeto	124	08/09/2017	27/8/2017
7	Documentação	214	30/8/2017	16/9/2017
18	Sprint 1	184	08/07/2017	27/08/2017
25	Sprint 2	172	28/8/2017	17/9/2017
32	Sprint 3	322	18/09/2017	15/10/2017
43	Sprint 4	286	16/10/2017	11/07/2017
54	Sprint 5	230	30/10/2017	25/11/2017
62	Sprint 6	68	24/11/2017	27/11/2017

QUADRO 10 - VISÃO GERAL DO CRONOGRAMA DO PROJETO

FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos
3	Tabela de precedência	8	08/09/2017	08/10/2017	Alexandre, Felipe
4	Plano de risco	8	08/11/2017	08/12/2017	Alexandre, Felipe
5	Diagrama de Gantt	36	13/8/2017	15/8/2017	Alexandre
6	Formatação de doc.	24	13/8/2017	14/8/2017	Felipe
7	WBS	8	16/8/2017	17/8/2017	Alexandre, Felipe
8	Backlog Trello	24	18/8/2017	21/8/2017	Alexandre
9	Plano de comunicação	16	25/8/2017	27/8/2017	Alexandre, Felipe, Guilherme e Luiz

QUADRO 11 - CRONOGRAMA DE GERENCIAMENTO DO PROJETO

FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
10	Requisitos Funcionais	40	30/8/2017	09/05/2017	Felipe	
11	Alteração dos Req. Func.	22	09/03/2017	09/05/2017	Alexandre, Guilherme e Luiz	10
12	Requisitos Não Funcionais	10	09/06/2017	09/08/2017	Felipe	
13	Diagramas de caso de uso	10	09/06/2017	09/08/2017	Alexandre	11
14	Especificação de caso de uso	33	09/09/2017	09/11/2017	Felipe	13
15	Diagrama de Classe	33	09/12/2017	17/9/2017	Felipe	11
16	Diagrama de Entidade Relacionamento	33	18/9/2017	21/9/2017	Guilherme e Luiz	11
17	Diagrama de Sequência	33	14/9/2017	16/9/2017	Alexandre, Felipe, Guilherme e Luiz	14

QUADRO 12 - CRONOGRAMA DA DOCUMENTAÇÃO

FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
18	Cronograma	40	08/07/2017	08/10/2017	Alexandre e Felipe	
19	Prototipação das Telas	24	08/11/2017	13/8/2017	Luiz	18
20	Análise de Requisitos	44	13/8/2017	17/8/2017	Felipe e Guilherme	18
21	Infraestrutura (servidor)	44	18/8/2017	22/8/2017	Luiz e Guilherme	20
22	Infraestrutura IDE, BD	24	23/8/2017	26/8/2017	Luiz e Guilherme	21
23	Trello - Atualizar	8	27/08/2017	27/08/2017	Alexandre e Felipe	22

QUADRO 13 - CRONOGRAMA DA SPRINT 1

FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
25	Alteração dos Req. Funcionais	24	28/8/2017	09/01/2017	Felipe	20
24	Plano de Risco	16	28/8/2017	30/8/2017	Alexandre	
25	Protótipo no Servidor	24	28/8/2017	09/01/2017	Luiz e Guilherme	22
26	Tabela de precedência	24	09/01/2017	09/04/2017	Alexandre	
27	Plano de comunicação	8	09/02/2017	09/05/2017	Felipe	
28	Criação diagrama lógico do BD	20	09/02/2017	09/04/2017	Luiz e Guilherme	22
29	Protótipo funcional Web	24	09/05/2017	09/11/2017	Luiz	21
30	Protótipo funcional App	24	09/05/2017	09/10/2017	Guilherme	21
31	Trello - Atualizar	8	16/9/2017	17/9/2017	Alexandre e Felipe	30

QUADRO 14 - CRONOGRAMA DA SPRINT 2

FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
32	Criação do Script do BD	36	18/09/2017	23/9/2017	Felipe e Guilherme	28
33	Prototipar todas as telas	40	18/9/2017	29/9/2017	Luiz	30
34	Diagrama de classe	36	30/9/2017	10/04/2017	Felipe e Guilherme	
35	Diagramas de Casos de uso	16	18/9/2017	20/9/2017	Alexandre	
36	Início da Especificações de caso de uso	48	10/05/2017	10/11/2017	Felipe	35
37	UC 01 - Efetuar Login WEB	24	30/9/2017	10/02/2017	Luiz	36
38	UC 02 - Realizar Cadastro WEB	24	10/03/2017	10/06/2017	Luiz	36
39	UC 01 - Efetuar Login APP	24	10/05/2017	10/08/2017	Guilherme	36
40	UC 02 - Realizar Cadastro APP	32	10/10/2017	14/10/2017	Guilherme	36
41	Correção de Bugs	36	21/9/2017	30/9/2017	Alexandre e Felipe	36
42	Trello - Atualizar	6	15/10/2017	15/10/2017	Alexandre	36

QUADRO 15 - CRONOGRAMA DA SPRINT 3

FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
43	Finalizar especificação de UC	24	16/10/2017	19/10/2017	Felipe	
44	Configurar Servidor	16	16/10/2017	17/20/2017	Luiz e Guilherme	
45	Consolidar diagramas	24	20/10/2017	22/10/2017	Alexandre e Felipe	
46	UC 03 - Solicitar Conta Premium WEB	24	18/10/2017	20/10/2017	Luiz e Guilherme	44
47	UC 04 - Gerenciar Local WEB	36	21/10/2017	24/10/2017	Luiz e Guilherme	44
48	UC 05 - Gerenciar Torneio WEB	36	25/10/2017	29/10/2017	Luiz	44
49	UC 03 - Arbitrar Partida APP	36	25/10/2017	29/10/2017	Guilherme	44
50	UC 04 - Monitorar Partida APP	36	30/10/2017	11/05/2017	Guilherme	44
51	Correção de Bugs	24	23/10/2017	25/10/2017	Alexandre e Felipe	46
52	Alteração de Layout	24	26/10/2017	30/10/2017	Alexandre	46
53	Trello - Atualizar	6	11/06/2017	11/07/2017	Alexandre	52

QUADRO 16 - CRONOGRAMA DA SPRINT 4

FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
54	UC 06 - Gerenciar Árbitro WEB	24	30/10/2017	11/03/2017	Luiz	
55	UC 07 - Gerar Chave WEB	56	11/06/2017	11/10/2017	Luiz e Guilherme	
56	UC 08 - Gerar Relatório WEB	36	11/11/2017	15/11/2017	Luiz	
57	UC 05 - Inscrever-se em Torneio APP	48	16/11/2017	20/11/2017	Luiz e Guilherme	
58	UC 06 - Visualizar Conquistas APP	36	21/11/2017	23/11/2017	Luiz e Guilherme	
59	Correção de Bugs	24	11/12/2017	25/11/2017	Alexandre e Felipe	54
61	Trello - Atualizar	6	25/11/2017	25/11/2017	Alexandre	60

QUADRO 17 - CRONOGRAMA DA SPRINT 5

FONTE: OS AUTORES, 2017

ID	Nome da Tarefa	Trabalho (horas)	Início	Término	Nome dos Recursos	Predecessoras
62	UC 09 - Gerenciar Usuário WEB	24	24/11/2017	26/11/2017	Guilherme e Luiz	
63	UC 10 - Gerenciar Requisição WEB	8	27/11/2017	27/11/2017	Guilherme e Luiz	
64	UC 07 - Visualizar Estatísticas APP	8	27/11/2017	27/11/2017	Guilherme e Luiz	
65	Correção de Bugs	24	27/11/2017	27/11/2017	Alexandre, Guilherme, Felipe e Luiz	62
67	Trello - Atualizar	4	27/11/2017	27/11/2017	Alexandre e Felipe	66

QUADRO 18 - CRONOGRAMA DA SPRINT 6

FONTE: OS AUTORES, 2017

O planejamento de atividades bem efetuado resulta no bom andamento das atividades e faz com que nenhum membro da equipe fique sobrecarregado. Apesar da dificuldade em entregar 100% das tarefas nas datas estipuladas, tudo ocorreu da forma esperada, sem prejuízos para o desenvolvimento dos sistemas.

3.3.2 Tabela de Precedência

Visando o sucesso do projeto, foi definida uma ordem para a realização das principais atividades. A definição dessas atividades foi baseada nos levantamentos de requisitos junto ao cliente, adicionada da experiência técnica da equipe de desenvolvedores. Essa ordem está representada no QUADRO 19.

Nº Identificador	Atividade	Precedência
1	Acesso ao sistema	
2	Manter Usuários	1
3	Manter Local	
4	Manter Torneios	3
5	Manter Partida	3
6	Gerar Chaves	4
7	Gerar Estatísticas	5
8	Manter Árbitro	5
9	Gerar Relatórios	2, 3, 4

QUADRO 19 – TABELA DE PRECEDÊNCIA

FONTE: OS AUTORES, 2017

3.3.3 WBS

Na WBS será criada uma árvore de atividades, onde no topo temos o projeto como atividade principal, abaixo as ramificações em 6 Sprints e em cada Sprint ramificações de atividades a serem desenvolvidas no sistema sem entrar muito em detalhes.

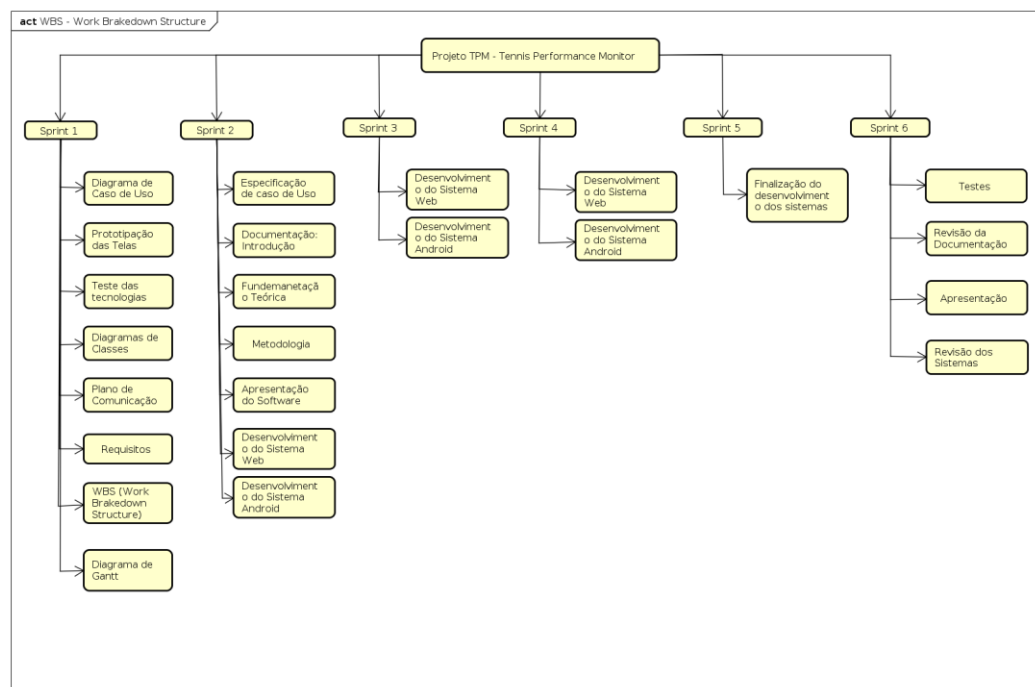


FIGURA 11 – WBS DO SISTEMA TPM
FONTE: OS AUTORES, 2017

3.3.4 Gráfico de Gantt

No diagrama de Gantt será inserido todas as atividades com o seu responsável e o tempo em que deverá terminar a mesma, caso ocorra uma atividade com possível descumprimento de prazo, o colaborador do projeto será realocado para que a mesma atividade possa ser completada em tempo hábil. O Diagrama de Gantt pode ser visto na FIGURA 12.

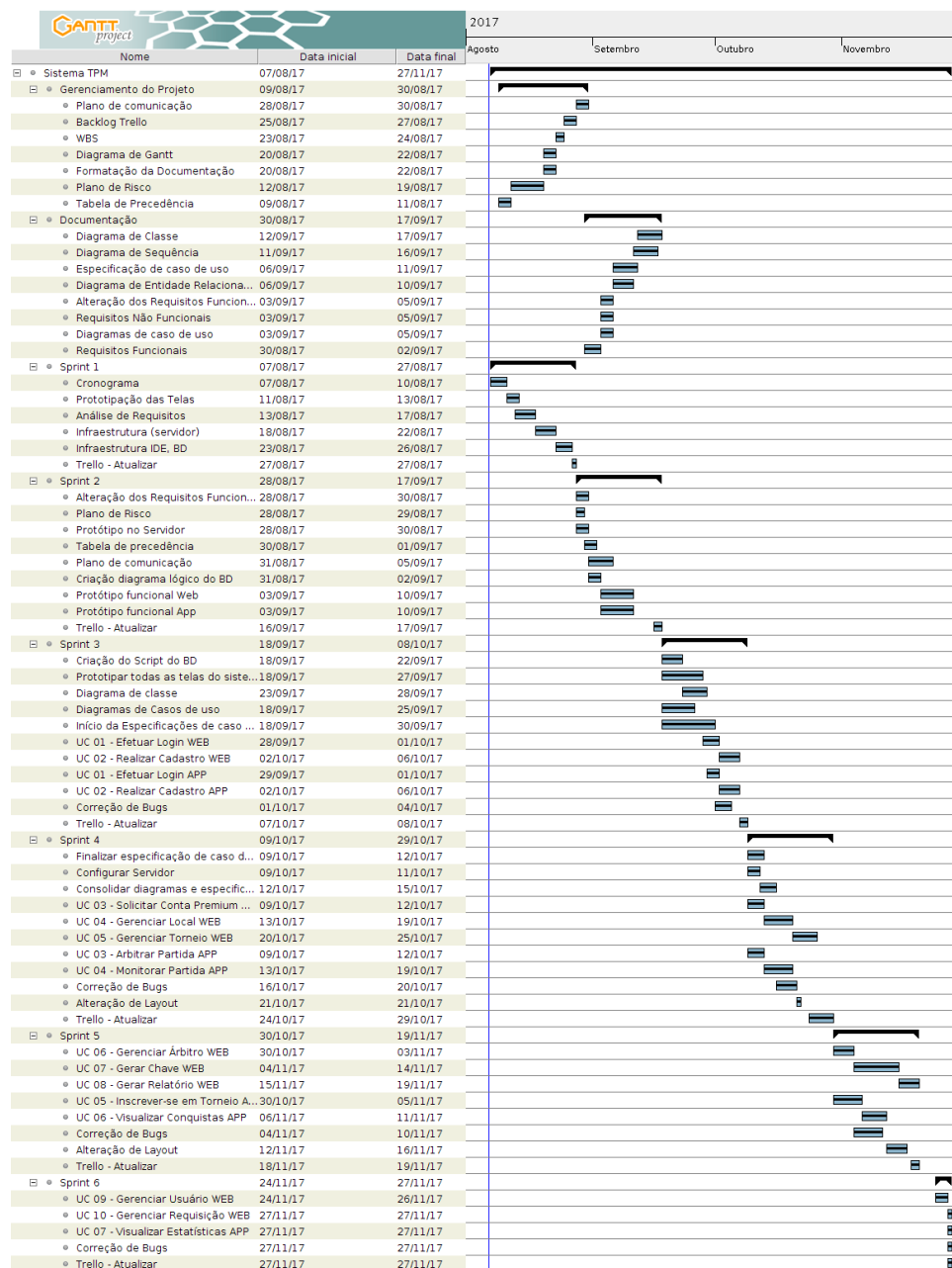


FIGURA 12 – GRÁFICO DE GANTT
FONTE: OS AUTORES, 2017

3.3.5 Requisitos Funcionais e Requisitos Não Funcionais

Nesta seção é demonstrado o que os sistemas devem oferecer ao usuário no que se refere as suas funcionalidades principais e secundárias.

Os requisitos, classificados como funcionais e não funcionais, foram levantados através do método CBL (*Challenge Based Learning*). O processo se inicia em: como “Desenvolver um sistema web e um aplicativo de geração de estatísticas relacionado aos jogos de tênis”. Definiu-se os requisitos para desenvolvimento, detalhados nas seções a seguir:

Os requisitos funcionais estão divididos quanto ao seu caráter operacional:

- RFU – Requisito Funcional de Usuário;
- RFP – Requisito Funcional de Usuário Premium;
- RFA – Requisito Funcional de Administrador.

REQUISITOS FUNCIONAIS DE USUÁRIO – SISTEMA WEB

Número	Requisito	Descrição
RFU1	Cadastro	O Sistema Web deve permitir o cadastro de todos os usuários que informarem dados corretos ou com a sincronização com a conta do Facebook. E de acordo com as regras de negócios.
RFU2	Solicitar Conta Premium	O Sistema Web deve permitir que o Usuário solicite uma conta premium na tela inicial.

QUADRO 20 – REQUISITOS FUNCIONAIS DE USUÁRIO – SISTEMA WEB

FONTE: OS AUTORES, 2017

REQUISITOS FUNCIONAIS DE USUÁRIO PREMIUM – SISTEMA WEB

Número	Requisito	Descrição
RFP1	Login	O Sistema Web deve permitir o acesso ao sistema mediante a inserção de dados válidos ou com a sincronização com a conta do Facebook.
RFP2	Geração de Chave	O Usuário Premium poderá solicitar, no Sistema Web, uma chave de torneio quando obtiver o mínimo de quatro inscritos.
RFP3	Geração de Relatório	O Usuário Premium poderá solicitar, no Sistema Web, relatórios com estatísticas de jogos, torneios e usuários.
RFP4	Gerenciar Torneio	O Usuário Premium poderá, no Sistema Web, criar, editar e cancelar torneio.
RFP5	Gerenciar Instalações	O Usuário Premium poderá, no Sistema Web, criar, editar e cancelar instalações.

QUADRO 21 – REQUISITOS FUNCIONAIS DE USUÁRIO PREMIUM – WEB

FONTE: OS AUTORES, 2017

REQUISITOS FUNCIONAIS DE ADMINISTRADOR – SISTEMA WEB

RFA1	Login	O Sistema Web deve permitir o acesso ao sistema mediante a inserção de dados válidos e não terá sincronização com o Facebook.
RFA2	Gerenciar Instalações	O Administrador poderá, no Sistema Web, criar, editar e cancelar instalações.
RFA3	Gerenciar Torneio	O Administrador poderá, no Sistema Web, criar, editar e cancelar torneio.
RFA4	Geração de Relatório	O Administrador poderá solicitar, no Sistema Web, relatórios com estatísticas de jogos, torneios e usuários.
RFA5	Gerenciar Requisição	O Administrador poderá, no Sistema Web, gerenciar solicitações de Usuários para obterem acesso de Usuário Premium.
RFA6	Gerenciar Usuário	O Administrador poderá, no Sistema Web, visualizar, editar informações e desativar Usuários e Usuários Premium.

QUADRO 22 – REQUISITOS FUNCIONAIS DE ADMINISTRADOR – WEB

FONTE: OS AUTORES, 2017

REQUISITOS FUNCIONAIS DE USUÁRIO – APP

Número	Requisito	Descrição
RFU1	Cadastro	O app deve permitir o cadastro de todos os usuários que informarem dados corretos ou com a sincronização com a conta do Facebook. E de acordo com as regras de negócios.
RFU2	Login	O app deve permitir o acesso ao sistema mediante a inserção de dados válidos ou com a sincronização com a conta do Facebook.
RFU3	Arbitrar Partida	O app deve permitir que o Usuário tenha acesso a arbitrar partidas de campeonatos, se for selecionado pelo Usuário Premium como árbitro.
RFU4	Monitorar Partida	O app deve permitir que o Usuário monitore partidas amistosas. Calculando cada pontuação e registrando as jogadas no banco de dados.
RFU5	Inscrever-se em torneio	O app deve permitir que o Usuário consiga inscrever-se em torneio que esteja disponível.
RFU6	Visualizar Conquistas	O app deve permitir que o Usuário consiga visualizar todas as conquistas realizadas.
RFU7	Visualizar Estatísticas	O app deve permitir que o Usuário consiga visualizar todas as estatísticas realizadas durante os jogos assinalados pelo usuário ou árbitro.

QUADRO 23 – REQUISITOS FUNCIONAIS DE USUÁRIO – APP

FONTE: OS AUTORES, 2017

REQUISITOS NÃO FUNCIONAIS

RNF1	Software SGBD	O SGBD será o SQL Server hospedado no servidor da Azure, pois possui a confiabilidade necessária, além de obter as ferramentas necessárias para as atividades do projeto.
RNF2	Linguagem de Programação e arquitetura - WEB	A linguagem utilizada para o desenvolvimento de todas as ferramentas será C#, utilizando também a arquitetura MVC (Model, View e Controller).
RNF3	Linguagem de Programação e arquitetura - Android	A linguagem utilizada para o desenvolvimento de todas as ferramentas será Java, utilizando também a arquitetura MVC (Model, View e Controller).
RNF4	Portabilidade	O Software será disponibilizado em uma versão <i>web</i> e outra para dispositivos móveis Android.
RNF5	Tipo de Interface desejada	A versão <i>web</i> poderá ser acessada em qualquer navegador.
RNF6	Mobilidade	O sistema deverá possuir uma versão para dispositivos móveis.
RNF7	Segurança	Os Sistemas realizarão a criptografia dos dados dos usuários.
RNF8	Desempenho	O Desempenho e a qualidade dos sistemas deverão ser garantidos através da realização de testes unitários.
RNF9	Integração dos Sistemas	Os sistemas deverão ser integrados através de um <i>webservice</i> que irá gerenciar todas as requisições ao banco de dados e disponibilizar as ferramentas.

RNF10	Servidor Disponível 99,99% do tempo	O servidor deve ficar disponível durante 99,99% do tempo durante a execução da aplicação.
RNF11	Validação dos campos	Os campos dos formulários devem ser validados tanto no lado do cliente quanto no lado do servidor.

QUADRO 24 – REQUISITOS NÃO FUNCIONAIS

FONTE: OS AUTORES, 2017

3.4 Plano de Riscos

No plano de risco situado no QUADRO 25, foram descritas as condições de risco que poderiam ter afetado o desenvolvimento do projeto, suas consequências e o plano de ação que deveria ser empregado caso houvesse necessidade. Também foi descrito quais eram os responsáveis pelo monitoramento de cada risco, sua probabilidade de ocorrência e classificação.

ID	Condição	Data Limite	Consequência	Ação	Monitoramento	Probabilidade	Impacto	Classificação
1	Perda de integrante	Sem limite	Acumulo de tarefas para os demais integrantes	Remanejamento das atividades aos integrantes da equipe	Toda a Equipe	Baixíssima	Altíssimo	4
2	Falta de conhecimento dos processos utilizados	27/08/2017	Atraso no desenvolvimento	Estudar os processos a serem utilizados	Toda a Equipe	Moderado	Alto	6
3	Falta de conhecimento da Tecnologia utilizada	17/09/2017	Atraso no desenvolvimento	Estudar a tecnologia a tecnologia utilizada	Toda a Equipe	Moderado	Alto	6
4	Problema de comunicação	17/09/2017	Atraso na conclusão das atividades e possível falha no desenvolvimento das mesmas	Verificar os pontos e serem corrigidos para não ocorrerem novamente	Toda a Equipe	Baixo	Baixo	4
5	Alteração dos requisitos	29/10/2017	Atraso no desenvolvimento devido ao retrabalho	Alteração no escopo e modificação no desenvolvimento do projeto	Toda a Equipe	Moderado	Moderado	5

QUADRO 25 – PLANO DE RISCOS

FONTE: OS AUTORES, 2017

3.5 Responsabilidades

As responsabilidades de cada integrante da equipe pode ser visualizada no QUADRO 26. Todas as tarefas foram determinadas antes do início de cada *Sprint*.

ID	Nome da Tarefa	Responsáveis pelas atividades
1	Tabela de precedência	Alexandre, Felipe
2	Plano de risco	Alexandre, Felipe
3	Diagrama de Gantt	Alexandre
4	Formatação da Documentação	Felipe
5	WBS	Alexandre, Felipe
6	Backlog Trello	Alexandre
7	Plano de comunicação	Alexandre, Felipe, Guilherme e Luiz
8	Requisitos Funcionais	Felipe
9	Alteração dos Requisitos Funcionais	Alexandre, Guilherme e Luiz
10	Requisitos Não Funcionais	Felipe
11	Diagramas de caso de uso	Alexandre
12	Especificação de caso de uso	Felipe
13	Diagrama de Classe	Felipe
14	Diagrama de Entidade Relacionamento	Guilherme e Luiz
15	Diagrama de Sequência	Alexandre, Felipe, Guilherme e Luiz
16	Cronograma	Alexandre e Felipe
17	Prototipação das Telas	Luiz
18	Análise de Requisitos	Felipe e Guilherme
19	Infraestrutura (servidor)	Luiz e Guilherme
20	Infraestrutura IDE, BD	Luiz e Guilherme
21	Trello - Atualizar	Alexandre e Felipe
22	Alteração dos Requisitos Funcionais	Felipe
23	Plano de Risco	Alexandre
24	Protótipo no Servidor	Luiz e Guilherme
25	Tabela de precedência	Alexandre
26	Plano de comunicação	Felipe
27	Criação diagrama lógico do BD	Luiz e Guilherme
28	Protótipo funcional Web	Luiz
29	Protótipo funcional App	Guilherme
30	Trello - Atualizar	Alexandre e Felipe
31	Criação do Script do BD	Felipe e Guilherme
32	Prototipar todas as telas do sistema	Luiz
33	Diagrama de classe	Felipe e Guilherme
34	Diagramas de Casos de uso	Alexandre
35	Início da Especificações de caso de uso	Felipe
36	UC 01 - Efetuar Login WEB	Luiz
37	UC 02 - Realizar Cadastro WEB	Luiz
38	UC 01 - Efetuar Login APP	Guilherme
39	UC 02 - Realizar Cadastro APP	Guilherme
40	Correção de Bugs	Alexandre e Felipe
41	Trello - Atualizar	Alexandre
42	Finalizar especificação de caso de uso	Felipe
43	Configurar Servidor	Luiz e Guilherme
44	Consolidar diagramas e especificação na documentação	Alexandre e Felipe
45	UC 03 - Solicitar Conta Premium WEB	Luiz e Guilherme

46	UC 04 - Gerenciar Local WEB	Luiz e Guilherme
47	UC 05 - Gerenciar Torneio WEB	Luiz
48	UC 03 - Arbitrar Partida APP	Guilherme
49	UC 04 - Monitorar Partida APP	Guilherme
50	Correção de Bugs	Alexandre e Felipe
51	Alteração de Layout	Alexandre
52	Trello - Atualizar	Alexandre
53	UC 06 - Gerenciar Árbitro WEB	Luiz
54	UC 07 - Gerar Chave WEB	Luiz e Guilherme
55	UC 08 - Gerar Relatório WEB	Luiz
56	UC 05 - Inscrever-se em Torneio APP	Luiz e Guilherme
57	UC 06 - Visualizar Conquistas APP	Luiz e Guilherme
58	Correção de Bugs	Alexandre e Felipe
59	Trello - Atualizar	Alexandre
60	UC 09 - Gerenciar Usuário WEB	Guilherme e Luiz
61	UC 10 - Gerenciar Requisição WEB	Guilherme e Luiz
62	UC 07 - Visualizar Estatísticas APP	Guilherme e Luiz
63	Correção de Bugs	Alexandre, Guilherme, Felipe e Luiz
64	Trello - Atualizar	Alexandre e Felipe

QUADRO 26 - TABELA DE RESPONSABILIDADES
 FONTE: OS AUTORES, 2017

3.6 Materiais

Foram utilizados recursos de *software* e de *hardware* para o desenvolvimento do projeto. Alguns desses *softwares* são gratuitos, outros possuem licença de uso paga, porém, com uso gratuito permitido para 30 dias. Nessa seção estão explicados cada um desses recursos.

3.6.1 Recursos de *Hardware*

Os *hardwares* utilizados para desenvolvimento dos sistemas foram 4 notebooks e 3 smartphones Android, também utilizamos um servidor gratuito para estudantes para armazenamento dos dados.

1. COMPUTADOR 01:

- a. Nome da Máquina: GUILHERME-PC;
- b. Proprietário: Guilherme Guimarães;
- c. Sistema Operacional: Windows 10 Pro;
- d. Memória RAM: 8 GB;
- e. Processador: Intel Core i7 2.4GHz;
- f. Espaço de armazenamento: 500GB;

g. Placa de vídeo: Nvidia GTX860;

2. COMPUTADOR 02:

- a. Nome da Máquina: BILL-PC;
- b. Proprietário: Luiz Bill;
- c. Sistema Operacional: Windows 10 Pro;
- d. Memória RAM: 8 GB;
- e. Processador: Intel Core i5 2.2GHz;
- f. Espaço de armazenamento: 1500GB;
- g. Placa de vídeo: Nvidia GT940m;

3. COMPUTADOR 03:

- a. Nome da Máquina: LIPE-PC;
- b. Proprietário: Felipe Almeida;
- c. Sistema Operacional: Windows 10 Pro;
- d. Memória RAM: 12 GB;
- e. Processador: Intel Core i7 2.4GHz;
- f. Espaço de armazenamento: 1000GB;

4. COMPUTADOR 04:

- a. Nome da Máquina: ALE-PC;
- b. Proprietário: Alexandre Benatto Cadan;
- c. Sistema Operacional: Linux Mint 18 Sonya e Windows 10 Pro;
- d. Memória RAM: 16 GB;
- e. Processador: Intel Core i5 2.5GHz;
- f. Espaço de armazenamento: 720GB;

3.6.2 Recursos de *Software*

Os *softwares* utilizados estão divididos em duas categorias, *software* de desenvolvimento e *software* de documentação.

3.6.2.1 *Softwares* de Desenvolvimento

A seguir estão listados os softwares utilizados para o armazenamento de dados, codificação e funcionamento referentes ao desenvolvimento do sistema.

1. *Android Studio*

O *Android Studio* é uma IDE para desenvolvimento de aplicativos Android baseado na IDE *Intelli IDEA*, e fornece ferramentas de desenvolvimento integradas *Android* para desenvolvimento e depuração.

2. *Azure*

O *Azure* é um conjunto de abrangentes de serviços de nuvem da Microsoft que desenvolvedores e profissionais de TI usam para criar, implantar e gerenciar aplicativos por toda a rede global de datacenters. As ferramentas integradas, o DevOps e o Marketplace dão suporte a você para criar de maneira eficiente desde aplicativos móveis simples, soluções usadas em escala da Internet até sistemas quando grande fluxo de dados e acessos.

3. *Firebase*

Firebase é uma plataforma em nuvem do Google com várias ferramentas para desenvolvimento de aplicações, tanto móveis quanto WEB, oferece uma excelente infraestrutura para criação de aplicações de qualidade. O *Firebase* foi utilizado para armazenamento de imagens e arquivos, pois o *Azure* necessitaria de uma conta paga para que pudessemos armazenar as imagens.

4. *SQL Server*

O *SQL Server Management Studio* é um SGBD (Software Gerenciador de Banco de Dados) de fácil e simples utilização, essa ferramenta possui alto

desempenho tanto em simples consultas SQL, quanto ao desenvolvimento de bases de dados complexas.

5. Visual Studio Community 2017

O Visual Studio é uma IDE para desenvolvimento de sistemas, tanto Web, móveis quanto Desktop e diversas outras plataformas. É uma ferramenta completa de fácil aprendizado e possui uma excelente performance, tornando os projetos mais performáticos e de qualidade.

6. C#

A linguagem de programação C# é altamente expressiva, simples e de fácil aprendizagem. A sintaxe do C# pode ser instantaneamente reconhecida por qualquer pessoa familiarizada com C, C++ ou Java. Os desenvolvedores que sabem qualquer uma dessas linguagens são geralmente capazes de começar a trabalhar de forma produtiva com C# dentro de um tempo muito curto. A sintaxe do C# simplifica muitas das complexidades do C++ e fornece recursos poderosos, como tipos de valor nulo, enumerações, delegações, expressões lambda e acesso direto à memória, que não são encontrados no Java. O C# suporta métodos e tipos genéricos, que fornecem uma melhor segurança de tipo e desempenho, e iteradores, que permitem implementadores de coleções de classes para definir comportamentos de iteração personalizados que são simples de usar pelo código cliente. Expressões LINQ (consulta integrada à linguagem) fazem a consulta fortemente tipada, facilitando a manipulação de coleções tanto simples quanto coleções complexas de dados.

7. ASP NET MVC

O ASP NET MVC é um padrão para arquitetura de projetos WEB, que complementa a linguagem ASP NET existente, mas permite um controle maior da complexidade ao dividir o aplicativo em modelo, exibição e controle. Para a renderização das páginas web utiliza o motor *Razor*, uma *Engine View* da Microsoft que nos permite a inserção da lógica da aplicação diretamente na cama de visualização do projeto WEB.

8. SAP Crystal Reports For Visual Studio

É uma ferramenta gratuita do SAP, empresa de desenvolvimento de softwares para gestão empresarial, que integra a IDE Visual Studio, onde temos a possibilidade de gerar relatórios completos com informações ricas de maneira rápida e prática.

9. JavaScript

O JavaScript é uma linguagem de programação interpretada. Foi originalmente implementada como parte dos navegadores web para que scripts pudessem ser executados do lado do cliente e interagissem com o usuário sem a necessidade de esse script passar pelo servidor, controlando o navegador, realizando comunicação assíncrona e alterando o conteúdo do documento exibido. É atualmente a principal linguagem para programação *client-side* em navegadores web. Começam também a ser bastante utilizada do lado do servidor através de ambientes como o node.js. Foi concebida para ser uma linguagem script com orientação a objetos baseada em protótipos, tipagem fraca e dinâmica, e funções de primeira classe. Possui suporte à programação funcional e apresenta recursos como fechamentos e funções de alta ordem comumente indisponíveis em linguagens populares como Java e C++. É a linguagem de programação mais utilizada do mundo. É baseada em ECMAScript padronizada pela Ecma *international* nas especificações ECMA-262 e ISO/IEC 16262.

10. Java

Java é uma linguagem de programação orientada a objeto que funciona através de uma máquina virtual, podendo ser executada em qualquer sistema operacional que tenha Java instalado, sendo utilizada e plataforma computacional lançada pela primeira vez pela Sun Microsystems em 1995. O Java é rápido, seguro e confiável. De laptops a datacenters, consoles de games a supercomputadores científicos, telefones celulares à Internet.

3.6.2.2 *Softwares de Documentação*

Nesta seção estão listados os softwares utilizados para a criação da documentação do sistema.

1. Astah Professional

O Astah foi utilizado para fazer a prototipação dos diagramas de caso de uso, diagramas UML, WBS.

2. Balsamiq

Foi utilizado o Balsamiq para prototipar as telas tanto do sistema web quanto do sistema android.

3. DBDDesigner

O DBDesigner foi utilizado para desenvolver o diagrama de Entidade Relacionamento (DER).

4. GanttProject

GanttProject é um software livre. O código que é escrito por autores e contribuidores do GanttProject é distribuído nos termos da GNU General Public License versão 3 (GPL 3). Também usamos várias bibliotecas com diferentes licenças que são conhecidas como compatíveis com a GPL 3. O software foi utilizado na criação do diagrama de Gantt.

5. Google Docs

No começo da documentação foi utilizado o Google Docs para que pudéssemos organizar os tópicos.

6. Microsoft Office

O pacote office foi utilizado para escrever a documentação e criar tabelas para o desenvolvimento da mesma.

7. Microsoft Visio

O Microsoft Visio é um *software* para desenvolvimento de fluxogramas e os mais diversos tipos de diagramas, durante o desenvolvimento desse projeto, foi utilizado para o desenvolvimento dos casos de uso.

8. WPS Office

Esse pacote office foi utilizado para escrever a documentação e criar tabelas para o desenvolvimento da mesma.

3.7 Desenvolvimento do Projeto

Foi elaborado um fluxograma referente ao desenvolvimento do projeto demonstrando passo a passo FIGURA 13.

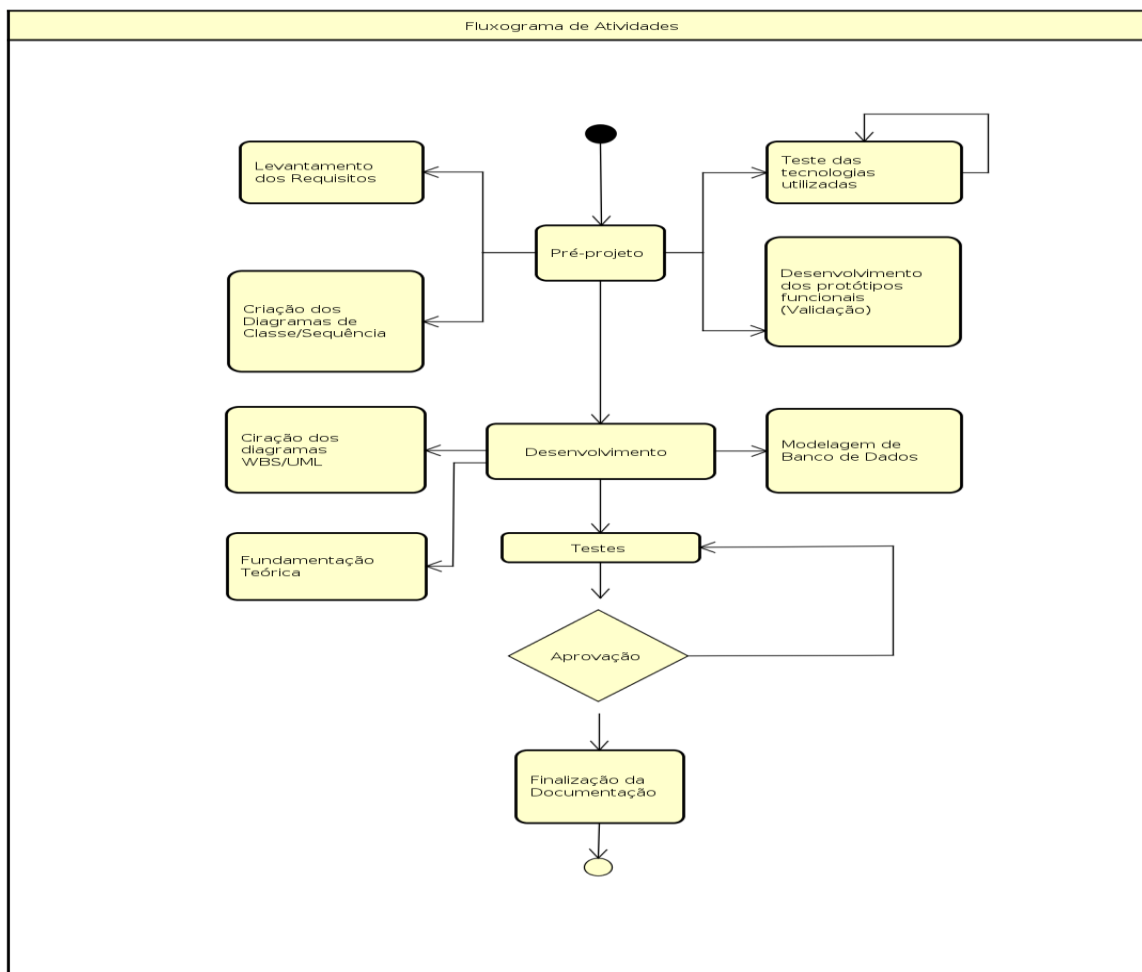


FIGURA 13 – FLUXOGRAMA DO PROJETO
FONTE: OS AUTORES, 2017

3.7.1 Definição do *backlog* do produto

O *backlog* do produto foi definido a partir do levantamento de requisitos, todos os requisitos foram divididos em partes menores. Estes requisitos menores são o que formam o *backlog* do produto desenvolvido, constituído de uma lista de requisitos. Nesta lista pode-se incrementar mais itens ao se perceber a necessidade de alguma funcionalidade adicional ou algum bug de maior importância durante as *sprints*. Os itens desta lista podem ser adicionados apenas pelo dono do produto.

3.7.2 Divisão do *backlog* do produto em *sprints*

O *backlog* do produto foi dividido em onze *sprints* logo no início do projeto, este fato gerou alguma dificuldade para definir com acurácia qual *sprint* deveria conter cada parte necessária para chegar no requisito. Sendo assim, algumas atividades foram reposicionadas no decorrer do processo de desenvolvimento do projeto. Caso a equipe não atingisse os objetivos de determinada *sprint*, as atividades incompletas retornavam para o *backlog* do produto, sendo, posteriormente, adicionados novamente a uma *Sprint* posterior. As *sprints* do projeto estão descritas a seguir. Tendo em vista a organização dos itens a serem desenvolvidos, a otimização do tempo de trabalho e a programação de maneira mais efetiva, as *sprints* apresentam os passos que foram realizados para chegar no requisito proposto, deixando exemplificados os procedimentos e não somente o objetivo final. Essa estratégia de organização de trabalho foi a escolhida por vários motivos, tendo como o mais relevante deles o fato de assim poder ter uma visão realista do trabalho que seria realizado durante cada *sprint*. As *sprints* do projeto estão descritas a seguir.

3.7.3 *Sprints* realizadas

Nessa seção teremos um resumo de todas as *sprints* realizadas no sistema e seus *backlogs* com figuras demonstrativas, também será possível visualizar a data de início e término juntamente com o tempo de desenvolvimento de cada *sprint*.

3.7.3.1 Sprint 1

Objetivos da *Sprint*:

- Cronograma;
- Prototipação das Telas;
- Análise de Requisitos;
- Infraestrutura (servidor);
- Infraestrutura IDE, BD;
- Trello – Atualizar.

Data de início: 07/08/2017

Data de entrega: 27/08/2017

Time box: 3 semanas

Demonstrado todas as atividades a serem desenvolvidas da Sprint 1 na FIGURA 14.



FIGURA 14 - BACKLOG DA SPRINT 1

FONTE: OS AUTORES, 2017.

3.7.3.2 Sprint 2

Objetivos da *Sprint*:

- Alteração dos Requisitos Funcionais;
- Plano de Risco;
- Protótipo no Servidor;
- Tabela de precedência;
- Plano de comunicação;
- Criação diagrama lógico do BD;
- Protótipo funcional Web;
- Protótipo funcional App;
- Trello – Atualizar.

Data de início: 28/08/2017

Data de entrega: 17/09/2017

Time box: 3 semanas

Demonstrado todas as atividades a serem desenvolvidas da Sprint 2 na FIGURA 15.



FIGURA 15 - BACKLOG DA SPRINT 2

FONTE: OS AUTORES, 2017.

3.7.3.3 Sprint 3

Objetivos da *sprint*:

- Criação do Script do BD
- Prototipar todas as telas do sistema
- Diagrama de classe
- Início da Especificações de caso de uso
- UC 01 - Efetuar Login WEB
- UC 02 - Realizar Cadastro WEB
- UC 01 - Efetuar Login APP
- UC 02 - Realizar Cadastro APP
- Correção de Bugs
- Trello - Atualizar

Data de início: 18/09/2017

Data de entrega: 08/10/2017

Time box: 3 semanas

Demonstrado todas as atividades a serem desenvolvidas da Sprint 3 na FIGURA 16.



FIGURA 16 - BACKLOG DA SPRINT 3

FONTE: OS AUTORES, 2017.

3.7.3.4 Sprint 4

Objetivos da Sprint:

- Finalizar especificação de caso de uso
- Configurar Servidor
- Consolidar diagramas e especificação na documentação
- UC 03 - Solicitar Conta Premium WEB
- UC 04 - Gerenciar Local WEB
- UC 05 - Gerenciar Torneio WEB
- UC 03 - Arbitrar Partida APP
- UC 04 - Monitorar Partida APP
- Correção de Bugs
- Alteração de Layout
- Trello - Atualizar

Data de início: 09/10/2017

Data de entrega: 29/10/2017

Time box: 3 semanas

Demonstrado todas as atividades a serem desenvolvidas da Sprint 4 na FIGURA 17.

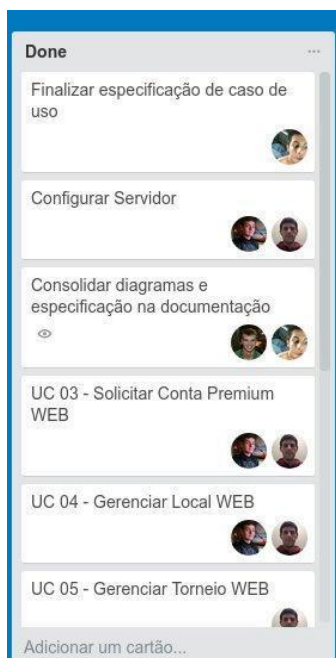


FIGURA 17 - BACKLOG DA SPRINT 4

FONTE: OS AUTORES, 2017.

3.7.3.4 Sprint 5

Objetivos da *Sprint*:

- UC 06 - Gerenciar Árbitro WEB
- UC 07 - Gerar Chave WEB
- UC 08 - Gerar Relatório WEB
- UC 05 - Inscrever-se em Torneio APP
- UC 06 - Visualizar Conquistas APP
- Correção de Bugs
- Trello - Atualizar

Data de início: 30/10/2017

Data de entrega: 19/11/2017

Time box: 3 semanas

Demonstrado todas as atividades a serem desenvolvidas da Sprint 5 na FIGURA 18.



FIGURA 18 - BACKLOG DA SPRINT 5

FONTE: OS AUTORES, 2017.

3.7.3.5 Sprint 6

Objetivos da *Sprint*:

- UC 09 - Gerenciar Usuário WEB
- UC 10 - Gerenciar Requisição WEB
- UC 07 - Visualizar Estatísticas APP
- Correção de Bugs
- Trello - Atualizar

Data de início: 20/11/2017

Data de entrega: 30/11/2017

Time box: 1 semana

Demonstrado todas as atividades a serem desenvolvidas da Sprint 6 na FIGURA 19.



FIGURA 19 - BACKLOG DA SPRINT 6

FONTE: OS AUTORES, 2017.

4 APRESENTAÇÃO DO SOFTWARE

Nesse capítulo será demonstrado as interfaces e suas funcionalidades de cada perfil de usuário com o sistema *web* e também sistema *mobile*.

4.1 Interfaces e Funcionalidades

O Aplicativo TPM – *Tennis Performance Monitor*, poderá proporcionar auxílio aos árbitros nas pontuações e estatísticas conforme o andamento do jogo e suas conquistas aos jogadores, estimulando que continuem tentando melhorar, no final do jogo o usuário poderá verificar suas estatísticas e pontuações sendo assim ver seus erros e acertos para poder evoluir a cada partida, também ajudará os usuários a encontrar quadras e torneios de forma rápida e prática. A utilização do sistema se dá por uma interface de *login* (FIGURA 21) em que ao usuário ser autenticado, caso ele seja um Usuário Comum, o sistema o redireciona para o painel com as funcionalidades de Usuário, caso seja um Usuário *Premium*, irá redirecioná-lo para o painel de Usuário *Premium*, caso contrário seja um Administrador, será redirecionado para o painel de Administrador.

Na FIGURA 20 está será demonstrado todo o fluxo de acesso ao sistema TPM.

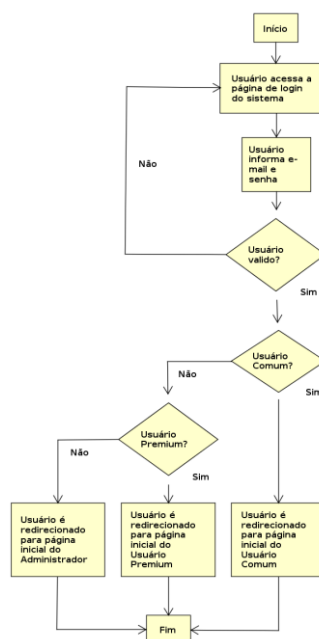


FIGURA 20 - FLUXO DE ACESSO AO SISTEMA TPM
FONTE: OS AUTORES, 2017.

4.1.1 Aplicativo

Nessa seção serão demonstradas as funcionalidades específicas do sistema mobile.

Na FIGURA 21 poderá ser visto a tela de *login* do sistema, aqui o usuário pode inserir seu usuário e senha ou realizar *login* com o *Facebook*.



FIGURA 21 – TELA DE LOGIN DO APLICATIVO TPM
FONTE: OS AUTORES, 2017.

No primeiro acesso ao sistema o usuário deverá realizar seu cadastro, para que isso ocorra ele deverá informar alguns dados assim como demonstrado na FIGURA 22.

FIGURA 22 – TELA DE CADASTRO DO APLICATIVO TPM
FONTE: OS AUTORES, 2017.

Conforme a FIGURA 23 será possível o usuário que possuir a permissão de árbitro selecionar um torneio e também obter informações do mesmo

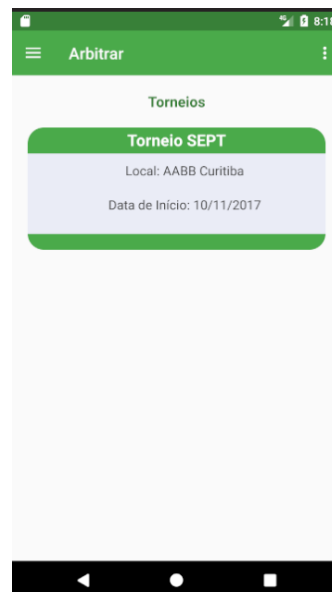


FIGURA 23 – TELA DE ARBITRAR TORNEIO DO APLICATIVO TPM
FONTE: OS AUTORES, 2017.

Na FIGURA 24 será possível o usuário que possuir a permissão de árbitro selecionar uma partida e arbitrá-la.

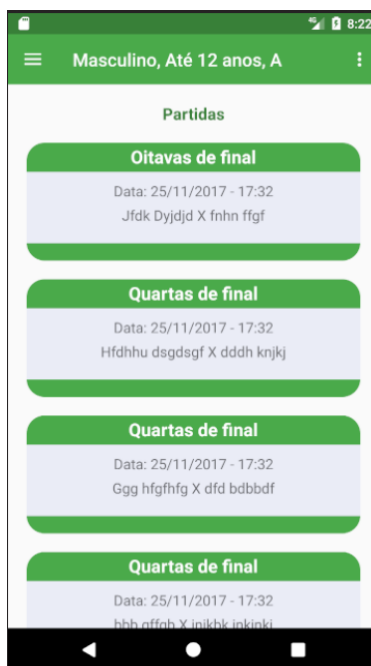


FIGURA 24 – TELA DE ARBITRAR PARTIDA DO APLICATIVO TPM
FONTE: OS AUTORES, 2017.

O árbitro da partida poderá inserir as pontuações na partida e seus erros, sendo assim ter acesso aos tipos de erros e acertos da mesma.

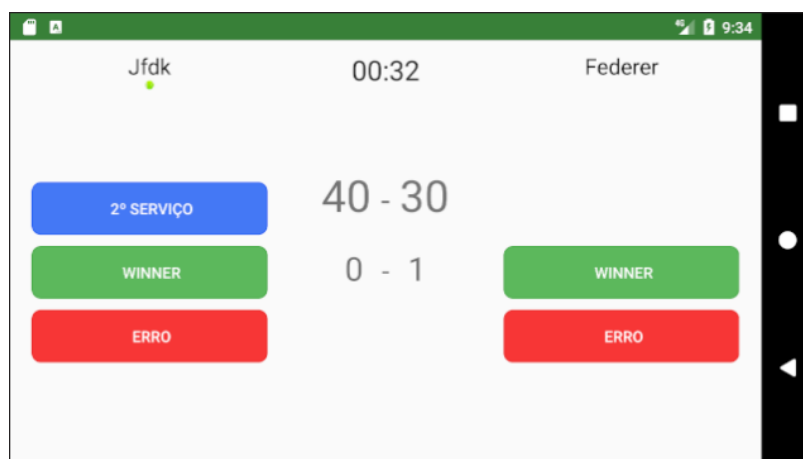


FIGURA 25 – TELA DE MONITORAR PARTIDA DO APLICATIVO TPM
FONTE: OS AUTORES, 2017.

O árbitro da partida poderá inserir as pontuações de acertos, sendo assim gerando estatísticas da partida que poderá ser vista pelos jogadores, conforme mostra a FIGURA 26.

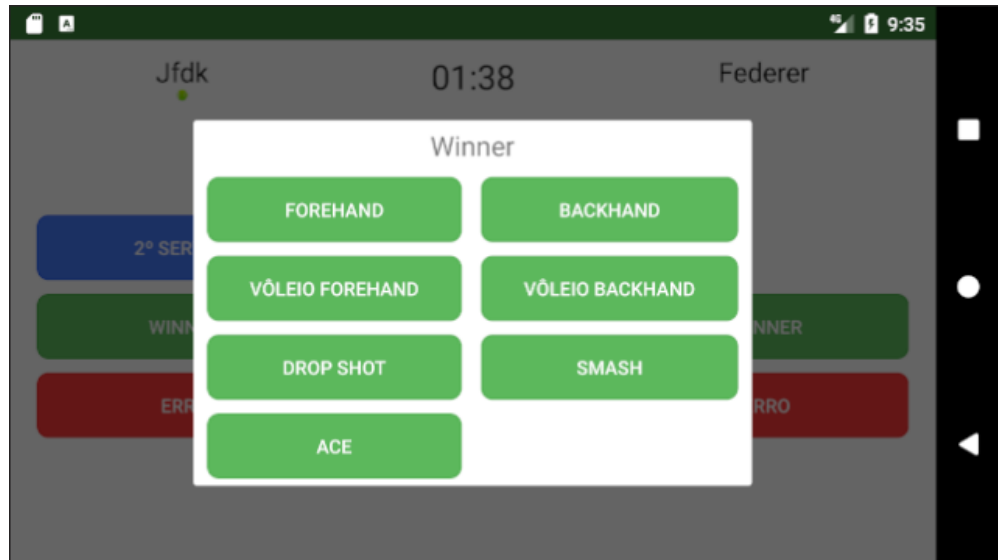


FIGURA 26 – TELA DE MONITORAR ACERTOS DO APLICATIVO TPM
FONTE: OS AUTORES, 2017.

O árbitro da partida poderá inserir as pontuações de erros, sendo assim gerando estatísticas da partida que poderá ser vista pelos jogadores, conforme mostra a FIGURA 27.

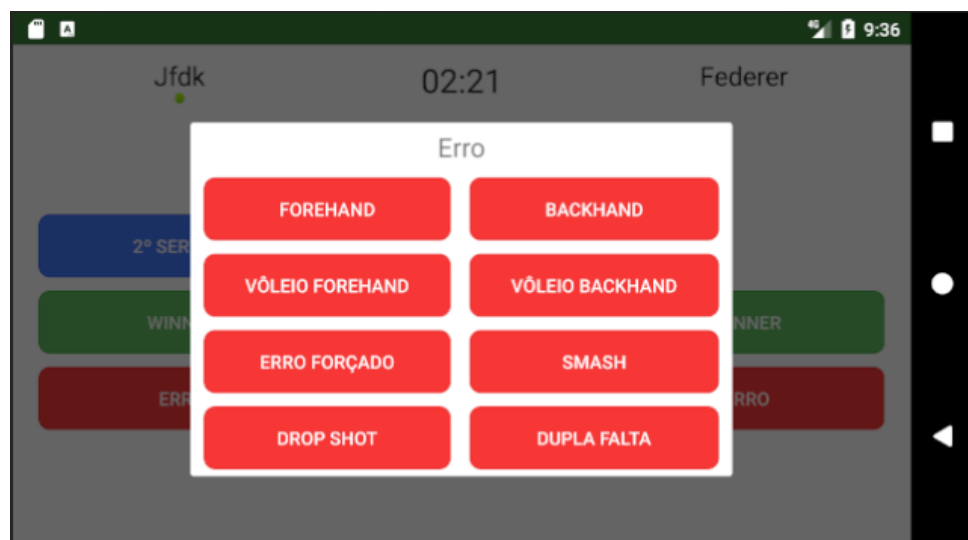


FIGURA 27 – TELA DE MONITORAR ERROS DO APLICATIVO TPM
FONTE: OS AUTORES, 2017.

O usuário poderá ter acesso aos torneios em que é possível se inscrever, conforme mostra a FIGURA 28

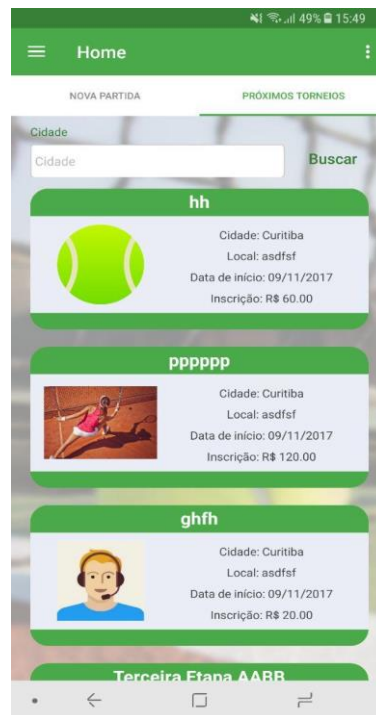


FIGURA 28 – TELA DE VISUALIZAR TORNEIOS DO APLICATIVO TPM
FONTE: OS AUTORES, 2017.

O usuário poderá visualizar todas suas conquistas e também as conquistas a serem obtidas conforme mostra a FIGURA 29.



FIGURA 29 – TELA DE VISUALIZAR CONQUISTAS DO APLICATIVO TPM
FONTE: OS AUTORES, 2017.

O usuário poderá visualizar as estatísticas de erros e acertos tanto de suas partidas e seus pontos em geral, conforme a FIGURA 30.

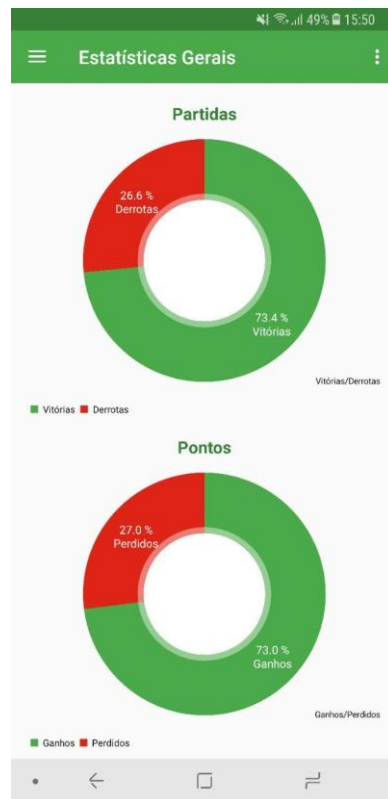


FIGURA 30 – TELA DE VISUALIZAR ESTATÍSTICAS DO APLICATIVO TPM
FONTE: OS AUTORES, 2017.

4.1.2 Portal Web

Nessa seção serão demonstradas as funcionalidades específicas do sistema web.

Na FIGURA 31 poderá ser visto a tela de *login* do sistema, aqui o usuário pode inserir seu usuário e senha ou realizar *login* com o *Facebook*.

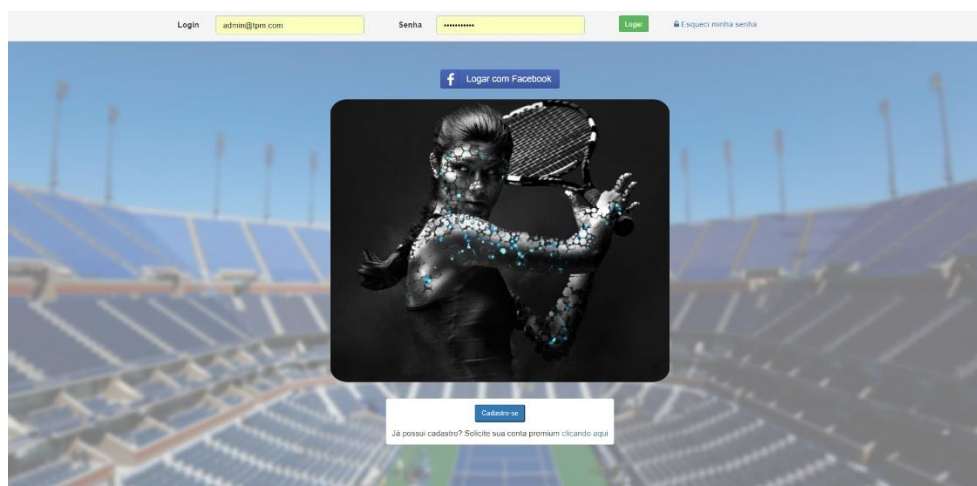


FIGURA 31 – TELA DE LOGIN DO SISTEMA WEB
FONTE: OS AUTORES, 2017.

Em seu primeiro acesso o usuário deverá realizar um breve cadastro, conforme mostra a FIGURA 32.

FIGURA 32 – TELA DE CADASTRO DE USUÁRIO DO SISTEMA WEB
FONTE: OS AUTORES, 2017.

O usuário poderá solicitar a qualquer momento após seu cadastro a solicitação de usuário *Premium*, conforme mostra a FIGURA 33.



Solicitação de Conta Premium ✕

Solicite acesso premium obter acesso completo ao nosso sistema!

Login

example@example.com

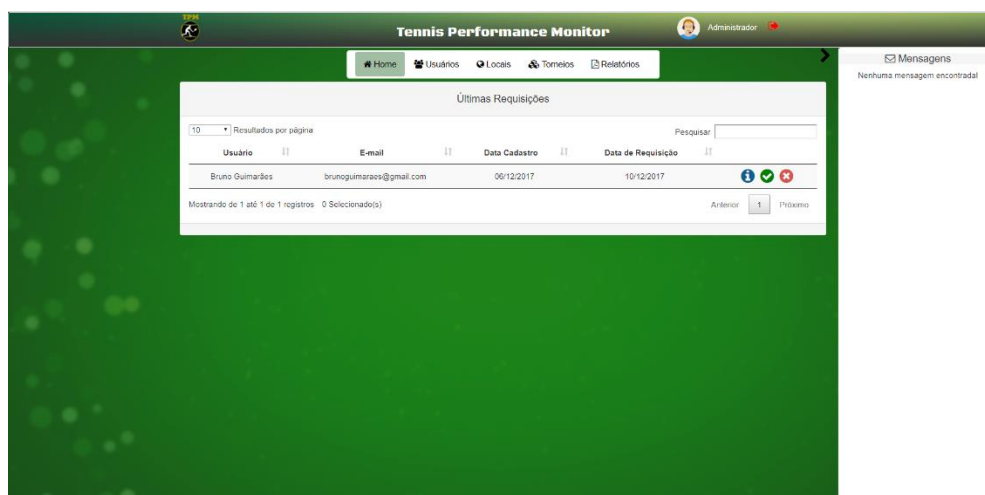
Senha

.....

Cancelar Facebook Solicitar

FIGURA 33 – TELA DE SOLICITAÇÃO DE USUÁRIO PREMIUM DO SISTEMA WEB
FONTE: OS AUTORES, 2017.

Na tela inicial do administrador ele poderá visualizar todas as requisições de acessos dos usuários comuns para *premiums*, sendo assim ele poderá autorizar ou não que os mesmos tenham acesso ao sistema, conforme mostra a FIGURA 34.



Tennis Performance Monitor Administrador

Home Usuários Locais Torneios Relatórios

Últimas Requisições

10 Resultados por página Pesquisar

Usuário	E-mail	Data Cadastro	Data de Requisição
Bruno Guimarães	brunoguimaraes@gmail.com	06/12/2017	10/12/2017

Mostrando de 1 até 1 de 1 registros 0 Solicionado(s)

Anterior 1 Próximo

Mensagens
Nenhuma mensagem encontrada

FIGURA 34 – TELA INICIAL DO ADMINISTRADOR DO SISTEMA WEB
FONTE: OS AUTORES, 2017.

Na tela de gerenciamento de usuário o administrador poderá remover um usuário do sistema ou editar seus dados, conforme mostra a FIGURA 35.

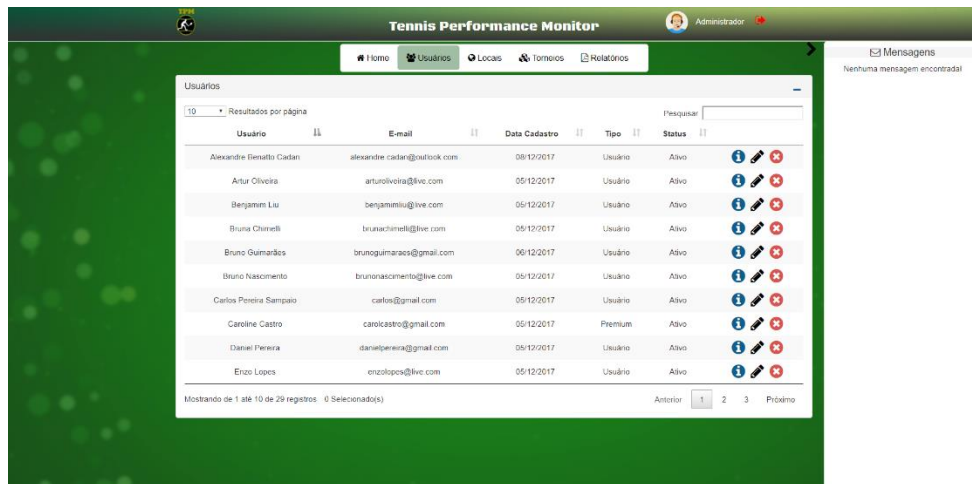


FIGURA 35 – TELA DE GERENCIAMENTO DE USUÁRIOS DO SISTEMA WEB
FONTE: OS AUTORES, 2017.

Na tela de relatórios será possível exibir diversos relatórios de usuários, instalações, torneios e inscrições *premium*, todas com determinados filtros, conforme mostra a FIGURA 36.

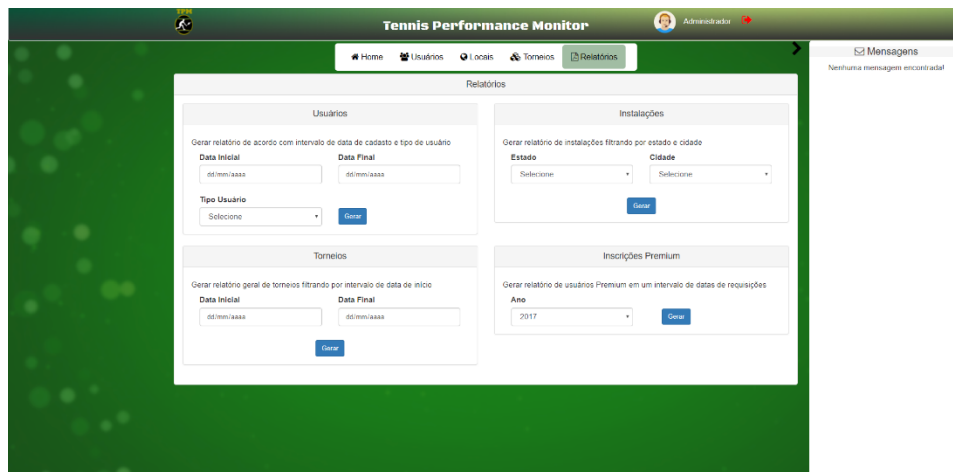


FIGURA 36 – TELA DE RELATÓRIOS DO ADMINISTRADOR DO SISTEMA WEB
FONTE: OS AUTORES, 2017.

Na tela de cadastro de locais, será possível cadastrar os locais onde serão disputados os torneios, conforme mostra a FIGURA 37.

Tennis Performance Monitor

Home Locais Torneios Relatórios Contato

Instalações

Editar Instalação

Nome: AABR - São Paulo | Email: aabre@aabr.com | Telefone: (11) 55185-514

Descrição: Associação Atlética Banco do Brasil - São Paulo

Proprietário: Guilherme Ungheddu | Imagem: 9ca7f85f-c957-436b-b239-ba6237031a78_12.jpg

CEP: 05325-000 | Logradouro: Estádio do Irapuema | Bairro: Vila Maracanã | N°: 1930 | Estado: SP | Cidade: São Paulo

Complemento: | Salvar

Próximas partidas:

10/12/2017 - 18:30
AABR SP - Terceira Etapa - Oitavas de Final
Marcelo, 15-34 anos A
Vitor Lopes X Adão Oliveira

FIGURA 37 – TELA DE CADASTRO DE LOCAIS DO SISTEMA WEB
FONTE: OS AUTORES, 2017.

Na tela de visualização de torneios será possível visualizar os torneios, conforme mostra a FIGURA 38.

Tennis Performance Monitor

Home Locais Torneios Relatórios Contato

Torneios

Novo Torneio

10 Resultados por página

Nome Data Início Data Limite Inscrição Total de Inscrições

Nome	Data Início	Data Limite Inscrição	Total de Inscrições
AABR SP - Terceira Etapa	12/10/2017	12/08/2017	15
AABR SP - Primeira Etapa	04/10/2017	03/10/2017	6
AABR SP - Quarta Etapa	15/12/2017	10/12/2017	1
AABR 2018	16/02/2018	28/12/2017	1

Mostrando de 1 até 4 de 4 registros. 0 Selecionado(s)

Anterior 1 Próximo

Próximas partidas:

10/12/2017 - 18:30
AABR SP - Terceira Etapa - Oitavas de Final
Marcelo, 15-34 anos A
Vitor Lopes X Adão Oliveira

FIGURA 38 – TELA DE VISUALIZAR TORNEIOS DO SISTEMA WEB
FONTE: OS AUTORES, 2017.

Na tela de edição de torneios será possível editar todos os dados do torneio e também os dados da premiação, inscritos e árbitros, conforme mostra a FIGURA 39.

FIGURA 39 – TELA DE EDITAR TORNEIOS DO SISTEMA WEB
FONTE: OS AUTORES, 2017.

Na tela de contato é possível enviar uma mensagem aos administradores do sistema, conforme mostra a FIGURA 40.

FIGURA 40 – TELA DE CONTATO DO SISTEMA WEB
FONTE: OS AUTORES, 2017.

Na tela de visualização de torneio é possível visualizar os torneios cadastrados pelos usuários *premium*, conforme mostra a figura 41.

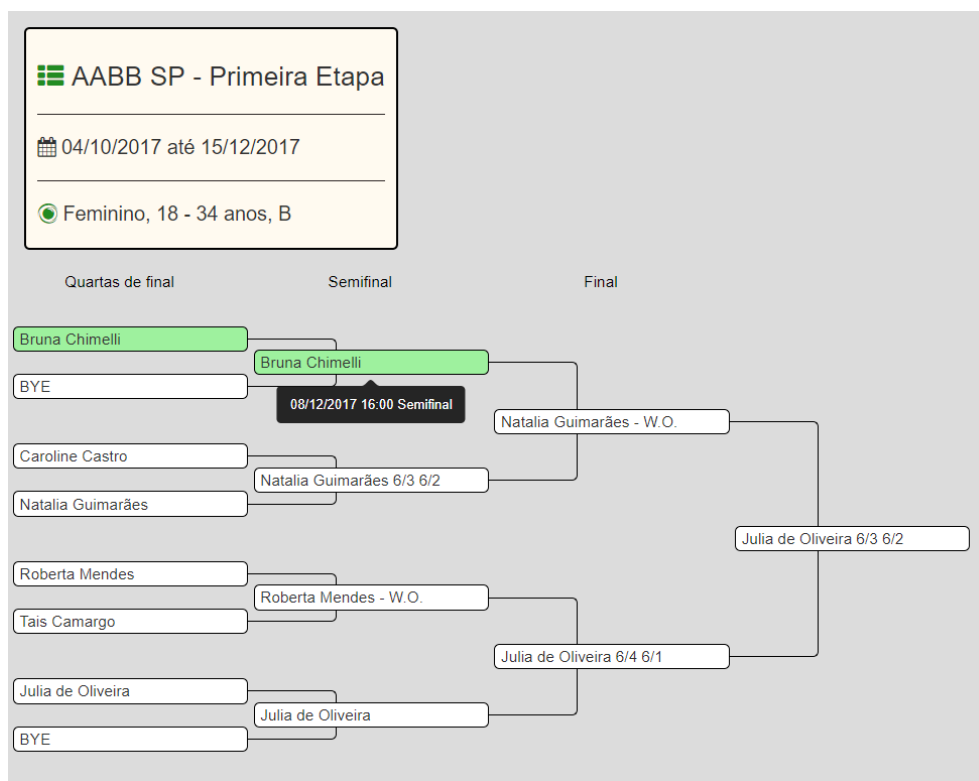


FIGURA 41 – TELA DE VISUALIZAÇÃO DO TORNEIO DO SISTEMA WEB
FONTE: OS AUTORES, 2017.

4.2 Perfis

Existem três tipos de perfis que podem acessar os sistemas TPM. Cada um possui diferentes acessos e funcionalidades.

4.2.1 Perfil Usuário Comum

O perfil de usuário comum é destinado aos jogadores e árbitros das partidas. Possuindo acesso apenas a funcionalidades do aplicativo, este tipo de usuário pode buscar e realizar inscrições em torneios, visualizar estatísticas gerais e separadas por jogo, visualizar todas as informações dos torneios em que é inscrito, visualizar suas conquistas e arbitrar partidas em que é arbitro.

4.2.2 Perfil de Usuário *Premium*

O perfil de usuário *premium* é destinado aos donos e gerenciadores de academias e clubes de tênis. Possuindo acesso apenas ao portal *web*, este tipo de usuário pode cadastrar e gerenciar suas instalações e torneios, além de gerar relatórios dos mesmos.

4.2.3 Perfil de Administrador

O perfil de administrador é destinado aos proprietários do sistema TPM. Com ele é possível gerenciar todos os usuários, removendo ou concedendo privilégios. Além disso pode gerenciar todas as instalações e torneios de todos os usuários e gerar relatórios mais detalhados dos mesmos.

5 CONSIDERAÇÕES FINAIS

Nas próximas seções serão apresentadas as dificuldades encontradas durante todo o projeto, o aprendizado obtido e os requisitos e melhorias determinados como implementações futuras no sistema desenvolvido.

5.1 Conclusão

Nesse projeto foram apresentadas todas as etapas de desenvolvimento do sistema TPM, durante este processo encontraram-se algumas dificuldades, tais como: curva de aprendizagem da biblioteca *Volley* para o sistema *Android*, biblioteca de galeria de imagens *JavaScript* para o sistema web, conflitos de horários do grupo gerando impossibilidade de reuniões diárias presenciais. O prazo para entrega foi de aproximadamente quatro meses, o que, para a quantidade de requisitos mensurados, pode ser considerado bastante escasso. Outra dificuldade foi a alocação de dois membros para desenvolvimento e outros dois para documentação, o que acabou impossibilitando que todos os membros dedicassem mais tempo em ambas as partes, isso foi resolvido nas últimas semanas com reuniões mais prolongadas. Percebeu-se também, que a cada modificação no escopo, o banco de dados sofria alterações, havendo atualizações constantes nos computadores dos membros da equipe.

Infelizmente, devido à escassez do tempo, não houve a possibilidade de realizar testes com o usuário final. Porém foi realizado a inserção de vários usuários, categorias, torneios e locais. Possibilitando realizar testes precisos, tanto no sistema aplicativo quanto no *Web*. A cada nova inserção, problemas de lentidão e até mesmo de erros encontrados no código eram corrigidos, chegando a um ponto onde a existência de erros era praticamente nula, salvo erros de rede e internet.

Os resultados destes testes foram bastante positivos, pois foi possível observar que tanto o processo de criação, atualização, inserção, quanto o de pontuação, foi bastante otimizado, gerando resultados proveitosos e ágeis.

Pode-se afirmar que o projeto foi bem-sucedido pois todos os objetivos específicos foram alcançados. Além disso, foi acrescentado durante a proposta mais itens, tais como: mensagens de usuário premium para administrador, inserção de

calendário no site para acompanhamento de datas, torneios e inscrições e um mapa no aplicativo que apresenta o local do torneio.

O processo de desenvolvimento dos sistemas forneceu à equipe várias novas experiências, como por exemplo: superar dificuldades no trabalho em equipe, e também a falta de uma melhor qualidade na comunicação, conduzindo atividades duplicadas. Os aprendizados mais importantes observados foram os seguintes: a utilização do software (TFS – *Team Foundation Server*) para controle de versão, pois praticamente não houve problemas na questão de versão dos códigos e em relação à forma como foi realizada a documentação, a utilização correta de *Frameworks* que facilitam o desenvolvimento de softwares, o enriquecimento de informações na utilização de cada linguagem de programação utilizada e ferramentas de desenvolvimento, entre vários outros. Para um próximo trabalho consideramos que toda a documentação poderia ser realizada de forma incremental, junto ao desenvolvimento, para evitar retrabalhos e deixando o processo mais fiel à metodologia ágil.

Os objetivos que serão alcançados ao concluir as implementações futuras serão a principal gratificação do grupo, que trará o sentimento de “dever cumprido” ao visualizar o aplicativo juntamente ao sistema web, beneficiando usuários, gerenciando locais e ajudando os atletas a buscarem um melhor desempenho dentro de quadra, com isso adquirindo uma melhor qualidade de vida.

5.2 Implementações Futuras

Ao longo do desenvolvimento do sistema pode-se observar a necessidade de algumas funcionalidades extras às inicialmente definidas. Algumas outras funcionalidades também acabaram por não ser implementadas, a maioria por falta de tempo hábil, tendo em vista que a duração do projeto foi de aproximadamente quatro meses. Todos estes requisitos novos e os pendentes ficam como sugestão para implementações futuras, os não existentes são:

5.2.1 Notificações

Tanto o aplicativo quanto no Sistema Web o usuário terá acesso a notificações de partidas próximas, quadras ou usuários que procuram por uma partida.

5.2.2 Adaptações para novas modalidades esportivas

O sistema poderá ser totalmente configurável, onde o usuário do mesmo poderá adicionar um esporte, regras, métodos de pontuação e penalidades.

5.2.3 Desenvolver em outras plataformas

Atualmente o sistema foi desenvolvido em modelos web (onde todos os usuários podem ter acesso através de um navegador) e sistema Android na plataforma mobile, nesse caso o desenvolvimento seria em Xamarin, possibilitando utilizá-lo nas 3 plataformas, Android, iOS e Windows Phone.

5.2.4 Criação de ranking de usuários

A última funcionalidade desejada é a criação de um ranking para cada categoria. Com isso, os usuários teriam uma motivação a mais para registrar suas partidas e se inscrever em torneios, já que isso faria ele ganhar pontos e subir de colocação no ranking.

5.3 Agradecimentos

Ao concluir o trabalho e o projeto, todos os integrantes da equipe perceberam e viram a grande importância dos esforços e trabalho em grupo que foram determinantes para o sucesso do desenvolvimento dos sistemas apresentados. Foi necessário o amplo envolvimento e dedicação sobre o projeto desenvolvido, inúmeras pesquisas, reuniões, comunicações, organização, coleta de informações referentes ao tênis e seu funcionamento, aplicação do conhecimento recebido na universidade e também no mercado de trabalho, divisão de responsabilidades, entre outros fatores positivos.

Gostaríamos de agradecer especialmente ao professor Dr. Razer Anthom Nizer Rojas Montaña por ter nos orientado durante todo o projeto, a todos os professores por terem repassado o conhecimentos necessário para a conclusão desse trabalho, aos colegas que passaram pela mesma experiência ao concluir o trabalho de conclusão de curso nos auxiliando, as empresas que trabalhamos ou já prestamos serviços por terem nos dado a oportunidade de aprimorar nossos conhecimentos, e a todos os nossos familiares e amigos pelo suporte prestado durante todo o desenvolvimento do projeto.

REFERÊNCIAS

- BECK, K.; BEEDLE, M.; BENNEKUM, A.; COCKBURN, A.; CUNNINGHAM, W.; FOWLER, M.; GRENNING, J.; HIGHSMITH, J.; HUNT, A.; JEFFRIES, R.; KERN, J.; MARICK, B.; MARTIN R. C.; MELLO, S.; SCHWABER, K.; SUTHERLAND, J.; THOMAS, D. **Manifesto para Desenvolvimento Ágil de Software, 2001**. Disponível em: <<http://agilemanifesto.org/iso/ptbr/>> . Acesso em: 05/11/2017.
- BERMIER, R. Total security in a Postgre SQL database. IBM, 2009. Disponível em: <<http://ibm.com/develiperworks/library/os-postgresecurity>>. Acesso em: 05/11/2015.
- BEZERRA, E. **Princípios De Análise E Projeto De Sistemas Com Uml**, 3ª Ed. Elsevier: Brasil, 2015.
- BOEHM, Barry W. Software risk management: principles and practices. **Software, IEEE**, v. 8, n. 1, p. 32-41, 1991.
- BOOTSTRAP. Disponível em :<<http://www.bootstrapzero.com/bootstraptemplate/grid>>. Acesso em: 28/10/2017.
- CHACON, S.; STRAUB, B. **Pro Git (2nd Edition)**, Apress, 2014
- CISNEIROS, D. **Modelo de Desenvolvimento Ágil. SCRUM**, 2009. Disponível em: <<http://www.devin.com.br/modelo-scrum/>>. Acesso em: 10/11/2017.
- COHN, M.; FORD, D. Introducing an agile process to an organization. **Computer**, n. 6, p. 74-78, 2003.
- DATATABLE. Disponível em: <<https://datatables.net/>>
- DEITEL, P.; DEITEL, H. **Java How to program**. Prentice Hall Press, 2011.
- EDITORA EUROPA, Almanaque Dos Esportes, 2009.
- FRAIN, B. **Responsive web design with HTML5 and CSS3**. Packt Publishing Ltd, 2012.
- FOWLER, M. **UML Essencial: um breve guia para linguagem padrão**. Bookman, 2014.
- GIT. Disponível em: <<https://git-scm.com/figures/18333fig0101-tn.png>>. Acessado em: 10/11/2017.
- GIT. Disponível em: <<https://git-scm.com/figures/18333fig0102-tn.png>>. Acessado em: 10/11/2017.
- GIT. Disponível em: <<https://git-scm.com/figures/18333fig0103-tn.png>>. Acessado em: 10/11/2017.
- GUEDES, G. TA. **UML 2: uma abordagem prática**. São Paulo: Novatec, v. 485, p. 3, 2009.

IEA. **Definition and Domains of ergonomics**. Disponível em: <<http://www.iea.cc/whats/index.html>>. Acesso em: 19/10/2017.

KNIBERG, H; SKARIN, M. **Kanban e Scrum** - obtendo o melhor de ambos. Estados Unidos: C4Media Inc, 2009.

LIBARDI, Paula LO; BARBOSA, Vladimir. **Métodos Ágeis**. 2010.

LIU, L. et al. Design and actualization of universal MVC pattern [J]. **Journal of University of Science and Technology of China**, v. 6, p. 017, 2010.

NIELSEN, J. **Usability Engineering**. New Jersey: Academic Press, 1993.

PEREIRA, Paulo; TORREÃO, Paula; MARÇAL, Ana Sofia. Entendendo Scrum para gerenciar projetos de forma ágil. **Mundo PM**, v. 1, p. 3-11, 2007.

PMI. **Um Guia do Conhecimento em Gerenciamento de Projetos**. Guia PMBOK®. Quarta Edição – EUA: Project Management Institute, 2008.

PRESSMAN, Roger S. **Engenharia de software**. McGraw Hill Brasil, 2011.

ORACLE. Disponível em:
<<http://www.oracle.com/technetwork/java/javaee/overview140548.html>>. Acessado em: 26/09/2017.

ROLL OF HONOUR. < <http://www.wimbledon.com/index.html>>. Acesso em 2017

SCHWABER, K.; SUTHERLAND, J. **Guia do Scrum: Um guia definitivo para o Scrum: As regras do jogo**. [SI], 2013.

SCHWARTZ, J. I. **Construction of software, Problems and Practicalities**. Addison-Wesley, Reading, MA, 1975.

SOMMERVILLE, Ian. **“Engenharia de Software”**, Addison Wesley, São Paulo, 2003.

STEFFEN, J. B. O que são essas tais de metodologias ágeis? IBM, 2012 .
Disponível em: <https://www.ibm.com/developerworks/community/blogs/rationalbrasil/entry/mas_o_que_s_c3_a3o_essas_tais_de_metodologias__c3_a1geis?lang=en>. Acessado em: 13/10/2017.

TECNOLOGIA. <<http://link.estadao.com.br/noticias/gadget,ate-o-fim-de-2017-brasil-tera-um-smartphone-por-habitante-diz-pesquisa-da-fgv,70001744407>>. Acesso em 2017.

TECNOLOGIA, Natação. <http://www.raiaoito.com.br/2016/01/evolucao-dos-trajes-de-natacao/>. Acesso em 2017.

THE TROPHIES. < <http://www.wimbledon.com/index.html>>. Acesso em 2017

TRELLO. Disponível em: <<http://help.trello.com/article/708-what-is-trello>>. Acessado em: 07/09/2017.

TREVISAN, Diogo Fernando; DA SILVA SACCHI, Rodrigo P.; SANABRIA, Lino. Estudo do Padrão Avançado de Criptografia AES–Advanced Encryption Standard. **Revista de Informática Teórica e Aplicada**, v. 20, n. 1, p. 13-24.

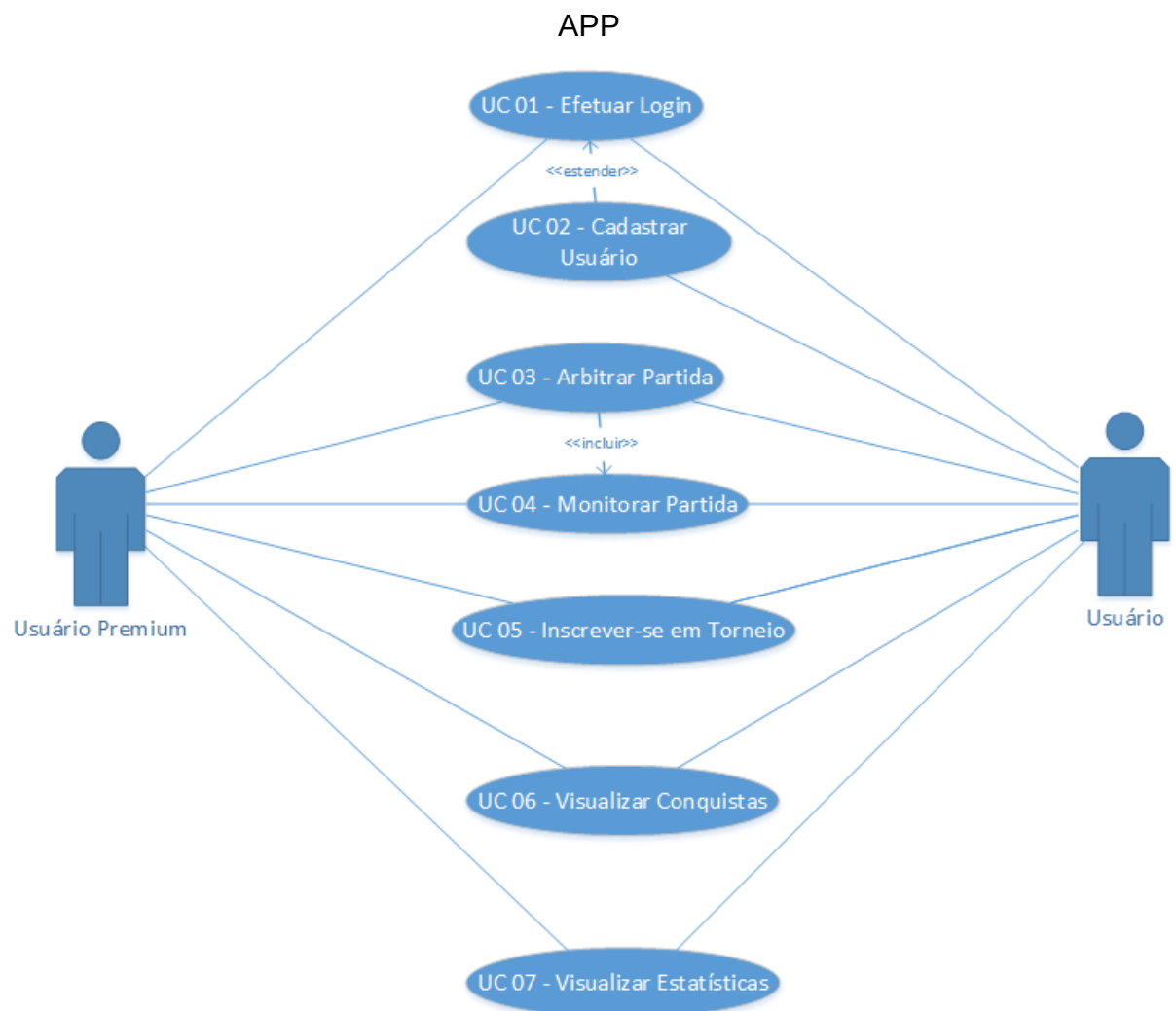
VIRTX. Soluções em TI . Virts, 2014. Disponível em:
<<http://virtx.com.br/wpcontent/uploads/2014/12/vps.png>>. Acessado em: 06/11/2017.

VOLLEY: Easy, Fast Networking for Android; Google I/O. Disponível em:
< <https://events.google.com/io/> >

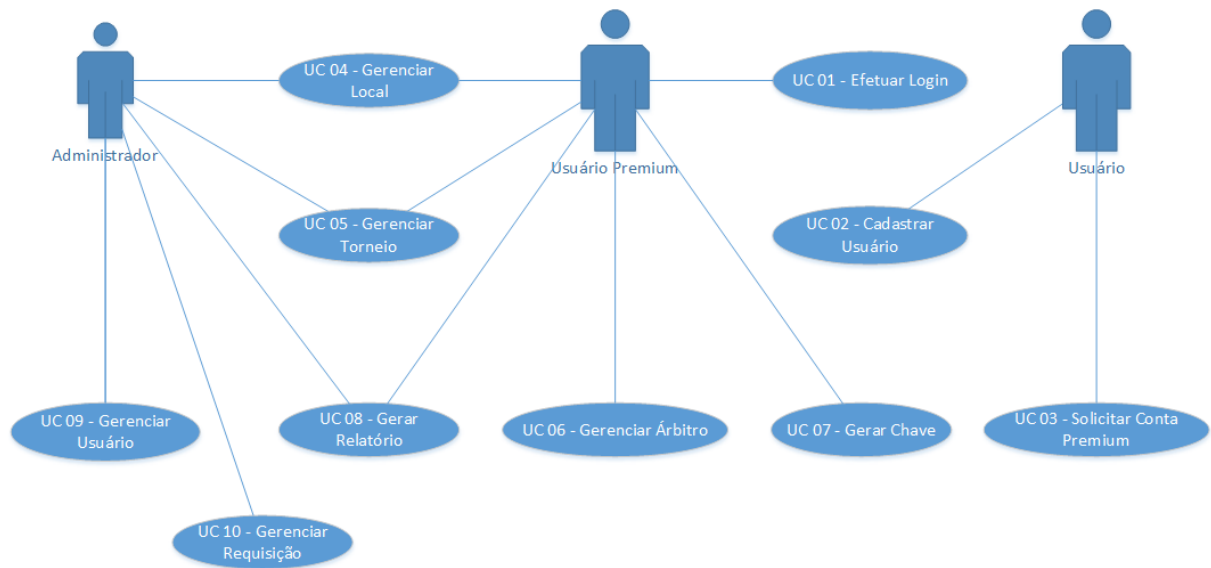
LISTA DE APÊNDICES

APÊNDICE A – DIAGRAMA DE CASOS DE USO	93
APÊNDICE B – ESPECIFICAÇÕES DE CASO DE USO.....	95
APÊNDICE C – DIAGRAMA DE CLASSE.....	144
APÊNDICE D – DIAGRAMA ENTIDADE RELACIONAMENTO.....	149
APÊNDICE E – PLANO DE COMUNICAÇÃO.....	150
APÊNDICE F – DIAGRAMA DE SEQUÊNCIA.....	151

APÊNDICE A – DIAGRAMA DE CASO DE USO



WEB



APÊNDICE B – ESPECIFICAÇÕES DE CASO DE USO

APP

UC01 – Efetuar Login

Descrição

Este caso de uso serve para que um Usuário ou Usuário *Premium* efetue *login* no sistema TPM *mobile*.

Data View

DV1 - Tela de login.



DV2 - Tela de Recuperação de Senha.**Pré-condições**

1. Ter *login* e senha registrados no sistema.

Pós-condições

Após o fim normal deste caso de uso sistema deve:

1. Ter autenticado o usuário permitindo o acesso ao sistema.

Ator Primário

Usuário.

Fluxo de Eventos Principal

1. O Sistema apresenta a Tela de Logar **(DV1)**;
2. O Usuário insere seu e-mail no campo E-mail;
3. O Usuário insere sua senha no campo senha;
4. O Usuário clica no botão “Entrar” **(A1) (A2) (A3)**;
5. O Sistema valida o usuário com o banco de dados **(E1) (E2)**;
6. O Sistema redireciona o usuário;
7. O caso de uso é finalizado.

Fluxos Alternativos

A1: Recuperação de senha.

1. O Usuário clica no botão “Esqueci minha senha”;
2. O Sistema apresenta a tela Recuperação de Senha **(DV2)**;
3. O Usuário informa seu e-mail de acesso ao sistema;
4. O Usuário clica no botão “Enviar”;
5. O Sistema verifica a existência do e-mail cadastrado no banco de dados **(E1)**;
6. O Sistema cria uma nova senha de acesso para o usuário e salva no banco de dados;
7. O Sistema envia para o e-mail do usuário sua nova senha;
8. O Sistema emite a mensagem “Nova senha foi enviada para seu e-mail”;
9. O Sistema redireciona para a tela de Logar **(DV1)**;
10. O caso de uso é reiniciado.

A2: Entrar com *Facebook*.

1. O Usuário clica no botão “Entrar com *Facebook*”;
2. O Sistema valida o usuário com o banco de dados **(E1) (E2)**;
3. O Sistema redireciona o usuário;
4. O caso de uso é finalizado.

A3: Cadastrar Usuário.

1. O Usuário clica no botão “Cadastre-se!”;
2. O Sistema carrega o UC002 – Cadastrar;
3. O caso de uso é finalizado.

Fluxos de Exceção

E1. E-mail Inválido:

1. O sistema consiste o e-mail;
2. O sistema retorna a mensagem “E-mail inválido! Por favor, insira um e-mail válido”;
3. O caso de uso é reiniciado.

E2. Senha Inválida:

1. O sistema consiste a senha;
2. O sistema retorna a mensagem “Senha inválida! Por favor, insira uma senha válida”.
3. O caso de uso é reiniciado.

Especificação de Caso de Uso

UC02 – Realizar Cadastro

Descrição

Este caso de uso serve para cadastrar o usuário no sistema TPM mobile.

Data View

DV1 - Cadastro.

The image shows three sequential screens of a mobile application for user registration. Each screen is displayed on a smartphone mockup with a status bar at the top showing 'ABC' and '06:19 PM'.

- Screen 1 (Cadastro-se):** Contains fields for 'Nome' (Name), 'Sobrenome' (Surname), 'Sexo' (Gender) with radio buttons for 'Masculino' (selected) and 'Feminino', 'Telefone' (Phone) with a country code dropdown, 'Data de Nascimento' (Date of Birth) with a date picker, 'Mão' (Hand) with a dropdown set to 'Destro', and 'Classe' (Class) with a dropdown set to 'A'.
- Screen 2:** Contains fields for 'E-mail', 'Senha' (Password), 'Confirme senha' (Confirm password), 'Foto de perfil' (Profile picture) with a camera icon, 'CEP' (Postal Code) with a 'Buscar' (Search) button, 'Logradouro' (Address), and 'Número' (Number).
- Screen 3:** Contains fields for 'Complemento' (Complement), 'Bairro' (Neighborhood), 'UF' (State) with a dropdown set to 'Selecione', and 'Cidade' (City). At the bottom are two buttons: 'Cancelar' (Cancel) in red and 'Cadastrar' (Register) in green.

Pré-condições

Este caso de uso pode iniciar somente se:

1. Ter acesso ao aplicativo TPM.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Cadastrar os dados do Usuário no sistema APP TPM.

Ator Primário

Usuário.

Fluxo de Eventos Principal

1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário insere Nome e Sobrenome;
3. O Usuário seleciona a opção de sexo no radio button Sexo;
4. O Usuário insere telefone;
5. O Usuário seleciona o campo Data de Nascimento **(E1)**;
6. O Usuário seleciona a Data de Nascimento;
7. O Usuário seleciona a opção no combo box Mão;
8. O Sistema retorna uma lista com as opções;
9. O Usuário seleciona a opção;
10. O Usuário seleciona a opção no combo box Classe;
11. O Usuário insere um e-mail **(E2)**;
12. O Sistema valida o e-mail;
13. O Usuário insere uma senha **(E3)**;
14. O Sistema valida a senha;
15. O Usuário confirma a senha **(E4)**;
16. O Usuário seleciona o campo Foto de Perfil;
17. O Sistema abre um caminho para o álbum de fotos;
18. O Usuário seleciona uma imagem **(E5)**;
19. O Sistema carrega a imagem;
20. O Usuário insere um CEP e seleciona o botão “Buscar” **(E6)**;
21. O Sistema preenche os campos CEP, Logradouro, Bairro, UF e Cidade;
22. O Usuário insere número;
23. O Usuário insere complemento;
24. O Usuário seleciona o botão “Cadastrar” **(A1)**;
25. O Sistema salva as informações no banco de dados;
26. O caso de uso é finalizado.

Fluxos Alternativos

A1. Botão Cancelar.

1. O Usuário seleciona o botão “Cancelar”;
2. O Sistema cancela as edições e retorna para a tela home;
3. O caso de uso é finalizado.

Fluxos de Exceção

E1. Data Inválida:

1. O sistema consiste a data;
2. O sistema retorna a mensagem “Data inválida”;
3. O caso de uso é reiniciado.

E2. E-mail Inválido:

4. O sistema consiste o e-mail;
5. O sistema retorna a mensagem “E-mail inválido! Por favor, insira um e-mail válido”;
6. O caso de uso é reiniciado.

E3. Senha Inválida:

1. O sistema consiste a senha;
2. O sistema retorna a mensagem “Senha inválida! Por favor, insira uma senha com no mínimo cinco caracteres”;
3. O caso de uso é reiniciado.

E4. Senha diferente:

1. O sistema consiste a senha;
2. O sistema retorna a mensagem “Senha diferente! Por favor, insira a senha igual à anterior”;
3. O caso de uso é reiniciado.

E5. Arquivo inválido:

1. O sistema consiste o arquivo;
2. O sistema retorna a mensagem “imagem inválida”;
3. O caso de uso é continuado.

E6. CEP inválido:

4. O sistema consiste o CEP;
5. O sistema retorna a mensagem “CEP inválido! Por favor, insira um CEP válido”;
6. O caso de uso é reiniciado.

Especificação de Caso de Uso

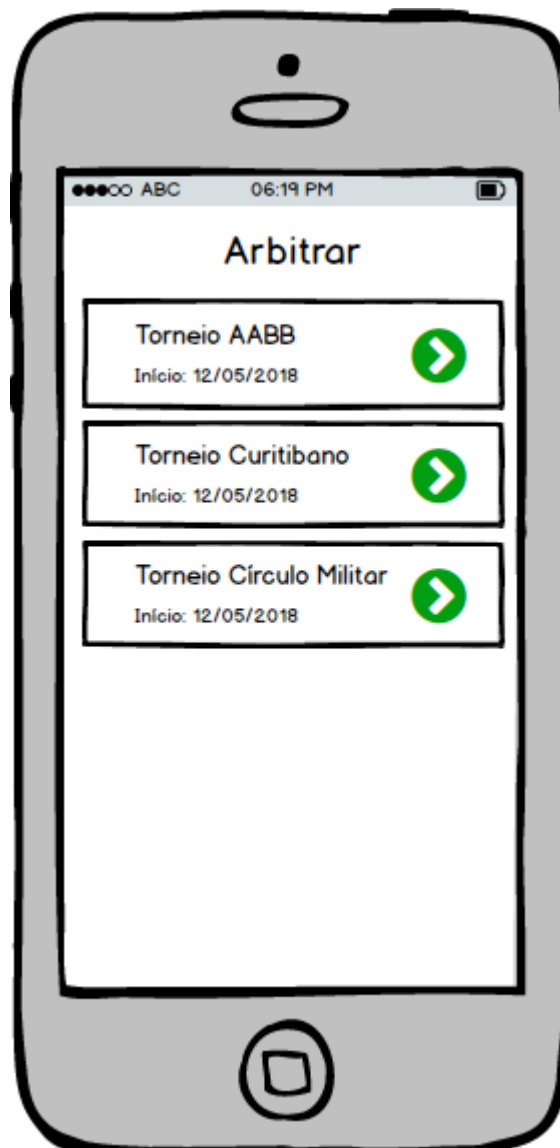
UC03 – Arbitrar Partida

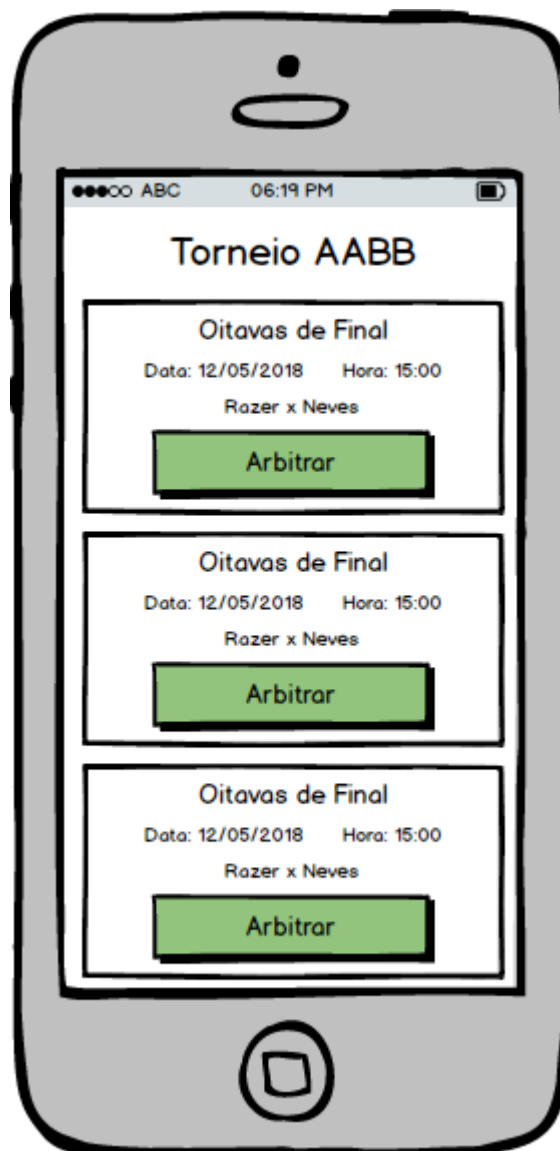
Descrição

Este caso de uso serve para arbitrar partidas em torneios de tênis.

Data View

DV1 – Arbitrar Partida Torneios



DV2 – Arbitrar Partida Torneio Detalhes**Pré-condições**

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC01 – Logar

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Ir para a tela de pontuação, permitindo assinalar as jogadas;
2. Registrar os dados da partida que foi arbitrada.

Ator Primário

Usuário.

Fluxo de Eventos Principal

1. O Sistema carrega do banco de dados os torneios disponíveis para arbitrar;
2. O Sistema apresenta a tela **(DV1)**;
3. O Usuário seleciona o torneio que irá arbitrar;
4. O Sistema apresenta a tela **(DV2)**;
5. O Usuário seleciona a partida que irá arbitrar;
6. O Sistema carrega o UC04 – Monitorar partida, na qual o árbitro usará para monitorar a partida;
7. O caso de uso é finalizado.

Especificação de Caso de Uso

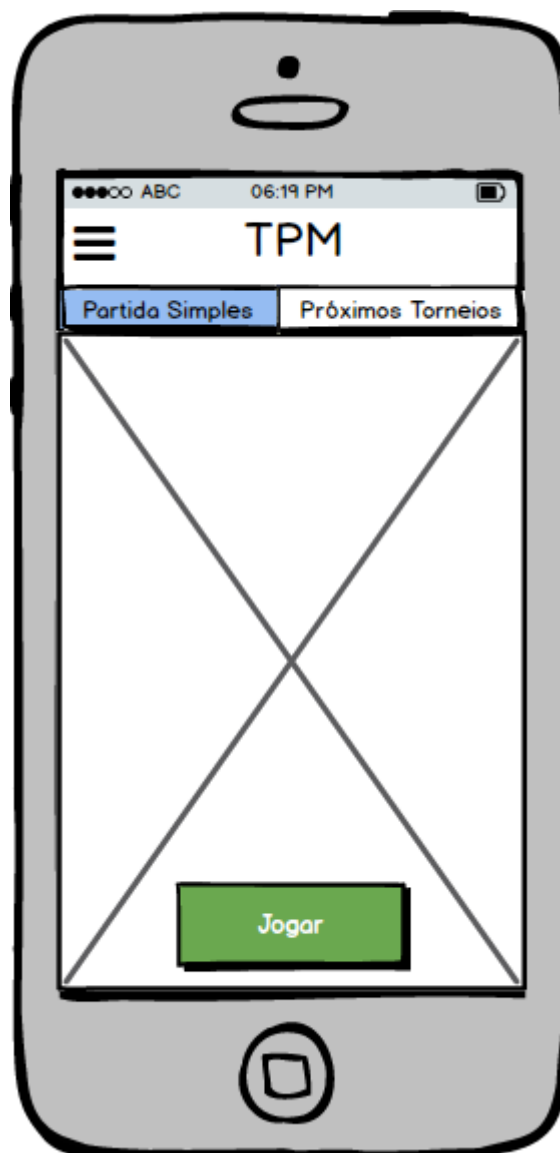
UC04 – Monitorar Partida

Descrição

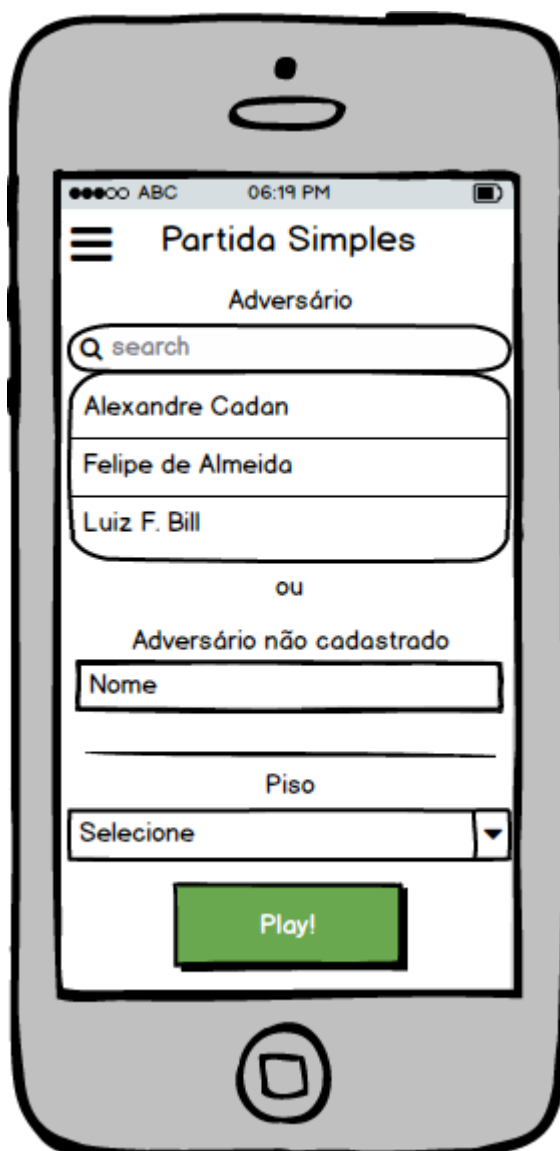
Este caso de uso serve para monitorar partidas amistosas de tênis.

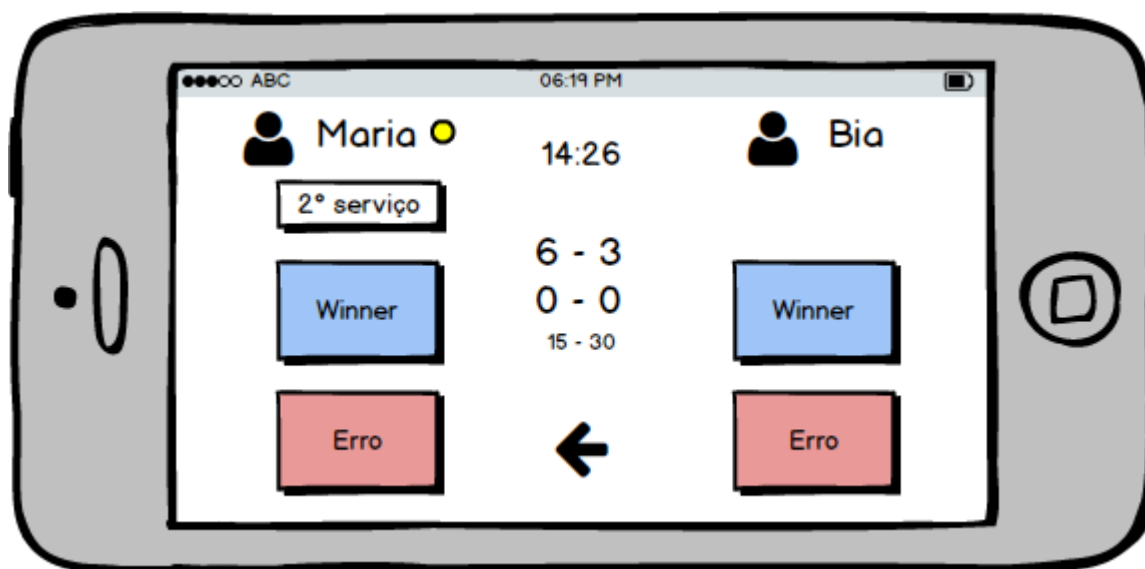
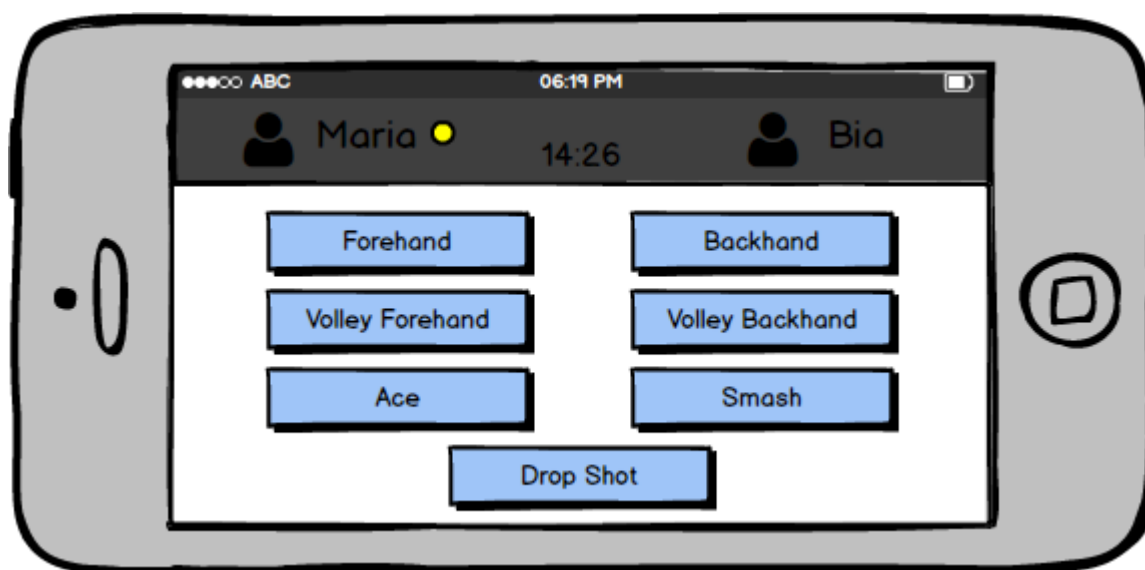
Data View

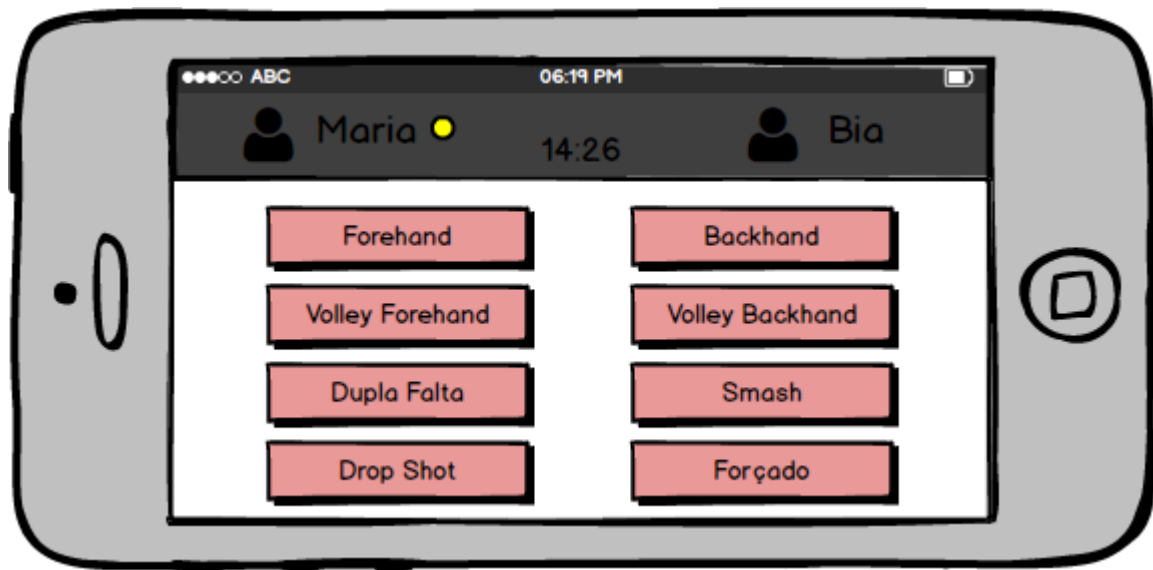
DV1 – Monitorar Partida Início



DV2 – Monitorar Partida Detalhes



DV3 – Monitorar Partida Pontuação**DV4 – Monitorar Partida Acertos**

DV5 – Monitorar Partida Erros**Pré-condições**

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC01 – Logar

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Registrar os dados da partida que foi jogada.

Ator Primário

Usuário.

Fluxo de Eventos Principal

1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário seleciona “Jogar”;
3. O Sistema apresenta a tela **(DV2)**;
4. O Usuário preenche o nome do adversário no campo pesquisa “Adversário”; **(A1)**.
5. O Usuário seleciona o adversário;
6. O Usuário seleciona o combo box “Piso”;
7. O Usuário seleciona um tipo de piso;
8. O Usuário seleciona o botão “Play”;
9. O Sistema apresenta a tela **(DV3)**;
10. O Usuário seleciona o botão “Winner” **(A2) (A3)**;
11. O Sistema apresenta a tela **(DV4)**;
12. O Usuário seleciona o tipo de acerto que foi realizado;
13. O Sistema apresenta a tela **(DV3)**;
14. O Sistema atualiza a tela **(DV3)** com o acerto assinalado pelo Usuário na tela **(DV4)**;
15. O Sistema contabiliza os pontos da partida até que um Usuário vença dois sets;
16. O caso de uso é finalizado.

Fluxos Alternativos

A1: Adversário não cadastrado.

1. O Usuário digita no campo “Adversário não cadastrado” o nome do oponente que não tem conta no aplicativo TPM;
2. O Sistema preenche o nome do adversário;
3. O Caso de Uso é mantido.

A2: Botão “Erro”.

1. O Sistema apresenta a tela **(DV5)**;
2. O Usuário seleciona o tipo de erro que foi realizado;
3. O Sistema apresenta a tela **(DV3)**;
4. O Sistema atualiza a tela **(DV3)** com o erro assinalado pelo Usuário na tela **(DV5)**;
5. O Sistema contabiliza os pontos da partida até que um Usuário vença dois sets;
6. O caso de uso é finalizado.

A3: Botão “Voltar”

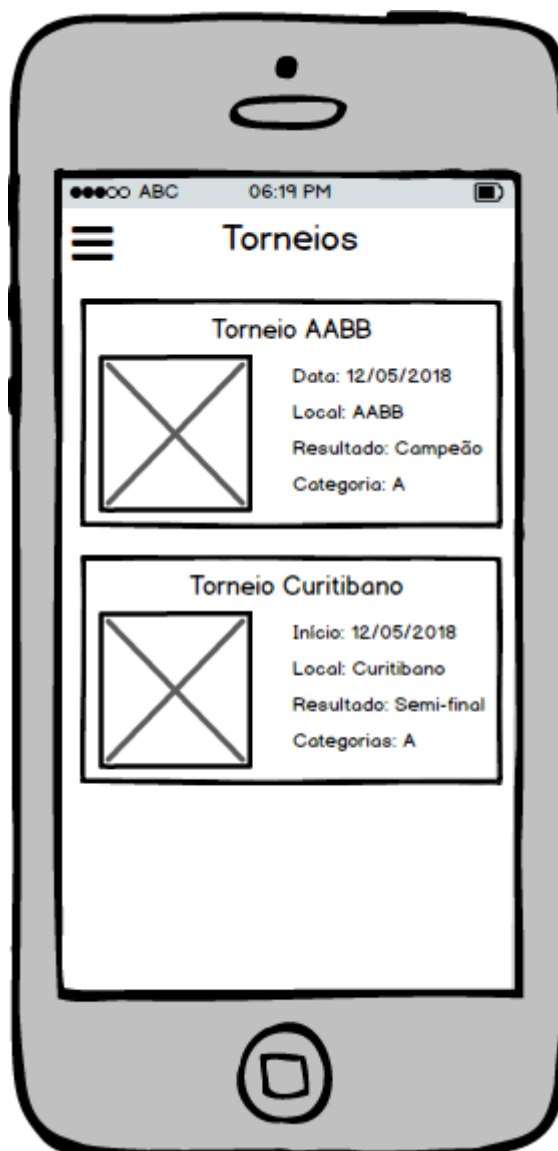
1. O Usuário seleciona o botão “Voltar”;
2. O Sistema desmarca a última pontuação assinalada;
3. O Caso de Uso é mantido.

Especificação de Caso de Uso

UC05 – Inscrever-se em Torneio

Descrição

Este caso de uso serve para inscrever o usuário em torneios de tênis.

Data View**DV1 – Visualizar Torneios**

DV2 – Confirmar Inscrição em Torneio**Pré-condições**

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC01 – Logar

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Registrar a inscrição do Usuário no torneio.

Ator Primário

Usuário.

Fluxo de Eventos Principal

1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário seleciona o torneio que deseja participar;
3. O Sistema apresenta a tela **(DV2)**;
4. O Usuário seleciona o botão “Inscrever”;
5. O Sistema salva as informações do Usuário no torneio que foi solicitado;
6. O caso de uso é finalizado.

Especificação de Caso de Uso

UC06 – Visualizar Conquistas

Descrição

Este caso de uso serve para visualizar conquistas do usuário adquiridas através do aplicativo.

Data View

DV1 – Visualizar Conquistas



DV2 – Visualizar Detalhes Conquista**Pré-condições**

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC01 – Logar

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Apresentar as conquistas e detalhes de conquistas do Usuário.

Ator Primário

Usuário.

Fluxo de Eventos Principal

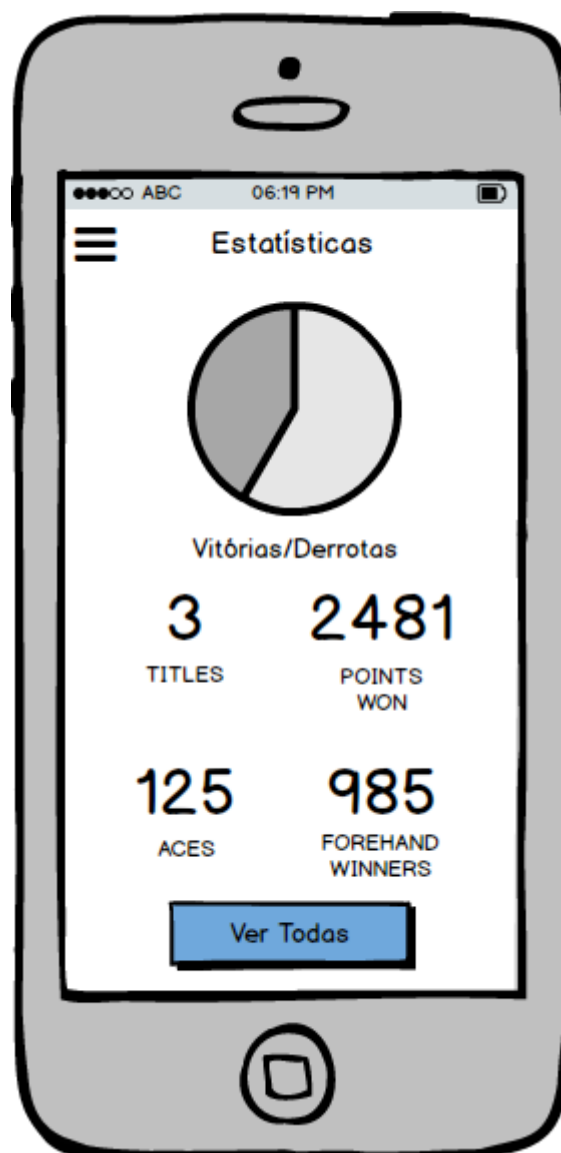
1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário seleciona a conquista que deseja verificar;
3. O Sistema apresenta a tela **(DV2)**;
4. O Usuário visualiza os detalhes da conquista selecionada;
5. O caso de uso é finalizado.

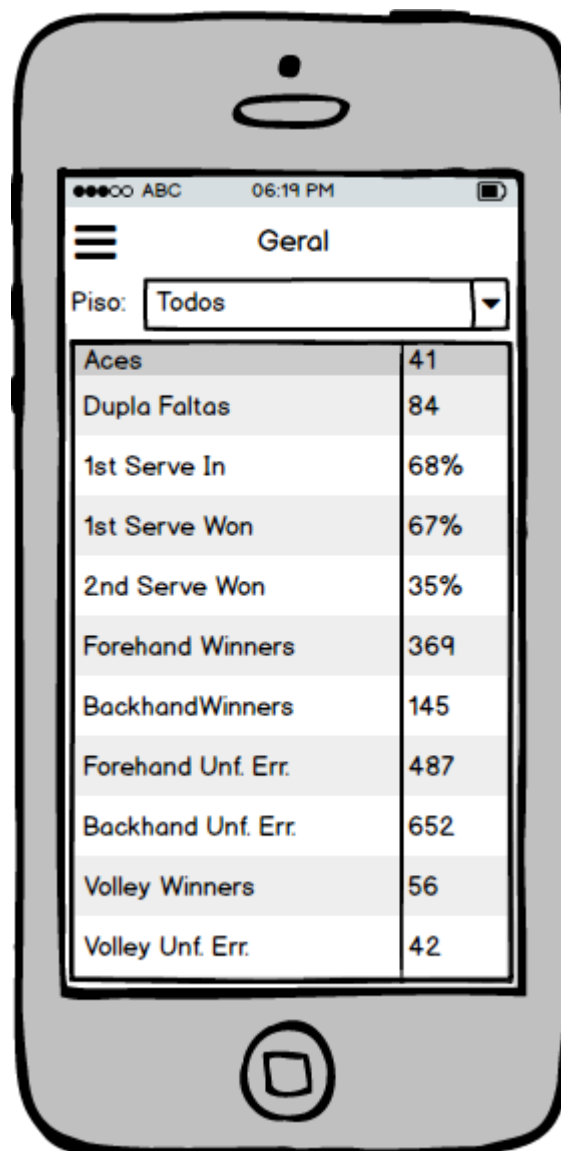
Especificação de Caso de Uso

UC07 – Visualizar Estatísticas

Descrição

Este caso de uso serve para visualizar as estatísticas das partidas assinaladas pelo aplicativo.

Data View**DV1 – Visualizar Estatísticas**

DV2– Visualizar Detalhes Estatísticas**Pré-condições**

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC01 – Logar

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Apresentar as estatísticas e detalhes de estatísticas do Usuário.

Ator Primário

Usuário.

Fluxo de Eventos Principal

1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário seleciona o botão “Ver todas”;
3. O Sistema apresenta a tela **(DV2)**;
4. O Usuário seleciona o combo box “Piso”;
5. O Usuário seleciona um tipo de piso;
6. O Sistema apresenta os detalhes das estatísticas de acordo com o piso;
7. O Usuário visualiza os detalhes da conquista;
8. O caso de uso é finalizado.

WEB

Especificação de Caso de Uso

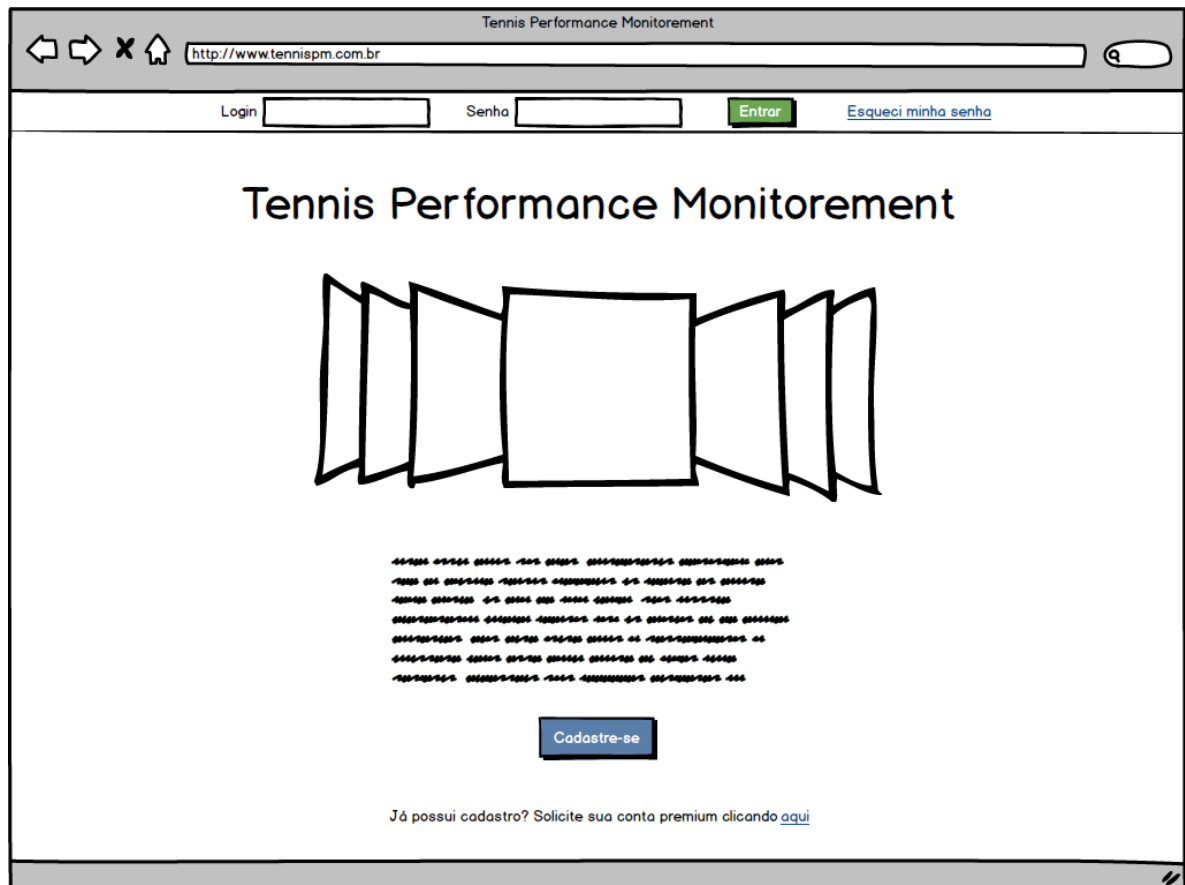
UC01 – Efetuar Login

Descrição

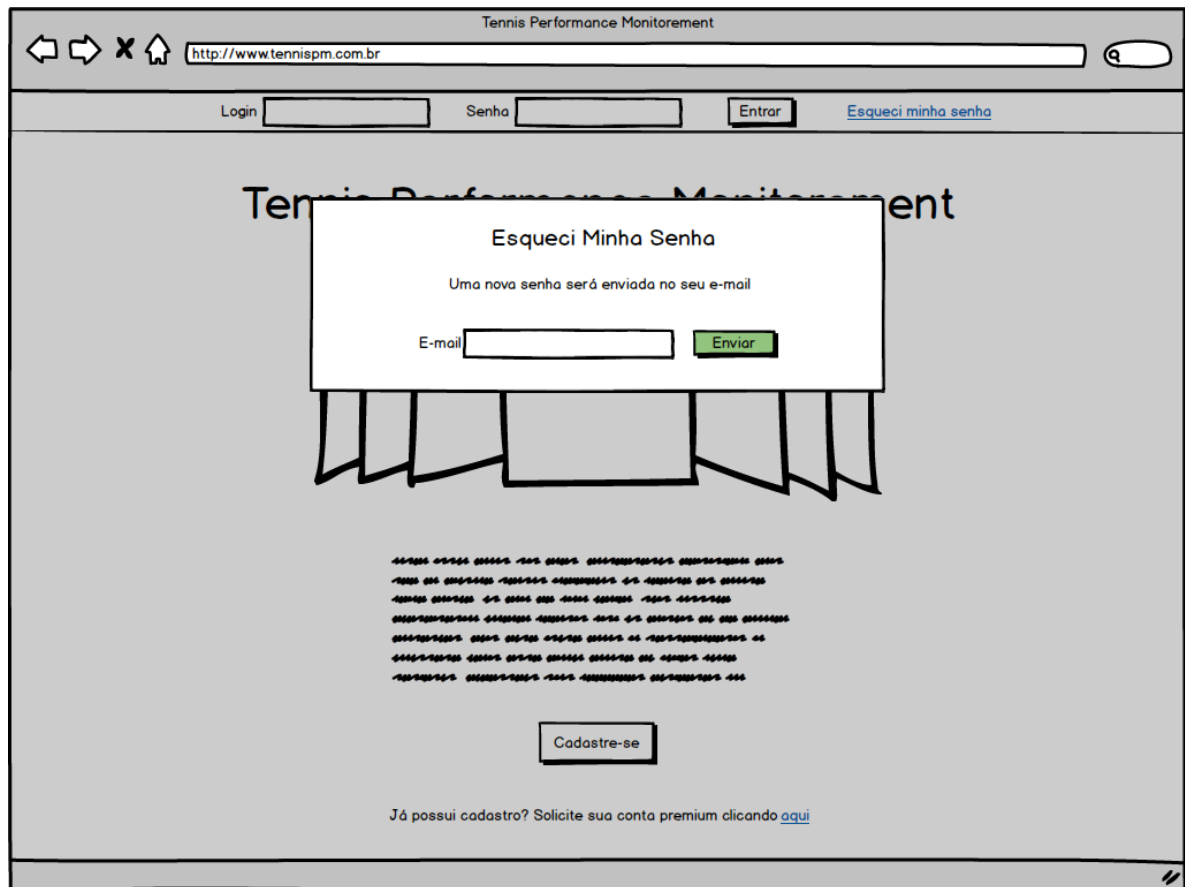
Este caso de uso serve para que um Usuário Premium ou Administrador efetue login no sistema TPM web.

Data View

DV1 - Tela de login.



DV2 - Tela de Recuperação de Senha.



Pré-condições

1. Ter login e senha registrados no sistema.

Pós-condições

Após o fim normal deste caso de uso sistema deve:

1. Ter autenticado o usuário permitindo o acesso ao sistema.

Ator Primário

Administrador ou Usuário Premium.

Fluxo de Eventos Principal

1. O Sistema apresenta a Tela de Logar **(DV1)**;
2. O Usuário insere seu e-mail no campo login;
3. O Usuário insere sua senha no campo senha;
4. O Usuário clica no botão “Entrar” **(A1)(A2)(A3)**;
5. O Sistema valida o usuário com o banco de dados **(E1)(E2)**;
6. O Sistema redireciona o usuário;
7. O caso de uso é finalizado.

Fluxos Alternativos

A1: Recuperação de senha.

1. O Usuário clica no botão “Esqueci minha senha”;
2. O Sistema apresenta a tela Recuperação de Senha **(DV2)**;
3. O Usuário informa seu e-mail de acesso ao sistema;
4. O Usuário clica no botão “Enviar”;
5. O Sistema verifica a existência do e-mail cadastrado no banco de dados **(E1)**;
6. O Sistema envia para o e-mail do usuário um token de recuperação de senha;
7. O Sistema emite a mensagem “Foi enviado ao seu e-mail um link para redefinição de senha”;
8. O Sistema redireciona para a tela de Logar **(DV1)**;
9. O caso de uso é reiniciado.

A2: Entrar com Facebook.

1. O Usuário clica no botão “Entrar com Facebook”;
2. O Sistema redireciona o usuário **(A4)**;
3. O caso de uso é finalizado.

A3: Cadastrar Usuário.

1. O Usuário clica no botão “Cadastre-se!”;
2. O Sistema carrega o UC002 – Cadastrar;
3. O caso de uso é finalizado.

A4: Cadastrar com Facebook.

1. O Sistema carrega informações do Facebook no UC02 – Cadastrar;
2. O Usuário preenche as informações que faltam;
3. O caso de uso é finalizado.

Fluxos de Exceção

E1. E-mail Inválido:

1. O sistema consiste o e-mail;
2. O sistema retorna a mensagem “E-mail inválido! Por favor, insira um e-mail válido”;
3. O caso de uso é reiniciado.

E2. Senha Inválida:

1. O sistema consiste a senha;
2. O sistema retorna a mensagem “Senha inválida! Por favor, insira uma senha válida”;
3. O caso de uso é reiniciado.

Especificação de Caso de Uso

UC02 – Realizar Cadastro

Descrição

Este caso de uso serve para cadastrar o usuário no sistema TPM web.

Data View

DV1 - Cadastro.

Pré-condições

Este caso de uso pode iniciar somente se:

1. Não há pré-condições.

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Cadastrar os dados do Usuário no sistema WEB TPM.

Ator Primário

Usuário.

Fluxo de Eventos Principal

1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário insere Nome e Sobrenome;
3. O Usuário seleciona a opção de sexo no radio button Sexo;
4. O Usuário insere telefone;
5. O Usuário seleciona o campo Data de Nascimento **(E1)**;
6. O Usuário seleciona a Data de Nascimento;
7. O Usuário seleciona a opção no combo box Classe;
8. O Usuário insere um e-mail **(E2)**;
9. O Sistema valida o e-mail;
10. O Usuário insere uma senha **(E3)**;
11. O Sistema valida a senha;
12. O Usuário confirma a senha **(E4)**;
13. O Usuário seleciona o campo Foto de Perfil;
14. O Sistema apresenta o explorador de arquivos;
15. O Usuário seleciona uma imagem **(E5)**;
16. O Sistema carrega a imagem;
17. O Usuário insere um CEP e seleciona o botão “Buscar” **(E6)**;
18. O Sistema preenche os campos CEP, Logradouro, Bairro, UF e Cidade;
19. O Usuário insere número;
20. O Usuário insere complemento;
21. O Usuário seleciona o botão “Cadastrar” **(A1)**;
22. O Sistema salva as informações no banco de dados;
23. O caso de uso é finalizado.

Fluxos Alternativos

A1. Botão Cancelar.

4. O Usuário Premium seleciona o botão “Cancelar”;
5. O Sistema cancela as edições e retorna para a tela home;
6. O caso de uso é finalizado;

Fluxos de Exceção

E1. Data Inválida:

4. O sistema consiste a data;
5. O sistema retorna a mensagem “Data inválida”;
6. O caso de uso é reiniciado.

E2. E-mail Inválido:

4. O sistema consiste o e-mail;
5. O sistema retorna a mensagem “E-mail inválido! Por favor, insira um e-mail válido”;
6. O caso de uso é reiniciado.

E3. Senha Inválida:

1. O sistema consiste a senha;
2. O sistema retorna a mensagem “Senha inválida! Por favor, insira uma senha com no mínimo quatro caracteres”;
3. O caso de uso é reiniciado.

E4. Senha diferente:

1. O sistema consiste a senha;
2. O sistema retorna a mensagem “Senha diferente! Por favor, insira a senha igual à anterior”;
3. O caso de uso é reiniciado.

E5. Arquivo inválido:

1. O sistema consiste o arquivo;
2. O sistema retorna a mensagem “imagem inválida”;
3. O caso de uso é continuado.

E6. CEP inválido:

1. O sistema consiste o CEP;
2. O sistema retorna a mensagem “CEP inválido! Por favor, insira um CEP válido”;
3. O caso de uso é reiniciado.

Especificação de Caso de Uso

UC03 – Solicitar Conta Premium

Descrição

Este caso de uso serve para o usuário solicitar conta premium.

Data View

DV1 – Solicitar Conta Premium

The screenshot shows a web browser window titled "Tennis Performance Monitorement" with the URL "http://www.tennism.com.br". The browser's address bar and navigation buttons are visible. On the website, there is a login section with fields for "Login" and "Senha" (Password), an "Entrar" (Enter) button, and a link "Esqueci minha senha". A modal dialog titled "Solicitação de Conta Premium" is displayed in the center. It contains fields for "Login" and "Senha", a blue "Solicite" (Request) button, a Facebook icon, and a red "Cancelar" (Cancel) button. Below the modal, there is a "Cadastre-se" (Sign up) button and a link "aqui" (here) for users who already have an account.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC02 – Realizar Cadastro

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Enviar uma solicitação para o Administrador do Sistema Web.

Ator Primário

Usuário.

Fluxo de Eventos Principal

1. O Sistema apresenta a Tela de Solicitação **(DV1)**;
2. O Usuário insere seu e-mail no campo Login;
3. O Usuário insere sua senha no campo senha;
4. O Usuário clica no botão “Solicite” **(A1)(A2)(E1)(E2)**;
5. O Sistema envia uma solicitação para o administrador do sistema web;
6. O Sistema redireciona o usuário para a tela de login;
7. O sistema retorna a mensagem “Solicitação realizada com sucesso”;
8. O caso de uso é finalizado.

Fluxos Alternativos

A1. Botão Cancelar.

1. O Usuário seleciona o botão “Cancelar”;
2. O Sistema cancela as edições e retorna para a tela de login;
3. O caso de uso é finalizado.

A2. Botão Facebook

1. O Usuário seleciona o botão “Facebook”;
2. O Sistema realiza login com o acesso do Facebook;
3. O Sistema envia uma solicitação para o administrador do sistema web;
4. O Sistema redireciona o usuário para a tela de login;
5. O caso de uso é finalizado.

Fluxos de Exceção

E1. E-mail Inválido:

1. O sistema consiste o e-mail;
2. O sistema retorna a mensagem “E-mail inválido! Por favor, insira um e-mail válido”;
3. O caso de uso é reiniciado.

E2. Senha Inválida:

1. O sistema consiste a senha;
2. O sistema retorna a mensagem “Senha inválida! Por favor, insira uma senha com no mínimo cinco caracteres”;
3. O caso de uso é reiniciado.

Especificação de Caso de Uso

UC04 – Gerenciar Local

Descrição

Este caso de uso serve para gerenciar locais de torneio.

Data View

DV1 – Novo Local

The screenshot shows a web browser window titled 'Tennis Performance Monitorement' with the URL 'http://www.tennispm.com.br'. The browser's address bar and search bar are visible. The page header includes the site name, a user profile for 'James Guimaraes', and a 'Sair' link. A modal window titled 'Novo Local' is open, containing the following fields and controls:

- Nome**: Text input field.
- Telefone**: Text input field.
- Proprietário**: Dropdown menu with 'Selecione' as the current selection.
- E-mail**: Text input field.
- CEP**: Text input field with a 'Buscar' button next to it.
- Observações**: Text input field.
- Logradouro**: Text input field.
- Número**: Text input field.
- Complemento**: Text input field.
- Bairro**: Text input field.
- Cidade**: Text input field.
- UF**: Dropdown menu with 'Selecione' as the current selection.
- Fotos**: File upload area with an 'Arquivo' button.

At the bottom of the modal, there are two buttons: 'Cancelar' (red) and 'Cadastrar' (green).

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC01 – Logar

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Cadastrar ou editar um local de jogo.

Ator Primário

Administrador ou Usuário Premium.

Fluxo de Eventos Principal

1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário insere Nome e Telefone.
3. O Usuário seleciona a opção no combo box Proprietário **(R1)**;
4. O Usuário preenche o campo de texto Observações;
5. O Usuário insere um CEP e seleciona o botão “Buscar” **(E1)**;
6. O Sistema preenche os campos CEP, Logradouro, Bairro, UF e Cidade;
7. O Usuário insere número;
8. O Usuário insere complemento;
9. O Usuário seleciona o campo Fotos;
10. O Sistema apresenta o explorador de arquivos;
11. O Usuário seleciona as imagens **(E2)**;
12. O Sistema carrega as imagens;
13. O Usuário seleciona o botão “Cadastrar” **(A1)**;
14. O Sistema salva as informações no banco de dados;
15. O caso de uso é finalizado.

Fluxos Alternativos

A1. Botão Cancelar.

1. O Usuário Premium seleciona o botão “Cancelar”;
2. O Sistema cancela as edições e retorna para a tela home;
3. O caso de uso é finalizado.

Fluxos de Exceção

E1. CEP inválido:

1. O sistema consiste o CEP;
2. O sistema retorna a mensagem “CEP inválido! Por favor, insira um CEP válido”;
3. O caso de uso é reiniciado.

E2. Arquivo inválido:

1. O sistema consiste o arquivo;
2. O sistema retorna a mensagem “imagem inválida”;
3. O caso de uso é reiniciado.

Regras de Negócio

R1. Se for Administrador irá listar todos os usuários. E se for Premium irá apresentar apenas o perfil dele.

Especificação de Caso de Uso

UC05 – Gerenciar Torneio

Descrição

Este caso de uso serve para gerenciar um torneio.

Data View

DV1 – Novo Torneio

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC01 – Logar

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Cadastrar ou editar um novo torneio de jogo.

Ator Primário

Administrador ou Usuário Premium.

Fluxo de Eventos Principal

1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário insere Nome e Descrição;
3. O Usuário seleciona o campo Início;
4. O Usuário seleciona a data de início do torneio **(E1)**;
5. O Usuário seleciona o campo Fim;
6. O Usuário seleciona a data de fim do torneio **(R1) (E1)**;
7. O Usuário seleciona a opção no combo box Proprietário **(R2)**;
8. O Usuário seleciona o campo Data Limite de Inscrição **(R3) (E1)**;
9. O Usuário preenche o campo Premiações;
10. O Usuário Premium preenche o campo Valor Inscrição;
11. O Usuário seleciona o campo Fotos;
12. O Sistema apresenta o explorador de arquivos;
13. O Usuário seleciona as imagens **(E2)**;
14. O Sistema carrega as imagens;
15. O Usuário seleciona o botão “Cadastrar” **(A1)**;
16. O Sistema salva as informações no banco de dados;
17. O caso de uso é finalizado.

Fluxos Alternativos

A1. Botão Cancelar.

1. O Usuário Premium seleciona o botão “Cancelar”;
2. O Sistema cancela as edições e retorna para a tela home;
3. O caso de uso é finalizado.

Fluxos de Exceção

E1. Data Inválida:

1. O sistema consiste a data;
2. O sistema retorna a mensagem “Data inválida”;
3. O caso de uso é reiniciado.

E2. Arquivo inválido:

1. O sistema consiste o arquivo;
2. O sistema retorna a mensagem “imagem inválida”;
3. O caso de uso é reiniciado.

Regras de Negócio

R1. A data fim do torneio tem que ser maior que a data de início.

R2. Se for Administrador irá listar todos os usuários. E se for Premium irá apresentar apenas o perfil dele.

R3. O limite de inscrição tem que ser menor que a data de início.
Especificação de Caso de Uso

UC06 – Gerar Chave

Descrição

Este caso de uso serve para gerar e mostrar as chaves do torneio.

Data View

DV1 – Gerar Chave

Tennis Performance Monitorement

Home Usários Locais **Torneios** Relatórios

Torneios

Novo Torneio

10 Resultados por página

1 search

Nome	Data Início	Limite Inscrição	Total Insritos		
Happy Camp	10/10/2017	01/10/2017	8	+	-
Pro Players	17/09/2017	08/09/2017	16	i	-
Man Camps	25/09/2017	20/09/2017	16	i	-

Ant 1 2 3 Pró

DV2 – Chaves de torneio

Tennis Performance Monitorement

http://www.tennispm.com.br

James Guimaraes Sair

Home Usários Locais **Torneios** Relatórios

Torneios

Novo Torneio

10 Resultados por página

1 search

Nome	Data Início	Limite Inscrição	Total Insritos		
Happy Camp	10/10/2017	01/10/2017	8	i	-
Pro Players	17/09/2017	08/09/2017	16	i	-
Man Camps	25/09/2017	20/09/2017	16	i	-

Ant. 1 2 3 Pró

DV3 – Informações chave torneio

Tennis Performance Monitorement

http://www.tennispm.com.br

James Guimaraes Sair

Home Usários Locais **Torneios** Relatórios

Informações do Torneio

Nome Data Inicial Data Final Categorias

Happy Camp 10/10/2017 14/10/2017 10 Resultados por página

Descrição Descrição Prêmio

Vamos lá meninas Campeonato Fem. R\$ 00,00 i

Realizador

Luiz Bill

Inscritos

8 Resultados por página

Ana Josefina	Beatriz Mara	Luisa Santos
Maria Antunes	Eduarda Sis	Maria Dias
Paula Julia	Josefina Tau	

DV4 – Chave torneio

The screenshot shows a web browser window with the URL <http://www.tennism.com.br>. The page title is "Tennis Performance Monitorement". The user is logged in as "James Guimaraes" and can click "Sair" to log out. The navigation menu includes "Home", "Usuários", "Locais", "Torneios" (selected), and "Relatórios".

The "Chave Torneio" (Tournament Bracket) window is displayed, showing details for the "Happy Camp" tournament. The dates are "10/10/2017 até 12/10/2017" and the category is "Feminino, até 12 anos, A".

1ª rodada	quartas	semi	final
Maria			
Ana			
Paula			
Beatriz			
Eduarda			
Josefina			
Luisa			
Maria Dias			

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC01 – Logar

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Cadastrar e apresentar a chave do torneio

Ator Primário

Administrador ou Usuário Premium.

Fluxo de Eventos Principal

1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário seleciona o botão “Gerar chave” **(R1) (R2) (A1)**;
3. O Sistema apresenta uma mensagem para confirmar a geração de chave;
4. O Usuário confirma a geração de chave;
5. O Sistema apresenta a tela **(DV2)**;
6. O Usuário seleciona o botão “Informações” **(A1)**;
7. O Sistema apresenta a tela **(DV3)**;
8. O Usuário seleciona o botão “Chave do torneio”;
9. O Sistema apresenta a tela **(DV4)**;
10. O caso de uso é finalizado

Fluxos Alternativos

A1. Botão Desativar.

1. O Premium seleciona o botão “Desativar”.
2. O Sistema apresenta uma mensagem para confirmar a desativação do torneio;
3. O Premium confirma a desativação do torneio;
4. O Sistema desativa o torneio;
5. O caso de uso é finalizado.

Regras de Negócio

R1. Só poderá gerar chave após a data limite de inscrição e ter inscritos no torneio.

R2. Só poderá desativar torneio se não tiver gerado chave e não ter iniciado a competição.

Especificação de Caso de Uso

UC07 – Gerar Relatório

Descrição

Este caso de uso serve para gerar relatórios do torneio.

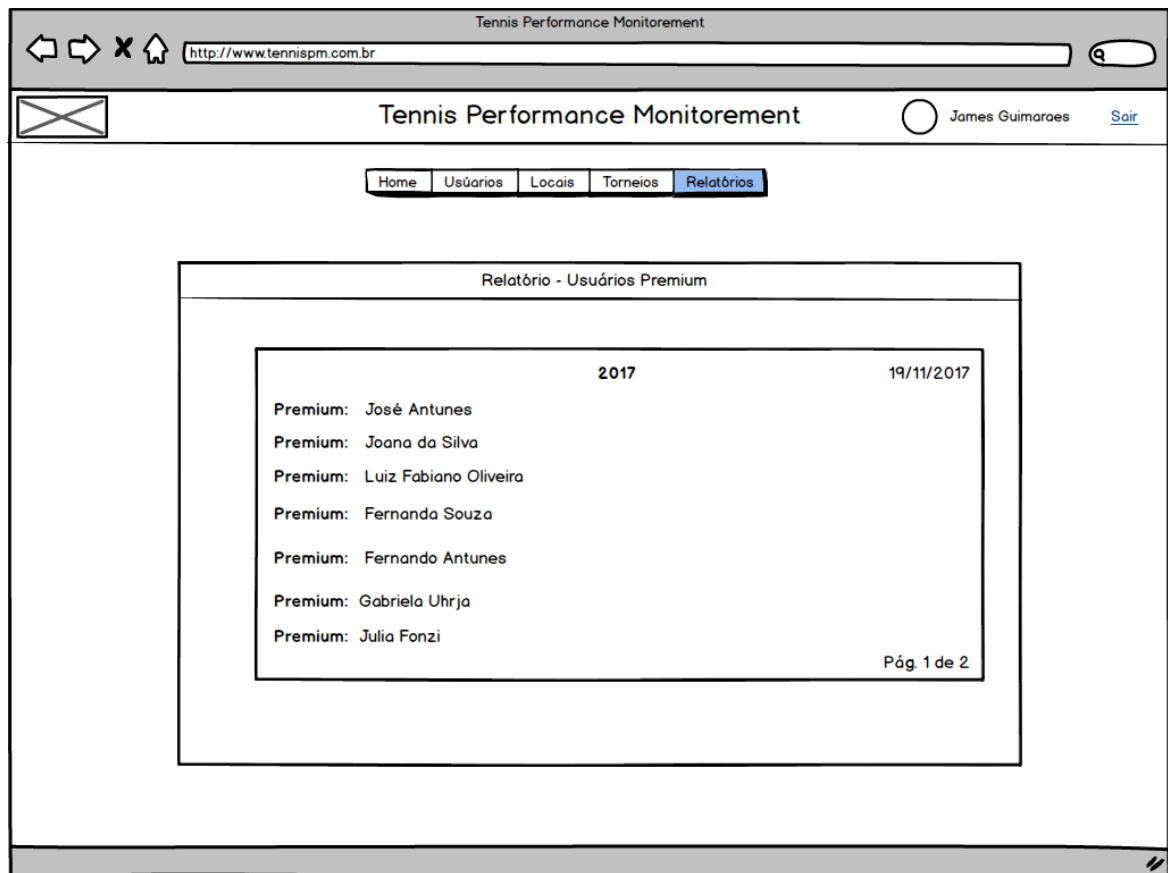
Data View

DV1 – Gerar Relatório

The screenshot displays a web browser window titled "Tennis Performance Monitorement" with the URL "http://www.tennispmm.com.br". The page features a navigation bar with links: Home, Usuários, Locais, Torneios, and Relatórios (highlighted). The main content area is titled "Relatórios" and contains four report generation forms:

- Gerar relatório de instalações filtrando por localização:** Includes dropdowns for "Estado" and "Cidade", both with "Selecione" as the placeholder, and a "Gerar" button.
- Gerar relatório de instalações filtrando por data:** Includes dropdowns for "Data Inicia" (placeholder "dd/mm/aaaa") and "Data Final" (placeholder "dd/mm/yyyy"), and a "Gerar" button.
- Gerar relatório detalhado do torneio:** Includes dropdowns for "Torneio" (placeholder "Selecione") and another "Selecione" dropdown, and a "Gerar" button.
- Gerar relatório de usuários premium em um intervalo:** Includes a dropdown for "Ano" with "2017" selected, and a "Gerar" button.

DV2 – Relatório



Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC01 – Logar

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Gerar um novo relatório do torneio.

Ator Primário

Administrador ou Usuário Premium.

Fluxo de Eventos Principal

1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário seleciona o filtro, nos combobox disponíveis, que irá usar no relatório;
3. O Usuário seleciona o botão “Gerar”;
4. O Sistema apresenta a tela **(DV2)**;
5. O caso de uso é finalizado.

Especificação de Caso de Uso

UC08 – Gerenciar Usuário

Descrição

Este caso de uso serve para gerenciar usuários.

Data View

DV1 – Editar Informações Usuário

The screenshot shows a web browser window with the URL <http://www.tennispm.com.br>. The page title is "Tennis Performance Monitorement". In the top right corner, the user "James Guimaraes" is logged in, with a "Sair" (Logout) link. A modal window titled "Detalhes" is open, displaying a form for editing user information. The form includes a profile picture placeholder and the following fields:

- Nome:** Felipe
- Sobrenome:** Macedo
- Email:** femado@gmail.com
- Data Nasc.:** / / (with a calendar icon)
- Sexo:** ☒ M ☐ F
- Telefone:** (empty field)
- CEP:** (empty field) with a "Buscar" (Search) button
- Logradouro:** (empty field)
- Número:** (empty field)
- Complemento:** (empty field)
- Bairro:** (empty field)
- Cidade:** (empty field)
- UF:** Selezione (dropdown menu)

At the bottom of the form, there are "Permissões" (Permissions) with checkboxes for ☒ Básico and ☐ Premium. At the very bottom of the modal are two buttons: "Cancelar" (Cancel) and "Salvar" (Save).

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC01 – Logar

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Editar as informações de um usuário.

Ator Primário

Administrador.

Fluxo de Eventos Principal

1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário insere Nome e Sobrenome.
3. O Usuário seleciona a opção no combo box Sexo;
4. O Usuário insere telefone;
5. O Usuário seleciona a Data de Nascimento **(E1)**;
6. O Usuário seleciona a opção no combo box Mão;
7. O Usuário seleciona a opção no combo box Classe;
8. O Usuário insere um e-mail **(E2)**;
9. O Sistema valida o e-mail;
10. O Usuário insere um CEP e seleciona o botão “Buscar” **(E3)**;
11. O Sistema preenche os campos CEP, Logradouro, Bairro, UF e Cidade;
12. O Usuário insere número;
13. O Usuário insere complemento;
14. O Usuário seleciona a opção de permissão “Básico” **(A1)**;
15. O Usuário seleciona o botão “Salvar” **(A2)**;
16. O Sistema salva as informações no banco de dados;
17. O caso de uso é finalizado.

Fluxos Alternativos

A1. Premium.

1. O Usuário seleciona a opção “Premium”;
2. O caso de uso é continuado.

A2. Botão Cancelar.

1. O Usuário seleciona o botão “Cancelar”;
2. O Sistema cancela as edições e retorna para a tela home;
3. O caso de uso é finalizado.

Fluxos de Exceção

E1. Data Inválida:

1. O sistema consiste a data;
2. O sistema retorna a mensagem “Data inválida”;
3. O caso de uso é reiniciado.

E2. E-mail Inválido:

1. O sistema consiste o e-mail;
2. O sistema retorna a mensagem “E-mail inválido! Por favor, insira um e-mail válido”;
3. O caso de uso é reiniciado.

E3. CEP inválido:

1. O sistema consiste o CEP;
2. O sistema retorna a mensagem “CEP inválido! Por favor, insira um CEP válido”;
3. O caso de uso é reiniciado.

DV2 – Visualizar Detalhes Requisição

The screenshot shows a web browser window titled "Tennis Performance Monitorement" with the URL "http://www.tennispm.com.br". The page has a navigation bar with links: Home, Usuários, Locais, Torneios, and Relatórios. The main content area is titled "Requisições". A modal window titled "Detalhes" is open, displaying a form for user details. The form includes fields for Nome, Sobrenome, CPF, Email, Telefone, Data Nasc. (with a calendar icon), Sexo (radio buttons for M and F), CEP, Logradouro, Número, Complemento, Bairro, Cidade, and UF (a dropdown menu labeled "Selecione"). A "Buscar" button is located next to the CEP field.

Pré-condições

Este caso de uso pode iniciar somente se:

1. O sistema tiver executado o UC001 – Logar

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Aprovar ou recusar requisições de usuários.

Ator Primário

Administrador.

Fluxo de Eventos Principal

1. O Sistema apresenta a tela **(DV1)**;
2. O Usuário seleciona o botão “Confirmar” **(A1) (A2)**;
3. O Sistema apresenta uma mensagem de confirmação;
4. O Usuário seleciona a opção “Sim”. **(A3)**;
5. O Sistema concede permissão Premium;
6. O caso de uso é finalizado.

Fluxos Alternativos

A1. Botão “Excluir”.

1. O Usuário seleciona o botão “Excluir”;
2. O Sistema exclui a requisição do Usuário;
3. O caso de uso é finalizado.

A2. Botão “Informações”.

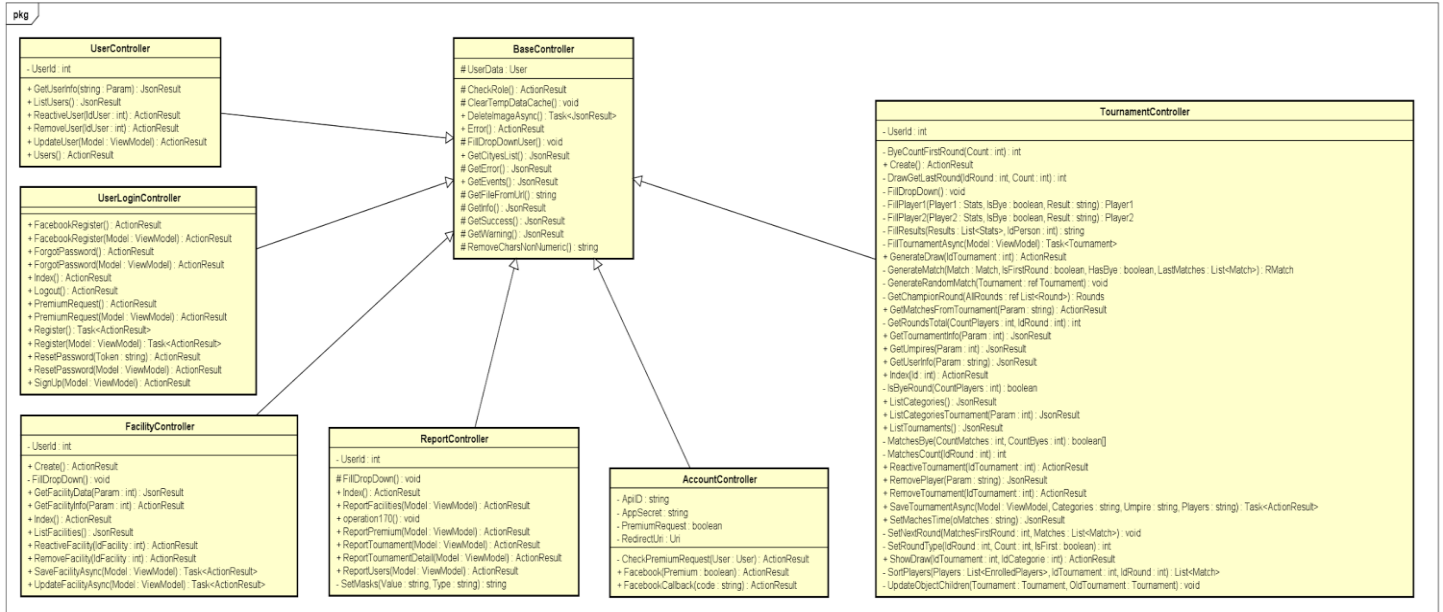
1. O Usuário seleciona o botão “Informações”;
2. O Sistema apresenta a tela **(DV2)**;
3. O Usuário seleciona o botão “Confirmar” **(A1)**;
4. O Sistema apresenta uma mensagem de confirmação;
5. O Usuário seleciona a opção “Sim”. **(A3)**;
6. O Sistema valida a confirmação e altera a permissão do Usuário para Usuário Premium;
7. O caso de uso é finalizado.

A3. Botão “Não”.

1. O Usuário seleciona o botão “Não”;
2. O Sistema retorna para a tela **(DV1)**;
3. O caso de uso é finalizado.



Diagrama de Classe – APP Controller



Realiza conexão com as classes presentes no diagrama



Diagrama de Classe – WEB DAO

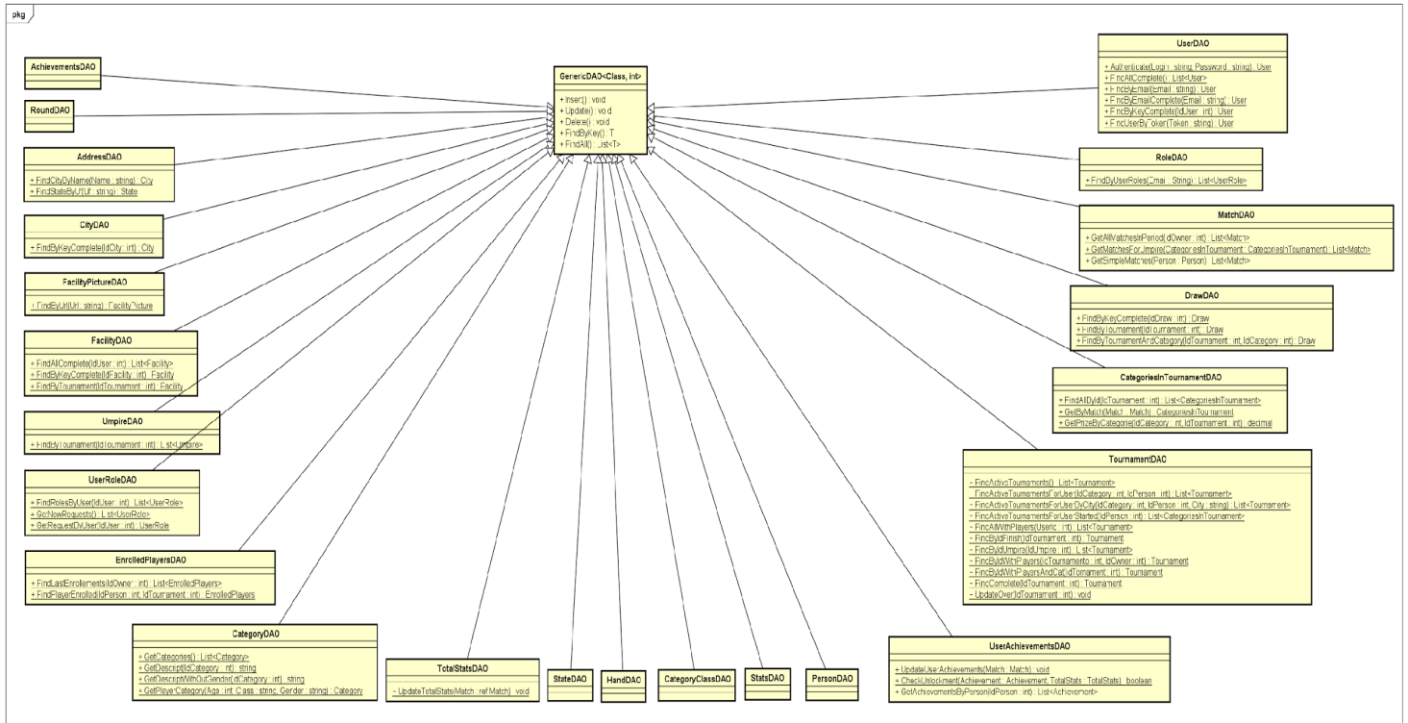
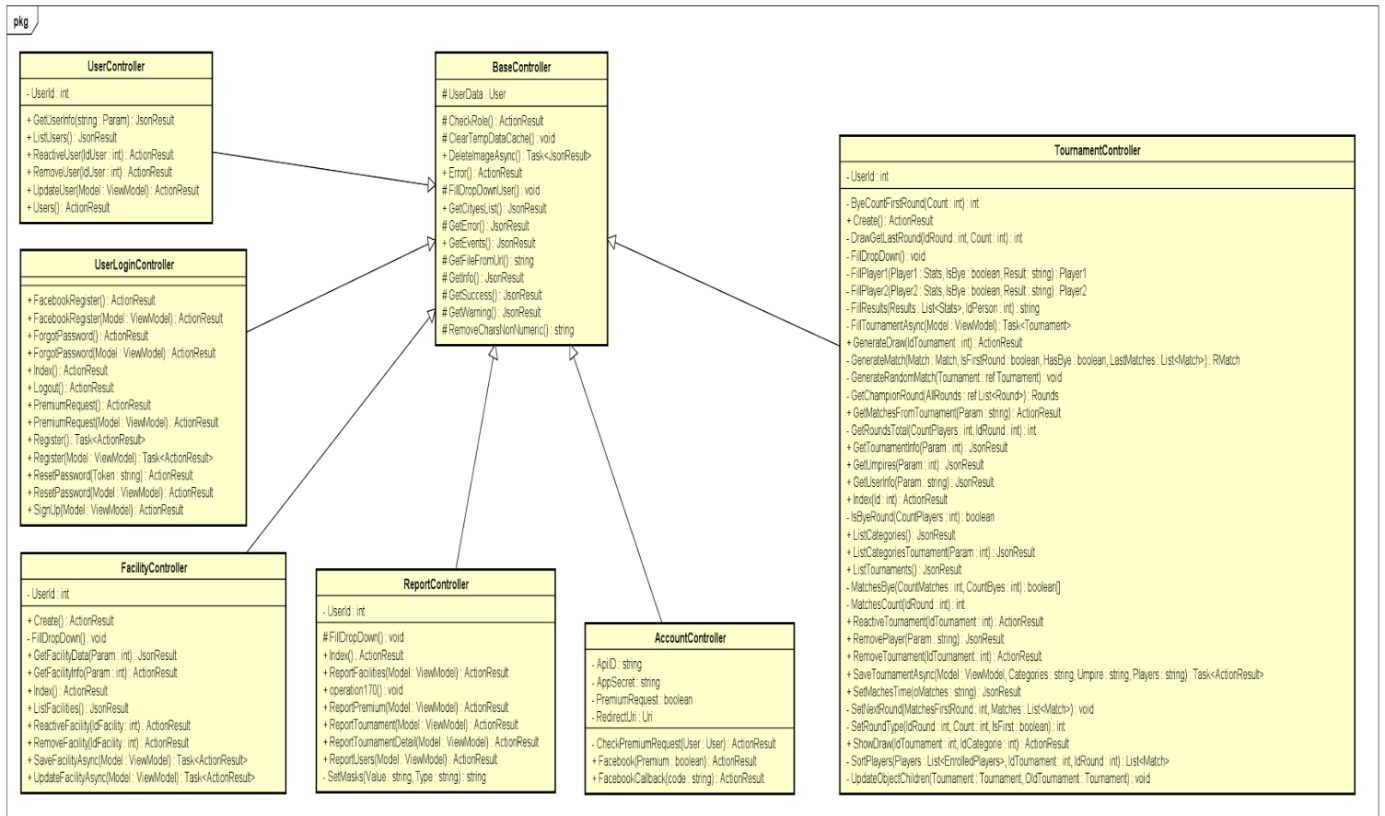
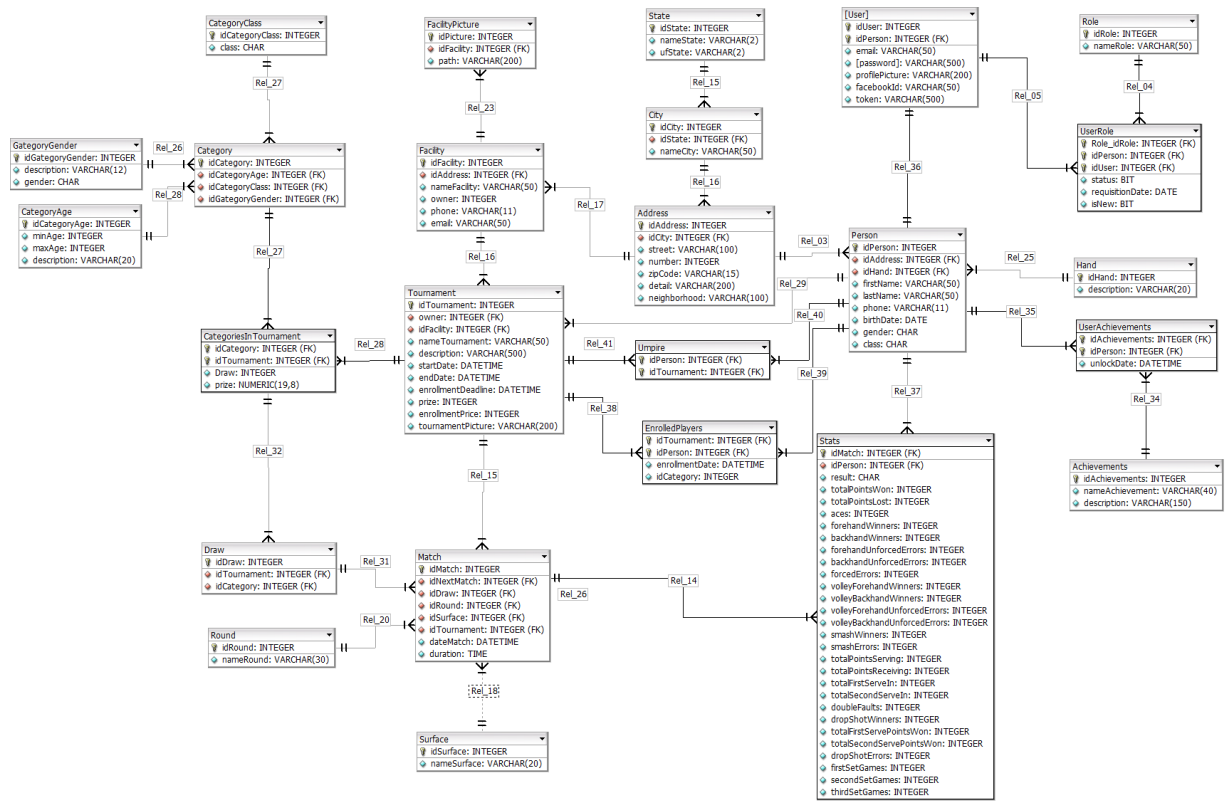


Diagrama de Classe – WEB Controller



APÊNDICE D – DIAGRAMA ENTIDADE RELACIONAMENTO

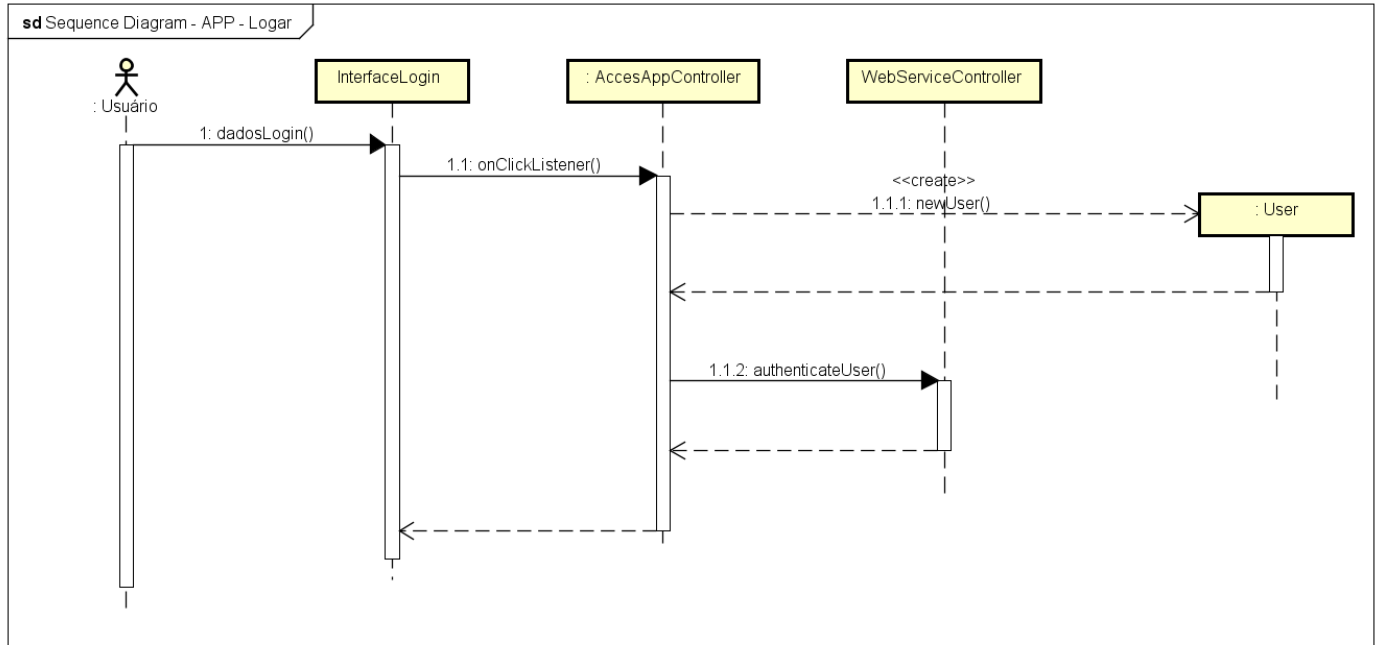


APÊNDICE E – PLANO DE COMUNICAÇÃO

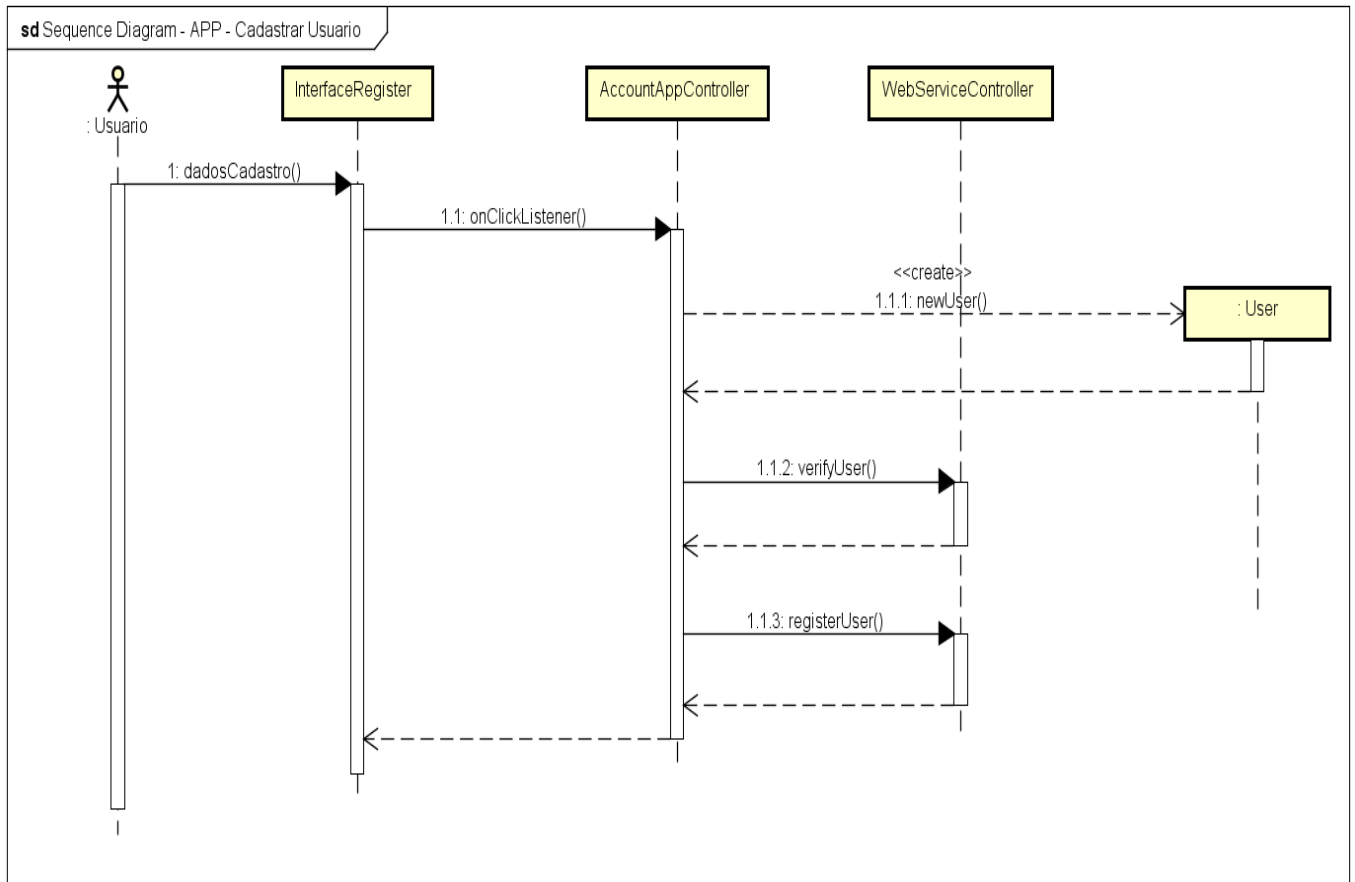
Grupo de Interessados	Foco	O que este grupo precisa saber?	Método	Quando?
INFORMAÇÕES INTERNAS DO PROJETO				
Equipe do projeto	Informações sobre o andamento do projeto	Andamento das atividades, dificuldades, impedimentos	Reunião	Diariamente
Equipe do projeto	Auxílio em atividades	Compartilhar conhecimentos para o desenvolvimento de atividades muito grandes ou difíceis	Reunião	Quando necessário
Equipe do projeto	Repasse de informações dos usuários	Compartilhar informações recebidas de futuros usuários do sistema	Reunião ou conversa pelo grupo do WhatsApp	Quando necessário
INFORMAÇÕES EXTERNAS AO PROJETO				
Equipe do projeto e professor orientador	Informações sobre o andamento do projeto	Andamento do projeto	Reunião	Semanalmente

APÊNDICE F – DIAGRAMA DE SEQUÊNCIA

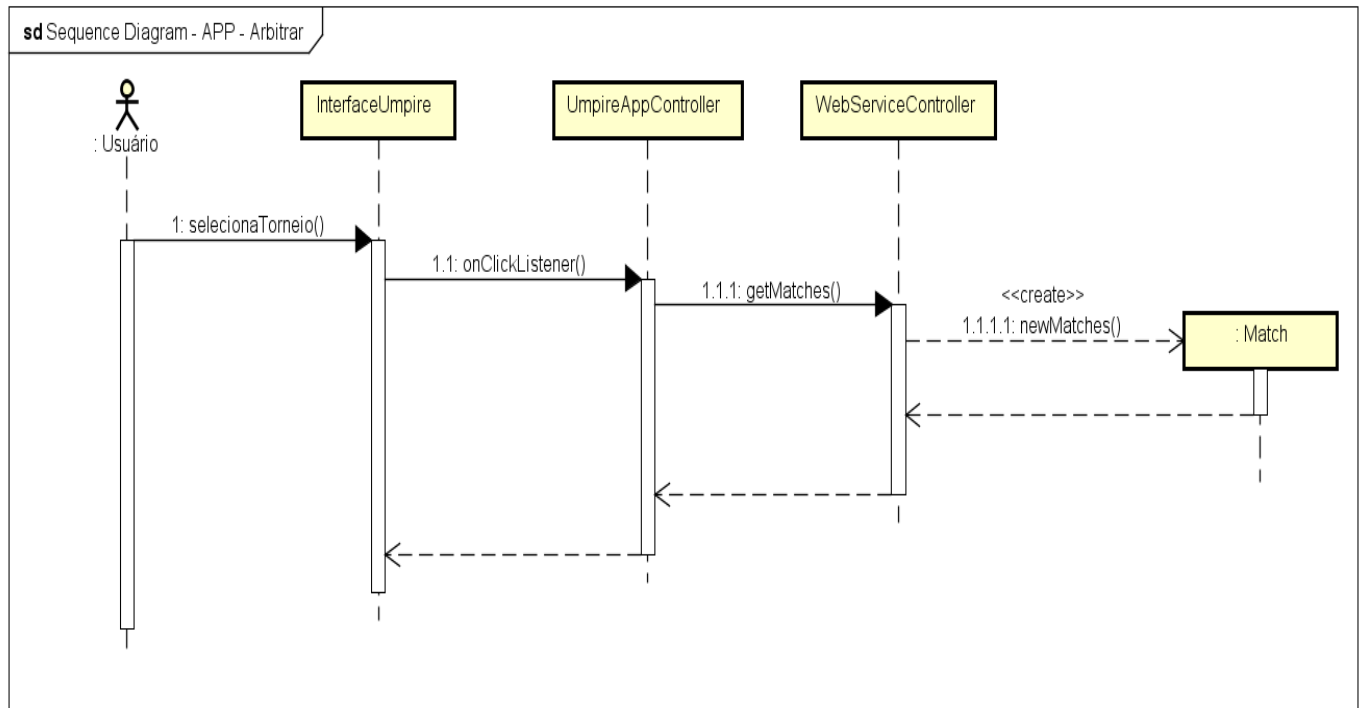
APP – Efetuar Login



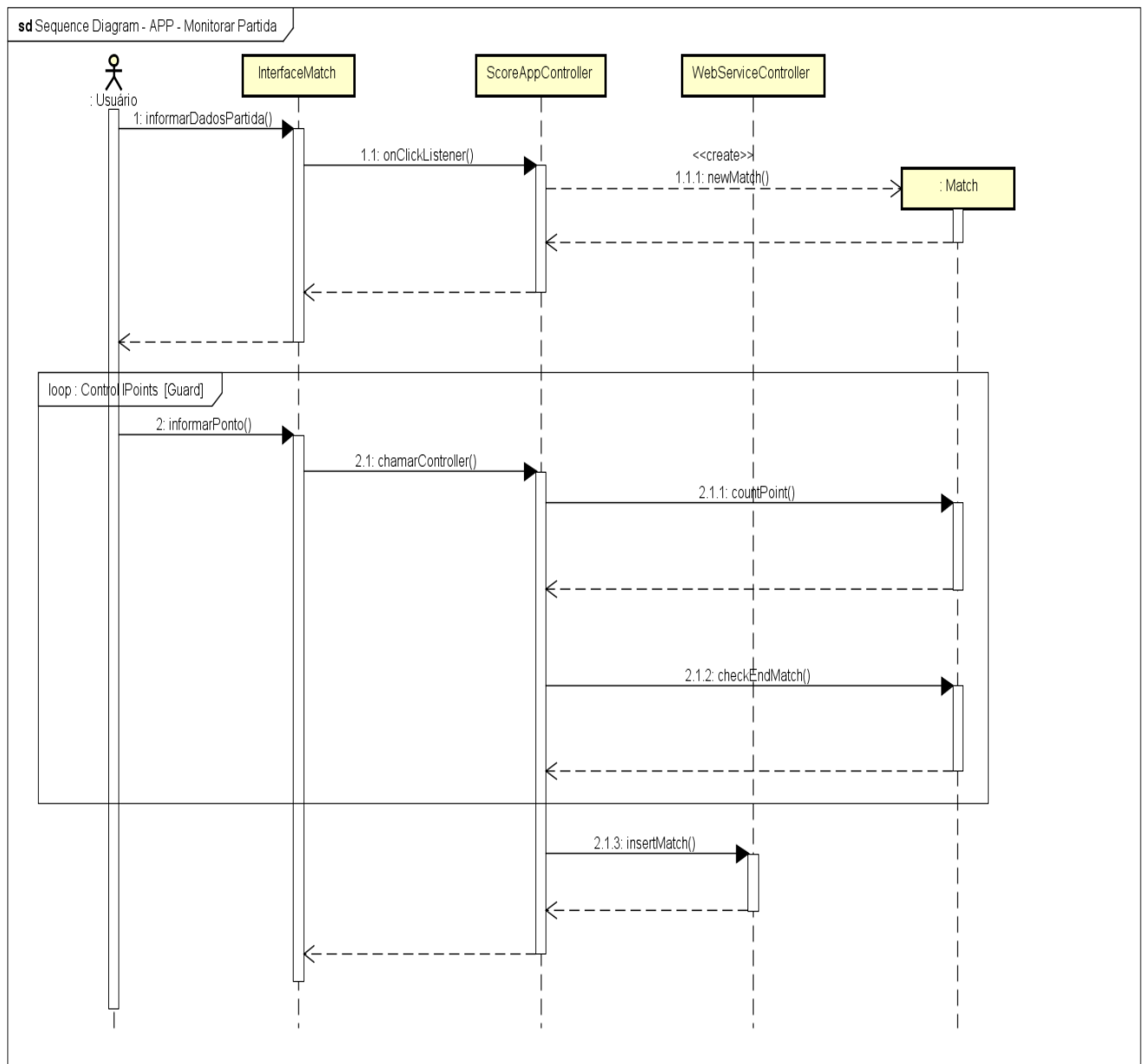
APP – Realizar Cadastro



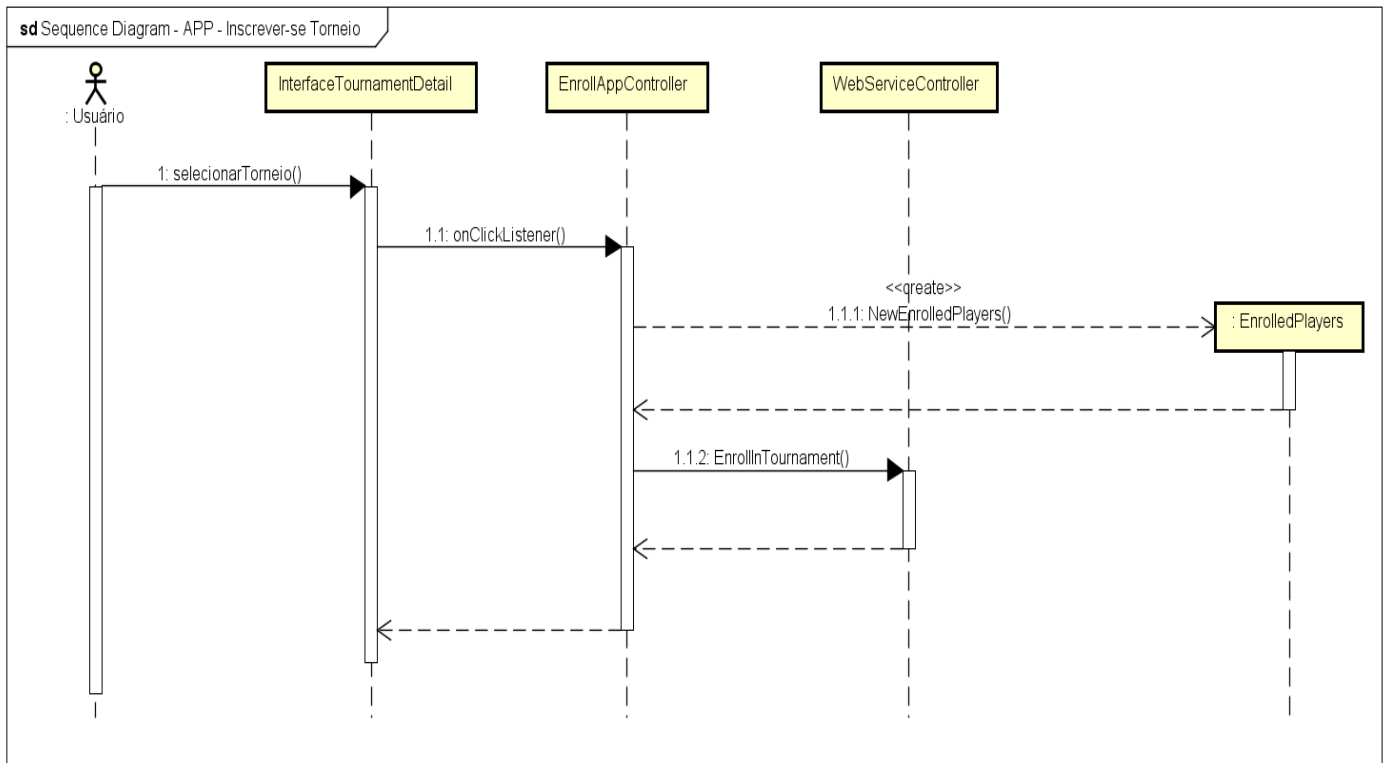
APP – Arbitrar Partida



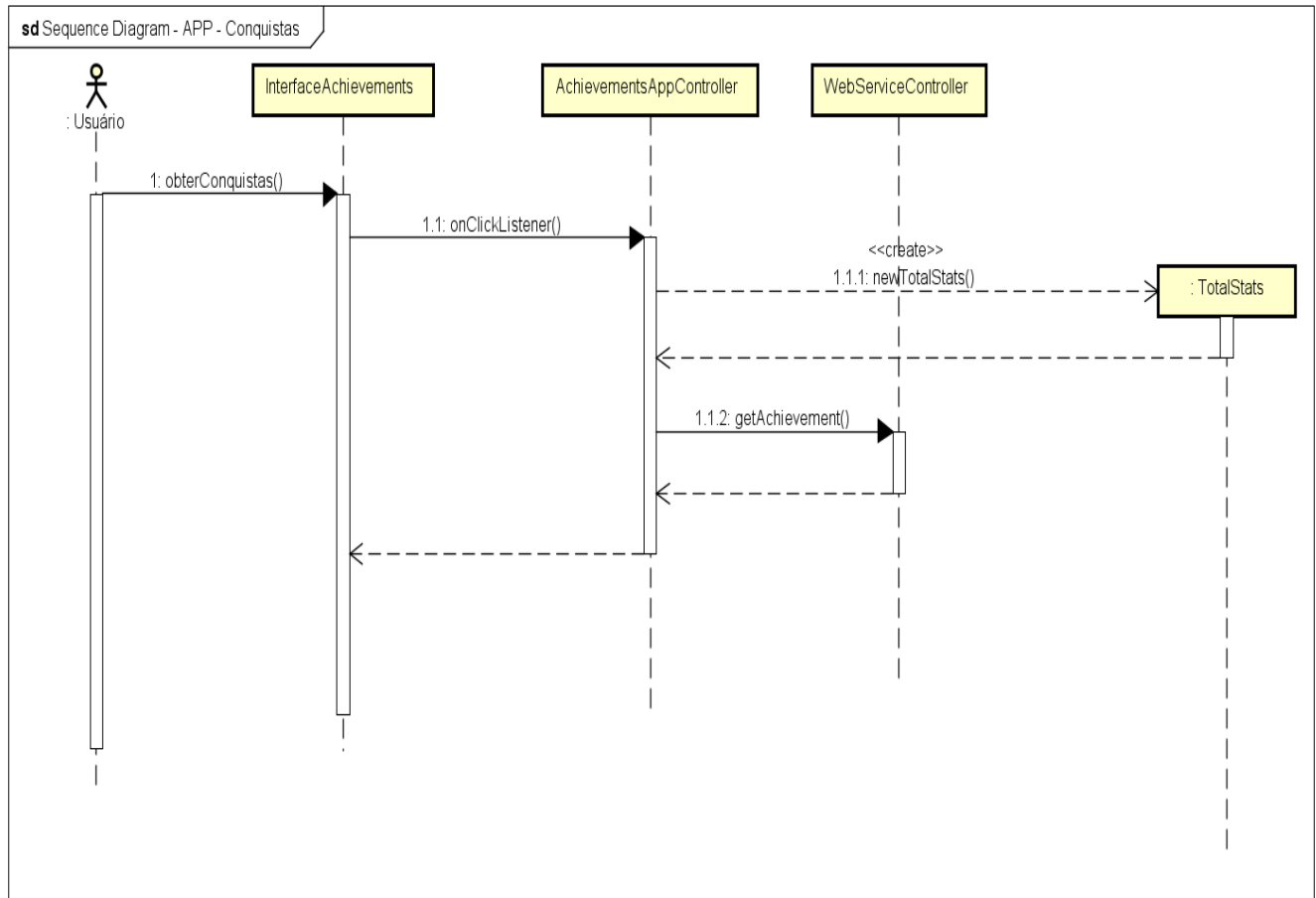
APP – Monitorar Partida



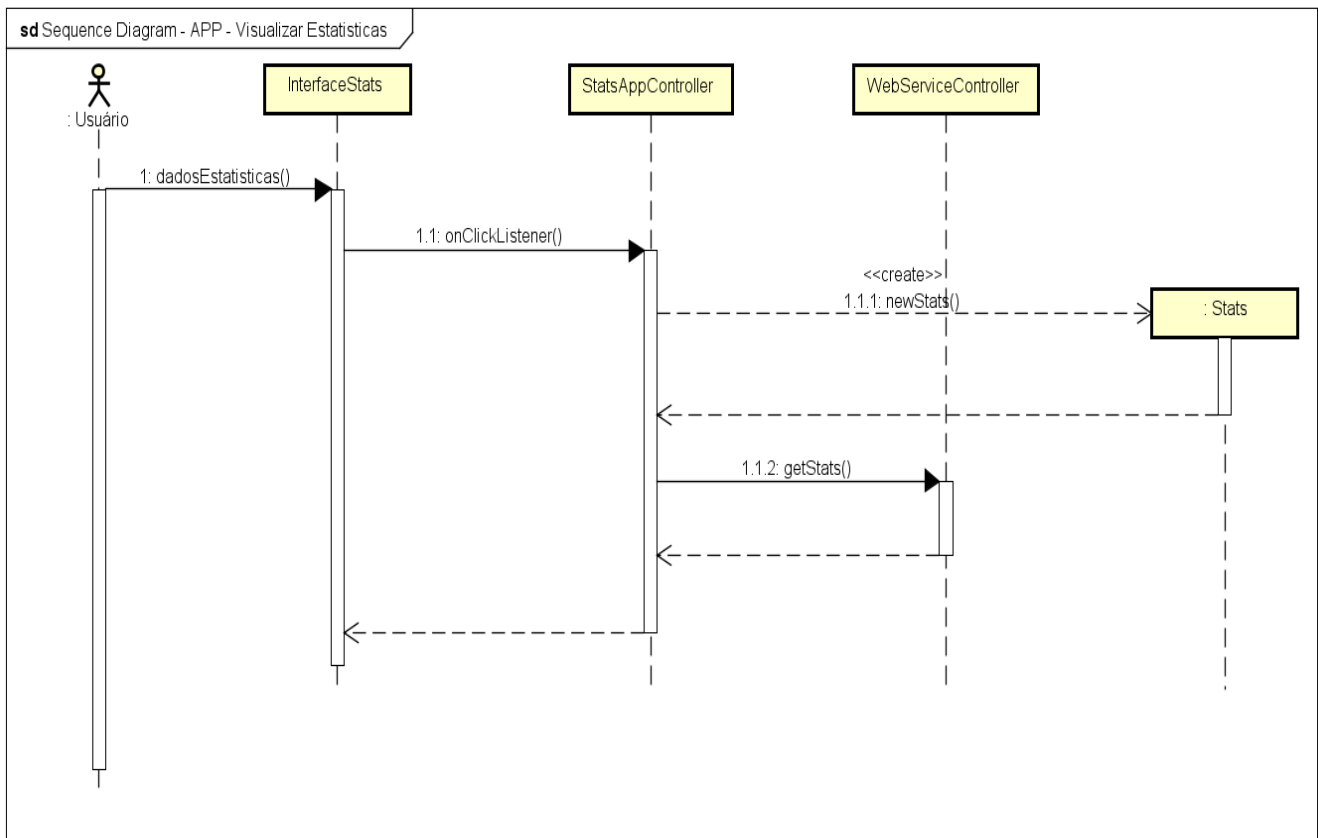
APP – Inscrever-se em Torneio



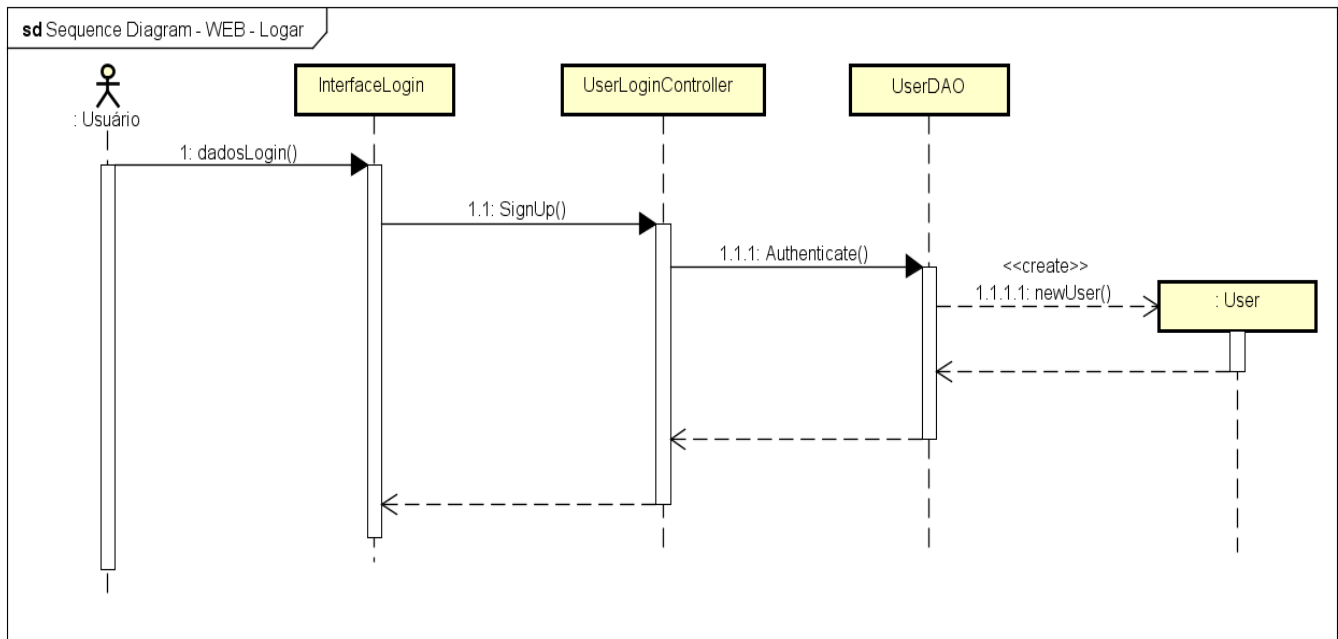
APP – Visualizar Conquistas



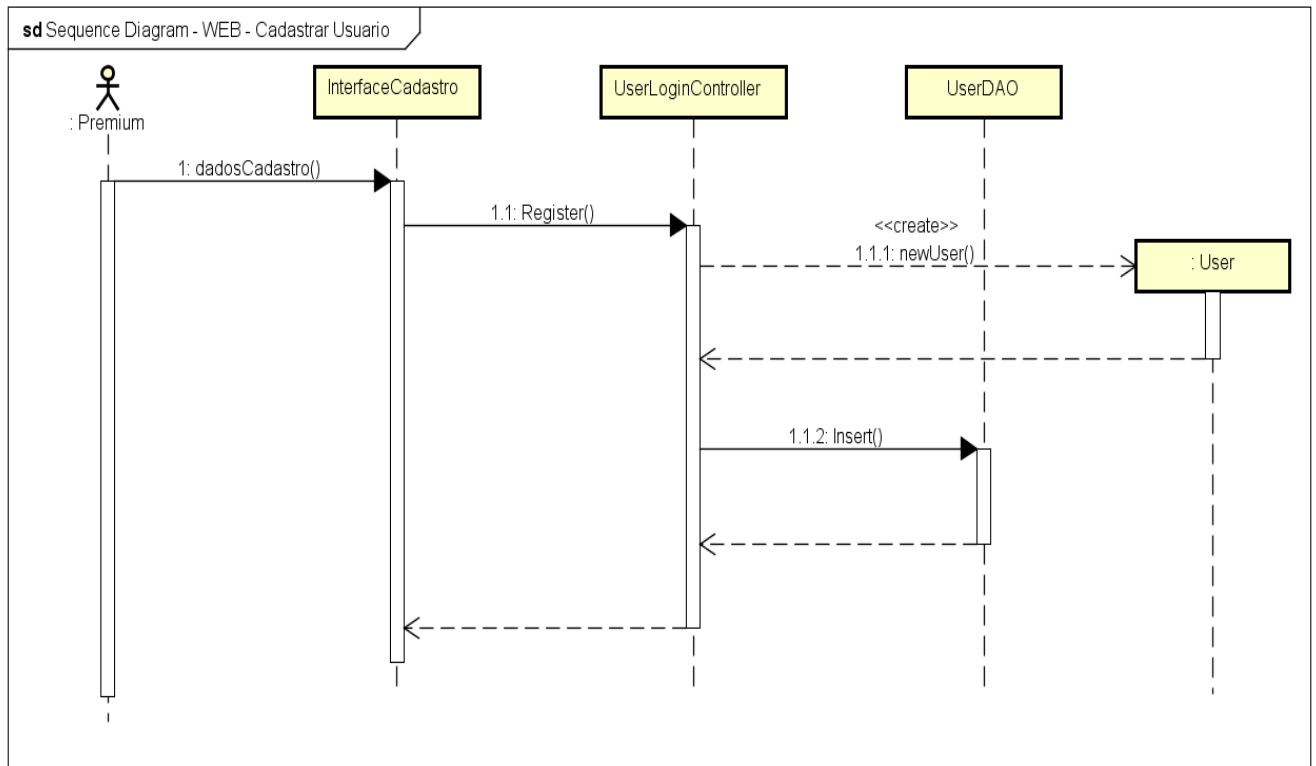
APP – Visualizar Estadísticas



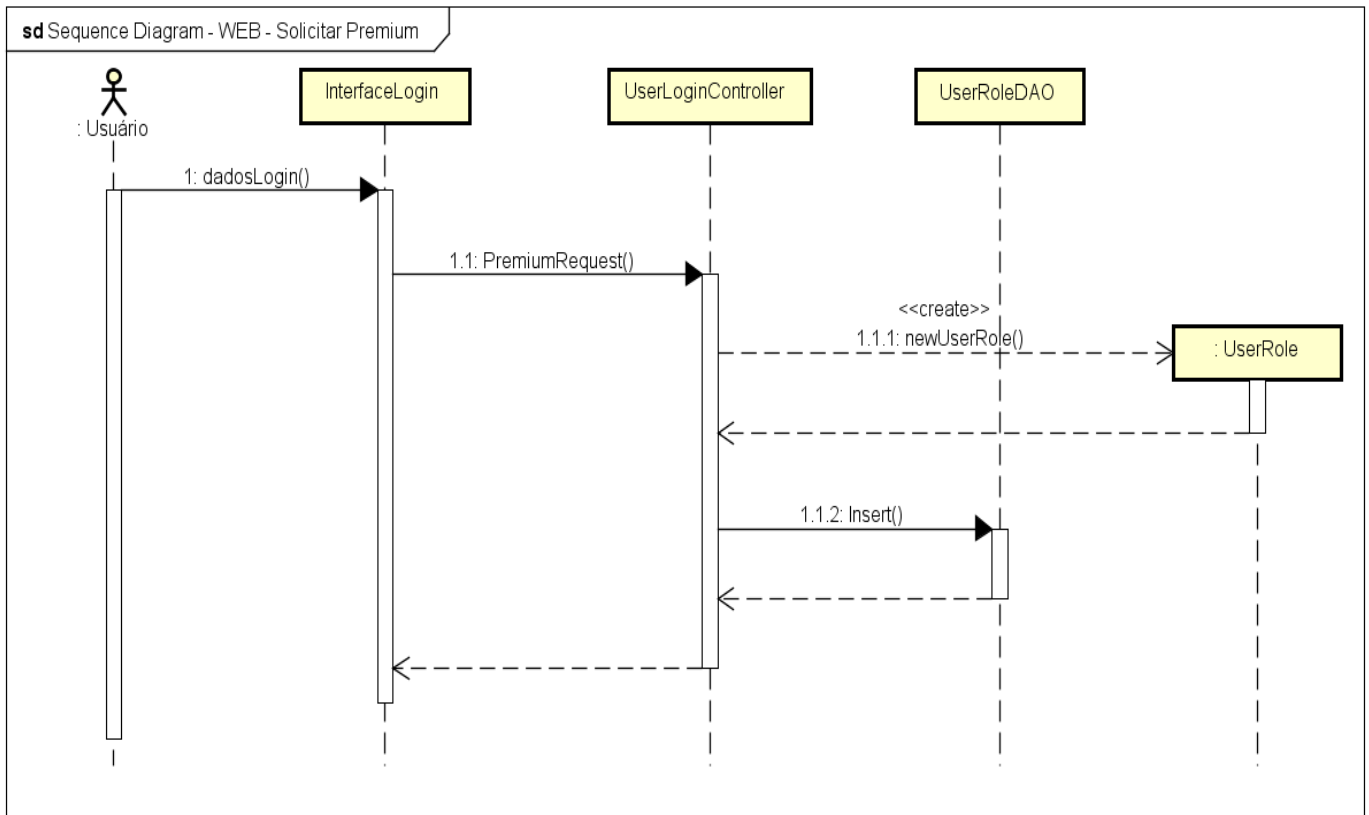
WEB – Efetuar Login



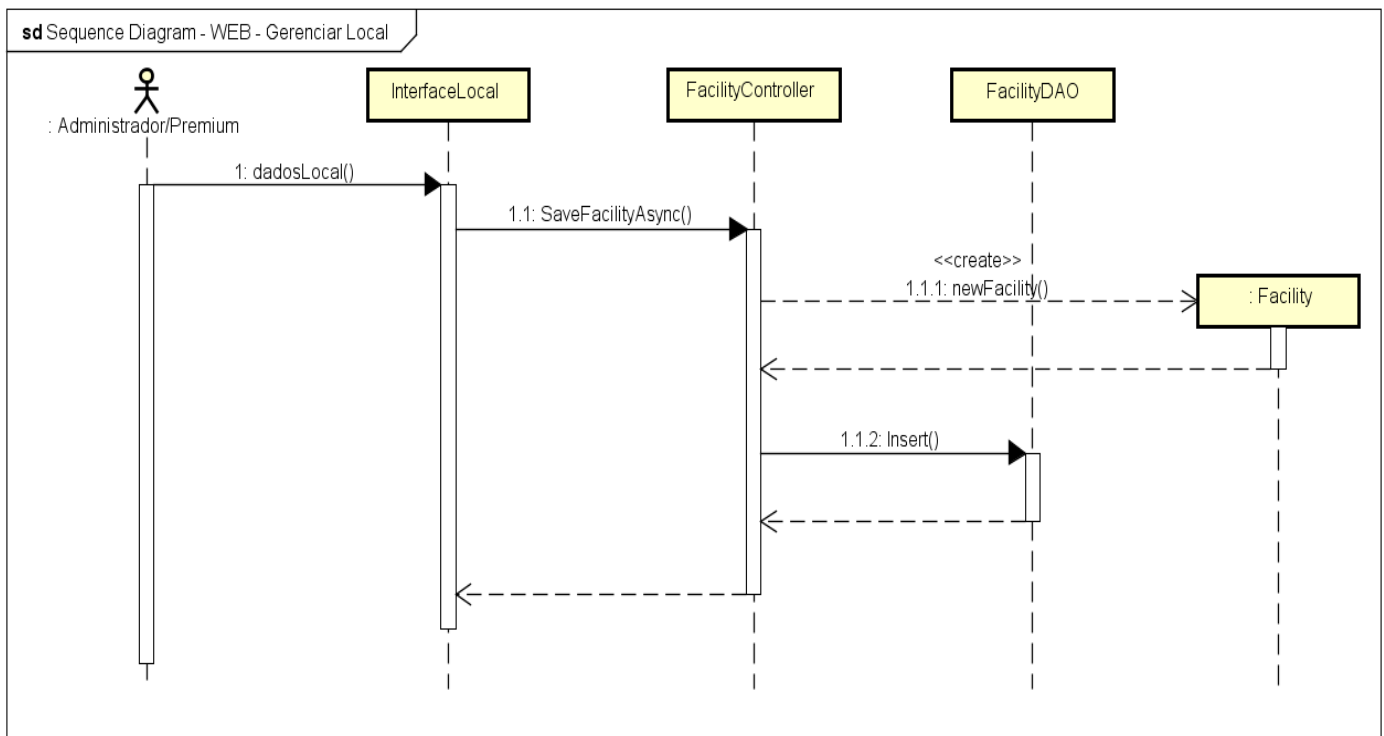
WEB – Realizar Cadastro



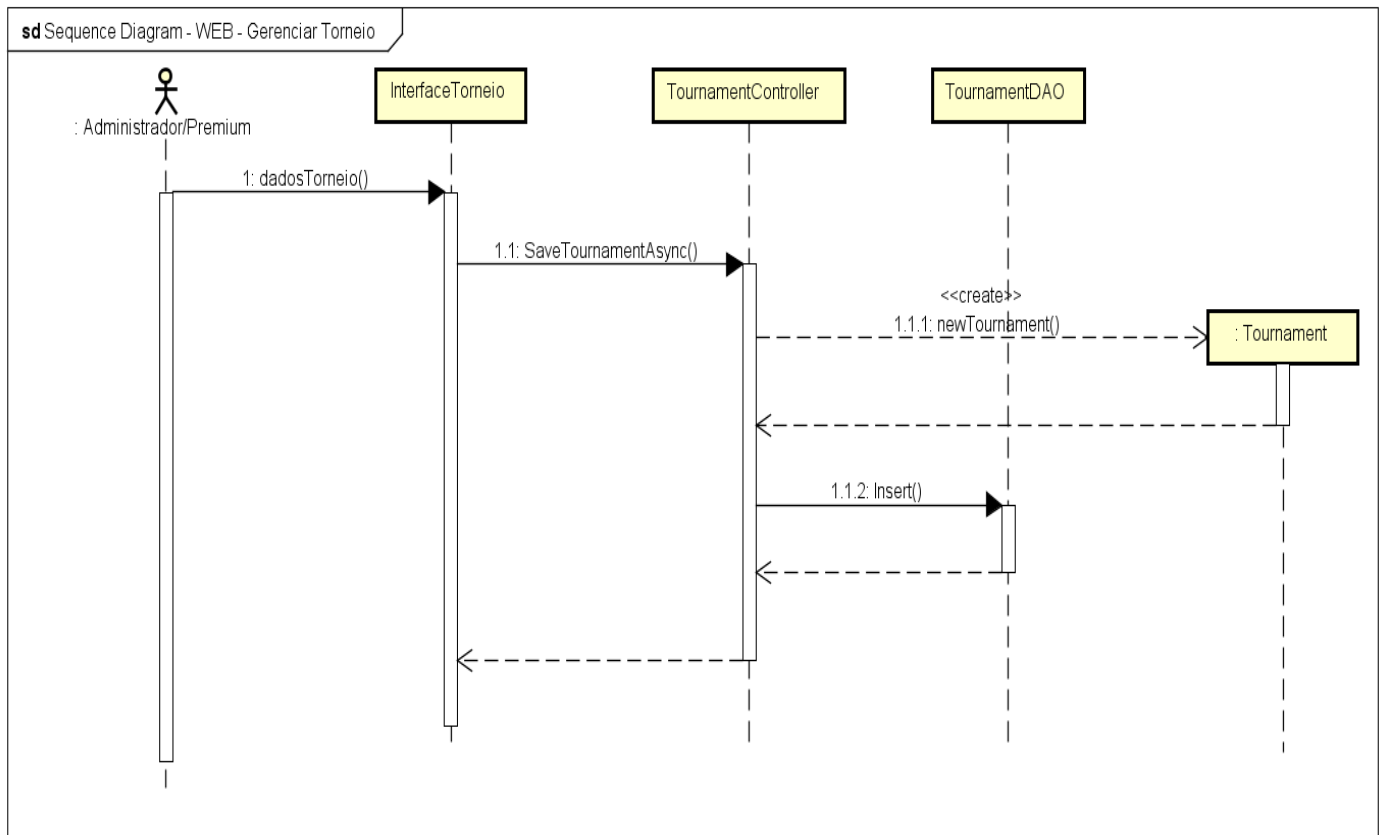
WEB – Solicitar Conta Premium



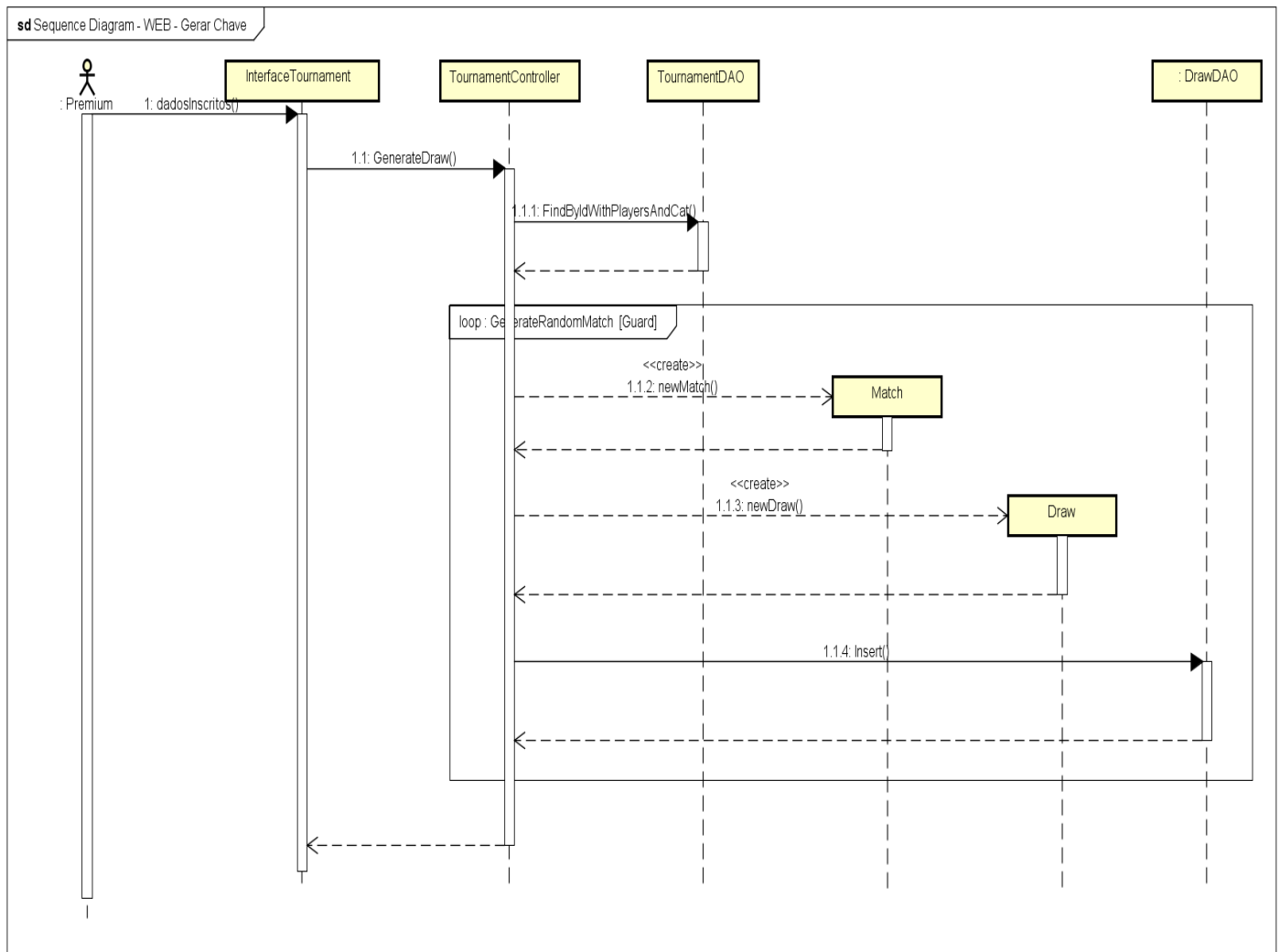
WEB – Gerenciar Local



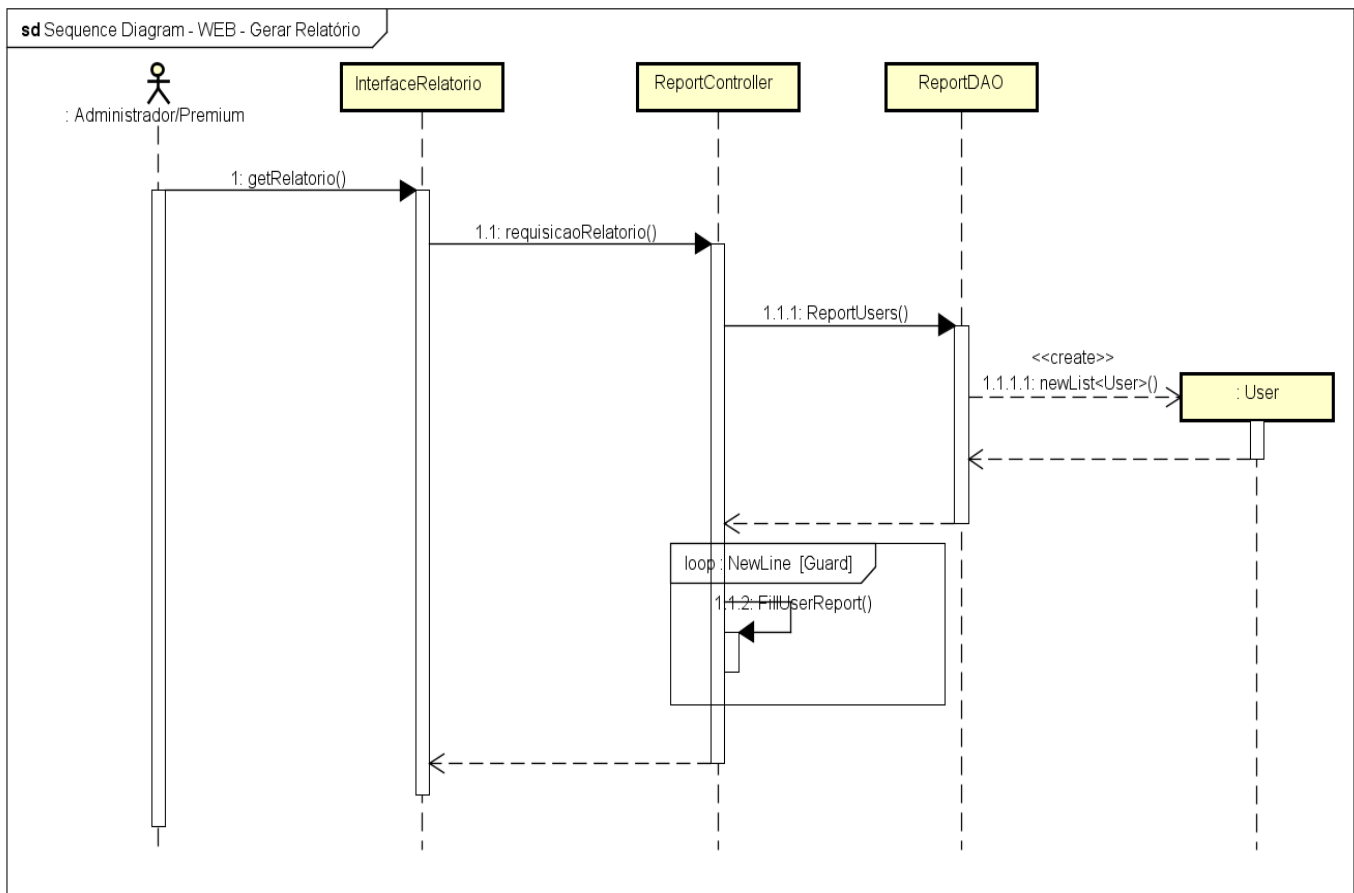
WEB – Gerenciar Torneio



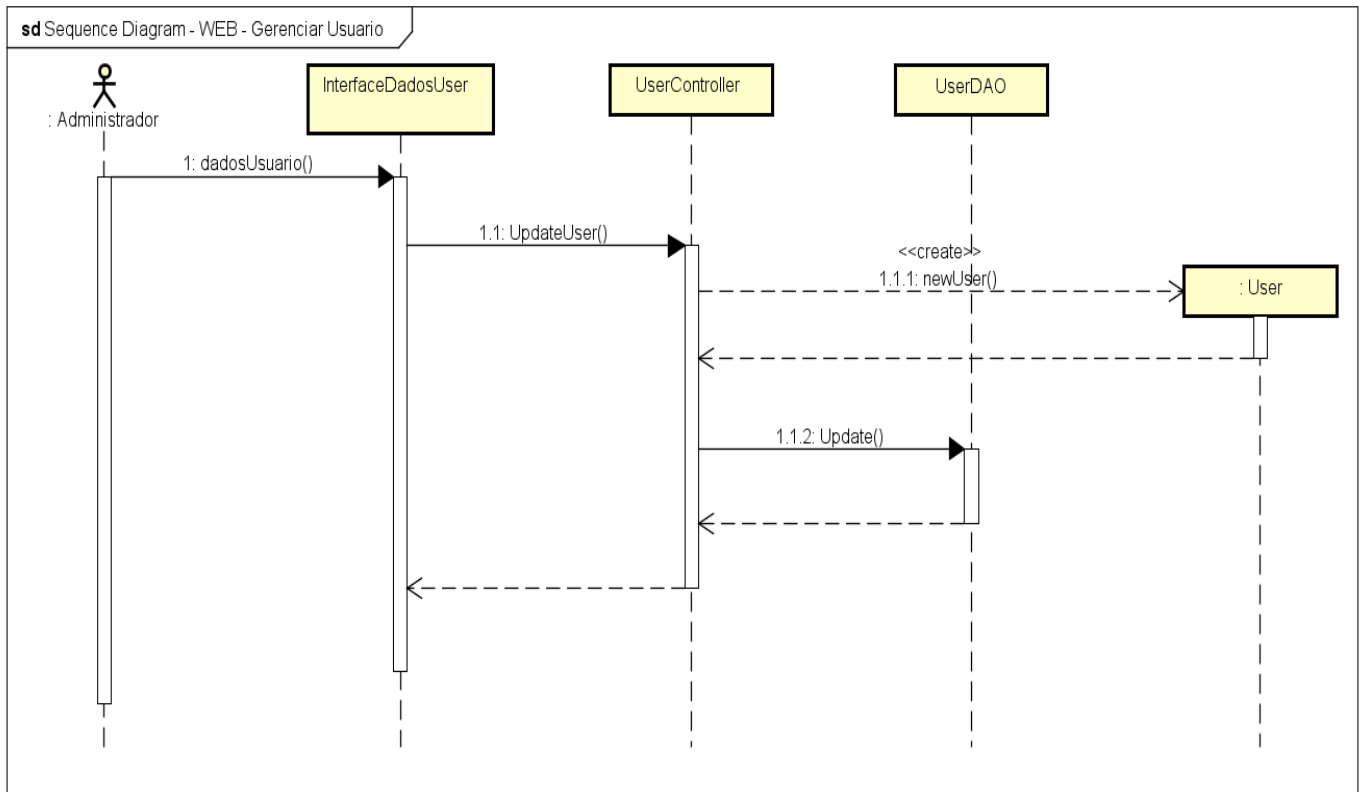
WEB – Gerar Chave



WEB – Gerar Relatório



WEB – Gerenciar Usuário



WEB – Gerenciar Requisição

