

Play Football: Protótipo de um jogo de futebol americano

Wesley Almeida da Silva, André Constantino da Silva

Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP) –
Campus Hortolândia - São Paulo – SP – Brasil

wsyalmeida@gmail.com, andre.constantino@ifsp.edu.br

Abstract. *The American football is a very popular sport in the United States of America, and recently its popularity has been increasing in Brazil. This prototype aims to bring Brazilians closer to this sport. This game works using probability considering the strategy chosen by user, and also considering the character-players skills. In this project we created the game design document (GDD), entity relationship model, class diagram, and developed using the Agile/Scrum methodology.*

Resumo. *O futebol americano é um esporte muito popular nos Estados Unidos, e nos últimos anos ele vem crescendo de forma significativa no Brasil, sendo assim este protótipo visa a aproximação dos brasileiros ao esporte. Sua mecânica faz o uso da probabilidade, baseando-se nas estratégias selecionadas pelo usuário e nas habilidades de seus personagens-jogadores. Para desenvolver este projeto foram criados o game design document (GDD), modelo entidade/relacionamento, diagrama de classe, e desenvolvido utilizando a metodologia ágil Scrum.*

1. Introdução

Este projeto aborda o *Play Football*, um protótipo de um jogo no contexto do esporte futebol americano. O futebol americano é um esporte coletivo composto por 2 times, onde cada time tem 3 equipes para fins diferentes, sendo o principal objetivo do jogo o ganho de território, que pode ocorrer por meio de jogadas terrestres ou áreas (SANTOS et al.,2016).

O desenvolvimento deste protótipo foi motivado pelo número de brasileiros que acompanham o futebol americano. Conforme apuração do IBOPE RECUPOM (2017), o número de simpatizantes pela principal liga americana de futebol americano a *NFL*, era de mais de 3 milhões em 2015, e cresceu para mais de 15 milhões em 2017.

Conforme dados levantados pela empresa IBOPE RECUPOM, foi possível elaborar o gráfico abaixo (Figura 1):

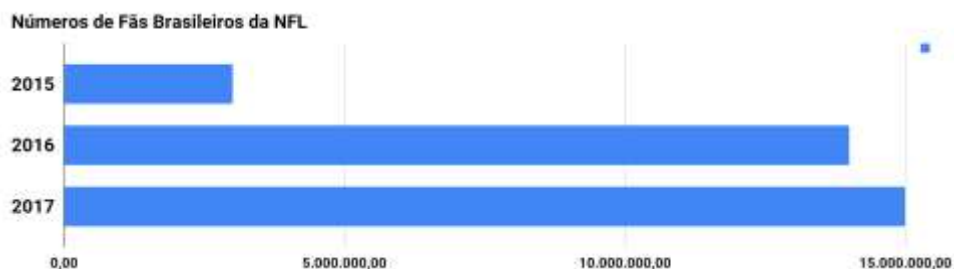


Figura 1. Valores aproximados referente ao aumento no número de fãs brasileiros pela *NFL*. Fonte: IBOPE RECUPOM (2015,2016,2017)

No Brasil, existem federações estaduais que gerenciam o campeonato dentro do

estado, exemplo disso ocorre no Rio Grande do Sul. Mediante um estudo apuraram que, durante a fase de grupos do campeonato gaúcho de futebol americano de 2018, foram vendidos 5.351 ingressos em 21 jogos, com isto foi arrecadado cerca de R\$ 46.745,00 (FEDERAÇÃO GAÚCHA DE FUTEBOL AMERICANO, 2018). Com relação a fase eliminatória excluindo a final, o campeonato obteve uma renda de R\$ 28.950,00 ou R\$ 4.825,00 por jogo (FEDERAÇÃO GAÚCHA DE FUTEBOL AMERICANO, 2018).

Após o término do campeonato, a Federação Gaúcha de Futebol Americano concluiu os seguintes dados (Figura 2):



Figura 2. Dados obtidos no final do campeonato. Fonte: (FEDERAÇÃO GAÚCHA DE FUTEBOL AMERICANO,2018)

Em outubro de 2017 foi publicado um artigo no site O Globo, escrito pelo técnico da seleção brasileira de futebol americano, na qual relata que os jogadores de futebol americano realizam diversos tipos de sacrifícios, como arcar com equipamentos e viagens do seu próprio bolso. No entanto este cenário vem mudando atualmente; o maior exemplo disso foi o amistoso entre Brasil e Argentina que ocorreu em um estádio de copa do mundo sem que os jogadores tivessem que se preocupar com qualquer tipo de despesa (MENDES, 2017).

Outro fator motivacional para o desenvolvimento do projeto é o número de jogadores de jogos eletrônicos no Brasil. O Brasil conta com 75,7 milhões de jogadores de jogos eletrônicos, e espera-se que gastem 1,5 bilhões de dólares em 2018; com estes números o Brasil ocupa o 13º maior mercado de games no mundo (NEWZOO, 2018).

2. Jogos Correlatos

Durante a pesquisa por jogos correlatos, encontramos 3 jogos com características que se aproximam dos objetivos finais deste projeto, são eles:

Brasfoot

Jogando o brasfoot você assume o papel de um gerente de futebol, tendo como funções: definição das táticas do seu time, compra e venda de jogadores, além de ser técnico de alguma seleção em tempo de copa do mundo (BRASFOOT, 2018). Este jogo se assemelha bastante aos objetivos pretendidos, no entanto neste jogo o usuário gerencia outra modalidade de esporte.

Madden NFL

É o principal jogo do mercado, ele utiliza todos os jogadores e times oficiais da NFL, e são atualizados semana a semana (TECHTUDO, 2017). Porém o jogo é pago, e com uma mecânica diferente da pretendida, além de contar com um alto investimento para seu desenvolvimento.

NFL Showdown

Dentro do jogo você será o técnico, terá atribuições como treinar seus jogadores, organizar e escolher as melhores estratégias. Este jogo está disponível em inglês e funciona no Windows Phone. Apesar de gratuito, limita demais os usuários que não estejam dispostos a pagar por recursos. Outros pontos negativos são que o jogo é inglês e disponível apenas para o Windows Phone (BAIXAKI, 2015).

Conforme abordado acima, podemos destacar algumas características nestes jogos que os distanciam do nosso projeto: são pagos e em inglês. Sendo assim, o objetivo foi desenvolver um protótipo de um jogo que simule a gerência de um time de futebol americano no Brasil, dentro e fora do campo, com o intuito de, futuramente, se tornar um jogo completo, gratuito, de código aberto, em português e disponível ao público brasileiro, gerando assim, mais adeptos deste esporte no Brasil.

3. Fundamentação Teórica

Tendo em vista ser um protótipo de um jogo, partimos do princípio que um jogo “é um sistema no qual jogadores estão engajados em um conflito artificial definido por regras que gera um resultado quantificável” (SALEN & ZIMMERMANN, 2004, p. 80).

A princípio foi pensado no desenvolvimento de um protótipo, tendo em vista a complexidade e demanda de tempo, dinheiro e pessoal para o desenvolvimento completo de um jogo completo. Objetivo do desenvolvimento do protótipo é testar o projeto e as suas funcionalidades (WEILLER, 2015).

A maioria das desenvolvedoras de jogos digitais, depositam o sucesso do jogo em 3 setores: a programação, a arte e o design. Às vezes estas áreas podem ser representadas apenas por uma pessoa, quando isto ocorre ela deve saber o momento adequado de executar cada tarefa (WEILLER, 2015).

Programação

O motor das desenvolvedoras dos jogos é o departamento de programação, quando o jogo tem um bom código ele passa despercebido ao público, enquanto em codificações ruins se destacam por *bugs* (WEILLER, 2015).

Arte

Dentro das desenvolvedoras, este departamento é representado pelos grupos de áudio e gráfico. O grupo de áudio fica responsável por trilhas sonoras, efeitos sonoros, *voice action* entre outros. Enquanto o grupo da arte trabalha com as artes 2D e 3D, pintura digital, pré-renderização entre outros (WEILLER, 2015).

Design

Neste departamento são definidos os conceitos e as mecânicas do jogo, após testadas e implementadas (WEILLER, 2015).

O gênero deste protótipo é de esporte e estratégia baseado em turnos. Conforme WEILLER(2015), o gênero de um jogo é composto por semelhanças comuns entre eles, como regras, dinâmica de acontecimentos do jogo, mecânica dos inimigos, visão e posicionamento da câmera entre outros.

O gênero de estratégia se caracteriza pela visualização geral do território e das suas unidades, organizando para que possam evoluir e aumentar, com o objetivo de ganhar do seu rival. Este gênero é dividido entre tempo real ou baseado em turnos

(WEILLER, 2015).

Salientamos que, os jogos digitais têm diversas vantagens como estimular os usuários a desenvolverem estratégias para resolução de problemas e exercitar o uso do raciocínio lógico e das tomadas de decisões rápidas, e depois de suas ações ter avaliação dos resultados (ABRAGAMES, 2004).

Idealizamos o jogo a partir do documento *game design document*. O *game design document*, descreve o que pensamos para o jogo, como regras, controles, interações e personagens. Este é um documento vivo, podendo sofrer mudanças constantes, devido a diversidade de profissionais em uma equipe de produção de jogos (OLIVEIRA, 2014).

Utilizamos a metodologia ágil Scrum para o desenvolvimento do projeto. O Scrum é uma metodologia ágil, que permite a divisão de grandes problemas em pequenas partes, utilizando um pequeno grupo de pessoas, sendo gerenciado pelo Scrum Master (PRESSMAN, 2011).

O Scrum define os seguintes papéis:

Product Owner

O *product owner* é o responsável pelas tomadas de decisões, gerenciamento e priorização do *product backlog* (DUARTE, 2016).

Desenvolvedores

É a equipe responsável pelo desenvolvimento do produto. Os membros da equipe têm como características serem: auto gerenciáveis e multifuncionais. A equipe de desenvolvedores não é composta apenas por programadores. Uma de suas funções é o estabelecimento do quanto conseguem entregar em cada iteração (DUARTE, 2016).

Scrum Master

Tem como responsabilidade aplicar e garantir a prática do Scrum dentro do projeto, assegurando ainda o alcance dos objetivos pretendidos pelo *product owner* (DUARTE, 2016).

A metodologia Scrum define os seguintes termos, documentos e atividades:

Sprint

As *sprints* são períodos curtos para a entrega de algo utilizável no produto. Além de desenvolvimento, as *sprints* são compostas por atividades, planejamentos, reuniões, revisões e retrospectivas (DUARTE, 2016).

Sprint Planning

São reuniões realizadas antes de cada *sprint*, onde o *product owner* fala sobre as prioridades e objetivos daquela *sprint*, e o *Scrum master* cuida para que os participantes não se distanciam do foco do *product owner* (DUARTE, 2016).

Artefatos

Artefatos são ferramentas que auxiliam na aplicação da metodologia Scrum (DUARTE, 2016).

Product Backlog

O *product backlog* é um documento gerenciado pelo *product owner*, onde contém a lista de funcionalidades a serem desenvolvidas ordenadas por prioridade (DUARTE, 2016).

Sprint Backlog

É uma pequena lista, retirada do *product backlog*, com o intuito de ser desenvolvida na próxima *sprint*. Esta lista pode ser organizada com o auxílio do Kanban (DUARTE, 2016).

Kanban

O Kanban é originário da indústria utilizado para transporte e produção de mercadorias, no entanto ele foi adotado no gerenciamento de software para registrar o andamento do projeto. Na metodologia ágil Scrum ele entra como um artefato (DUARTE, 2016).

4. Metodologia

Durante a elaboração da metodologia foram desenvolvidos os seguintes documentos: *game design document*, diagrama entidade relacionamento, diagrama de classes, kanban, e foi aplicado a metodologia ágil SCRUM. A Figura 3 mostra, de uma forma geral, a estrutura da metodologia aplicada do projeto.

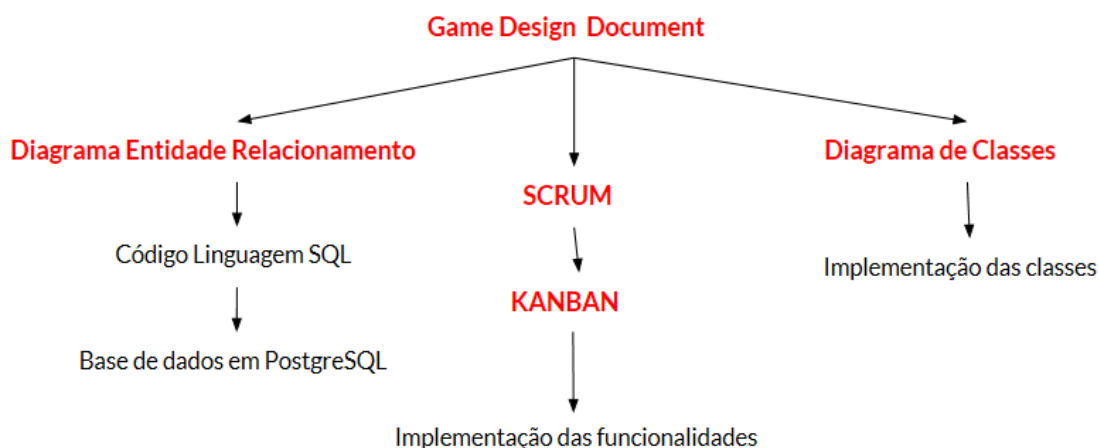


Figura 3. Estrutura do desenvolvimento da metodologia

Iniciamos o desenvolvimento deste projeto documentando a ideia do protótipo por meio do *game design document*. Dentro deste documento descrevemos o seu funcionamento, suas regras e suas interações. A Figura 4 apresenta as seções elaboradas do *game design document*.



Figura 4. Elementos contidos no *game design document*

Este documento foi preenchido com base em conhecimentos obtidos por experiência prática do autor, por conjunto de regras obtidas da *NFL* (a principal liga de futebol americano nos Estados Unidos), e do *IFAF* (regras aplicadas nos jogos universitários dos Estados Unidos e na principal liga brasileira de futebol americano a Brasil Futebol Americano-BFA).

Como dito na introdução, este protótipo simula a gerência de um time de futebol americano no Brasil. A ideia principal do *Play Football* se baseia na escolha da melhor formação, no momento adequado, selecionando os jogadores com as características que mais combinam na formação, no entanto o jogo ocorre por probabilidade, sendo assim ambos times tem a chance de vencer. Como ocorre na vida real, nem sempre o melhor time vence a partida. A Figura 5 apresenta o conceito da partida.

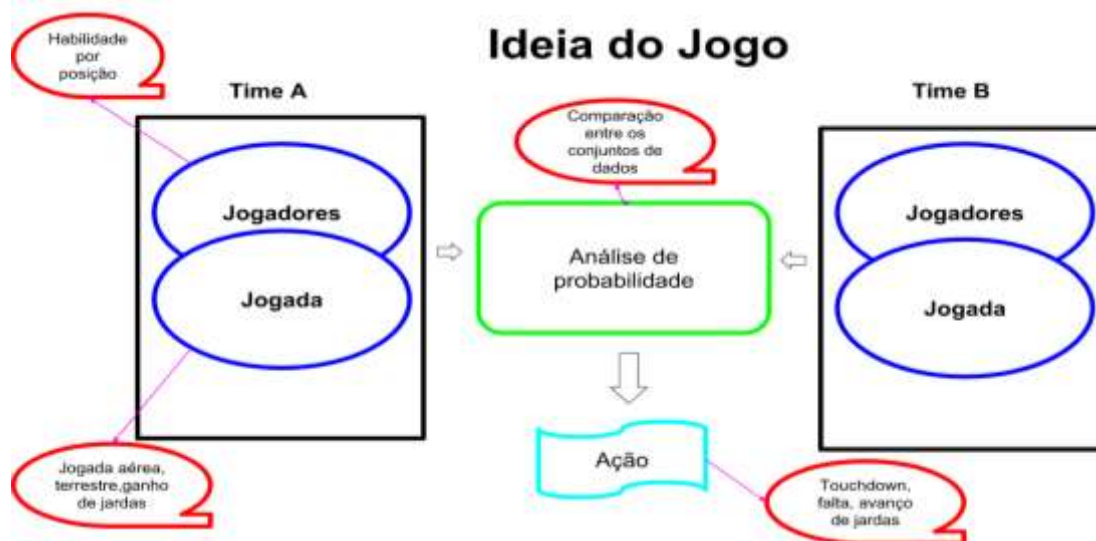


Figura 5. Conceito da partida

Logo após a construção do documento, extraiu-se informações como suas entidades, atributos e relacionamentos. Sendo assim, foi possível conciliar a necessidade de visualizar graficamente o esquema do banco de dados inicial, com os dados obtidos, gerando assim o diagrama entidade/relacionamento. “O modelo de dados de entidade/relacionamento (E-R) é baseado em uma percepção de um mundo real que consiste em uma coleção de objetos básicos, chamados entidades, e de relações entre

esses objetos."(SILBERSCHATZ, KORTH, SUDARSHAN, 2012, p. 10). A Figura 6 apresenta um recorte da entidade jogador no diagrama criado.



Figura 6. Recorte da entidade jogador no diagrama entidade/relacionamento

Com a conclusão do diagrama, elaboramos o código em linguagem *SQL* fazendo os ajustes necessários com objetivo de realizar um código simples. Para a execução das consultas *SQL* foi utilizado o pgAdmin III. A Figura 7 apresenta o código em *SQL* de criação da tabela jogador.

```
CREATE TABLE Jogador(
    Id SERIAL,
    nome VARCHAR(20),
    sobrenome VARCHAR(50),
    numero int,
    idade int,
    peso int,
    altura int,
    posicao VARCHAR(50),
    agilidade int,
    chute int,
    forca int,
    lancamento int,
    pegar int,
    marcacao int,
    resistencia int,
    salto int,
    tackle int,
    velocidade int,
    preco money,
    salario money,
    time int,
    campeonato int,

    FOREIGN KEY (campeonato) REFERENCES Campeonato (id),
    FOREIGN KEY (time) REFERENCES Time (id),
    PRIMARY KEY (id)
);
```

Figura 7. Código para a tabela de dados sobre jogadores

Antes de iniciar a programação, foi necessário pensar no paradigma a ser utilizado. Entre os tipos de paradigmas de programação aprendidas no decorrer do curso, foi escolhido o de orientação a objetos. A programação orientada a objetos junto com seus princípios e boas práticas visam uma arquitetura flexível, com um código legível, fácil de compreender e reutilizar, facilitando assim futuras alterações no código (SILVEIRA, 2012).

Com intuito de estruturar o código, foi utilizada a *UML*. “A *UML* é uma linguagem que define elementos gráficos (visuais) que podem ser utilizados na modelagem de sistemas. Esses elementos permitem representar os conceitos do paradigma da orientação a objetos” (BEZERRA, 2015, p. 15).

Para ter uma pré-visualização do que o sistema faria, foi desenvolvido o diagrama de classes. “O modelo de classe de análise são aquelas que se tornam

evidentes na medida em que focamos a atenção sobre “o que” o sistema deve fazer” (BEZERRA, 2015, p. 15). A Figura 8 apresenta o diagrama de classes do protótipo.

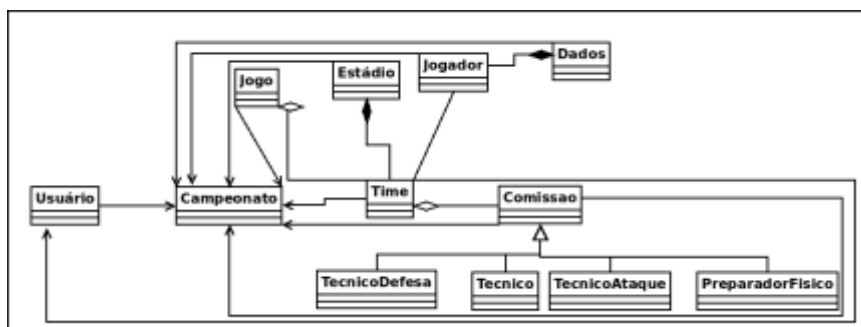


Figura 8. Diagrama de classes

Feito isto, foi necessário pensar, analisar e construir todas as funcionalidades, diante deste problema, decidimos utilizar uma metodologia ágil que suprisse a necessidade de dividir um grande projeto em pequenas partes, então foi usado o Scrum.

Antes do início de cada *sprint*, foi realizado o *sprint planning* onde foi apresentado ao orientador o resultado anterior, e definidos os passos seguintes com base no *product backlog*, documento montado no início do projeto. Neste projeto foi utilizado o artefato kanban para acompanhar o decorrer do desenvolvimento das *sprints backlogs*.

Sendo assim, foi desenvolvido o *product backlog* para armazenar todas as funcionalidades, definindo prioridade, dificuldade e tempo. Ao total, foram identificadas 35 funcionalidades. A Figura 9 apresenta um recorte do *product backlog*.

Tela de dashboard	Tela contendo todas opções de visualizações	1	Fácil	Média
Tela Login	Tela de login	1	Fácil	Média
Tela Cadastro	Tela de cadastro	1	Fácil	Média

Figura 9. Recorte de algumas funcionalidades do *product backlog*

Logo após, foi pensado em uma maneira de dividir as funcionalidades levando em consideração a entrega de um código funcional na primeira *release*. Para isto, tivemos que analisar as dependências de cada funcionalidade sobre as outras, e priorizá-las. Feito isto a entrega das releases seguintes ocorreriam de forma incremental. A Figura 10 apresenta a divisão do *product backlog* por *releases*.

Release		
Release 1	Release 2	Release 3
Modelo Banco de Dados	Criação de tecnico	Criação do juiz
Criação do banco	Criação do preparador fisico	Criação de lojas
Diagrama de Classe	Criação do tecnico de ataque	Tela de compra e venda jogadores
Cadastro de Usuario	Criação do tecnico de defesa	Edição de usuário
Criação do login	Criação de patrocinios	Evolução de jogadores
Criação de carregamento	Criação de estádio	
Criação de Campeonato	Criação de academia	
Criação de times	Tela estádio	
Criação de Jogadores	Tela Jogos	
Criação de estratégias	Tela Campeonato Grupo	
Tela de dashboard	Tela Campeonato eliminatória	
Tela Login	Tela de pontuações	
Tela Cadastro	Tela de patrocinio	
Tela Carregamento	Análise financeira	
Tela do jogo		
Análise de jogadas		

Figura 10. Release com as funcionalidades destacadas

Com as *releases* identificadas, a primeira foi distribuída no *sprint backlog*, levando em conta o tempo e dificuldade para o desenvolvimento. Definimos que cada *sprint* deveria durar 1 semana. A Figura 11 apresenta um recorte sobre o *sprint 5* do *sprint backlog*.

Sprint 5	Duração : 1 Semana
Tela de dashboard	
Tela Login	
Tela Cadastro	
Tela Carregamento	

Figura 11. Recorte da sprint 5 no sprint backlog

Para acompanhar a prazo e localização do trabalho atual, foi utilizado o Kanban. O Kanban define o uso de um painel sinalizando em 3 colunas; as colunas representam o andamento da tarefa, podendo a tarefa estar sinalizada para fazer, em progresso ou feito (DUARTE, 2016). A Figura 12 apresenta um recorte do documento Kanban.

Release 1 - Sprint 5	Atividades			Status	
	to do	in progress	done	Início	Fim
Desenvolver front end das telas			Tela de dashboard	28/07/2018	29/07/2018
			Tela Login	28/07/2018	29/07/2018
			Tela Cadastro	28/07/2018	29/07/2018
			Tela Carregamento	28/07/2018	29/07/2018

Figura 12. Recorte do Kanban

Para o desenvolvimento e estruturação do código foi utilizada a ferramenta Eclipse, servindo como ambiente de programação do *back end* e *front end*.

Desenvolvemos o *back end* com o *hypertext preprocessor (PHP)* uma linguagem de programação voltada para a internet, possibilitando o desenvolvimento de sites dinâmicos fácil de usar que conta com uma grande quantidade de recursos, além de ser gratuita e com código aberto, com sua documentação disponível no site oficial, podendo ser programado orientado a objetos, e suportando diversos bancos de dados e com portabilidade para o *Windows, GNU/Linux* e o *Unix* (NIEDERAUER,2011).

Utilizamos o HTML5 e o JQuery para desenvolvimento do *front end*. Como sistema de gerenciamento de banco de dados usamos o PostgreSQL, e para testarmos a protótipo usamos o navegador Firefox.

Padrões de projeto apresentam soluções para um problema em determinado contexto, são um conjunto de situações com problemas, sendo essas soluções aplicáveis em diversas situações (MCLAUGHLIN, POLLICE, 2007).

Diante de todos documentos produzidos e pesquisas elaboradas, foi possível medir o nível de complexidade e tamanho do código. Com o intuito de melhorar o mesmo foi utilizado padrão arquitetural modelo-visualização-controlador (*MVC*).

O modelo-visualização-controlador é um padrão arquitetural que pode ser composto por outros padrões de projeto, ele é uma arquitetura utilizada para delimitar seu código em segmentos funcionais, em que cada parte do seu código seja responsável por determinadas tarefas, obtendo assim a sua reusabilidade (MCLAUGHLIN, POLLICE, 2007).

Optou-se por dividir o código em pacotes com os nomes e funções que atendem aos critérios modelo, visão e controle (*MVC*), conforme a responsabilidade do arquivo no código. Foi criada uma pasta para armazenar os *assets* do jogo (como música e fotos). A Figura 13 apresenta uma visualização da organização do código.

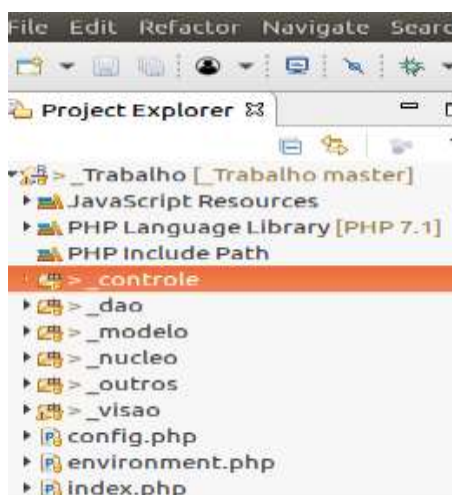


Figura 13. Visualização do código com a implementação do padrão de projeto modelo-visualização-controlador

Ainda foi utilizado o *Data Access Object (DAO)* para isolar a comunicação com o banco de dados. A Figura 14 apresenta uma visualização do *DAO*.



Figura 14. Demonstração do *Data Access Object* implementado no código

Tendo em vista a quantidade de *sprints* e alterações no decorrer do projeto, também foi usado o GIT como gerenciador de repositório, permitindo um controle seguro do código. É um ato comum entre as pessoas realizarem seus *backups* copiando os arquivos de um diretório para o outro, no entanto isto é uma prática perigosa, porque os diretórios podem ser apagados acidentalmente. Utilizando o git podemos salvar versões dos códigos tendo a opção de recuperar a versão determinada a qualquer momento (CHACON, STRAUB, 2018). A Figura 15 apresenta a execução do comando *git log* no terminal.

```
usuario@usuario-270E5G-270E5U: /var/www/html/_Trabalho$ git log
commit be890f7cf79a953bc659d76a600111d5c487c1c8
Author: Usuario <wsyalneida@gmail.com>
Date: Tue Jul 31 18:08:40 2018 -0300

    Sprint 5

commit 3c11f55afa6cf159a10b5ebaf72ff27c98fcce07
Author: Usuario <wsyalneida@gmail.com>
Date: Thu Jul 12 19:52:57 2018 -0300

    Sprint 3

commit 0c02fb49da2fb2435af4902b5fd2b6a1252702e5
Author: Usuario <wsyalneida@gmail.com>
Date: Fri Jul 6 14:37:34 2018 -0300

    Sprint 2
usuario@usuario-270E5G-270E5U: /var/www/html/_Trabalho$
```

Figura 15. Execução do comando *git log* no terminal

5. O Protótipo do *Play Football*

O protótipo¹ se inicia com a tela de login tocando uma música, dando as opções de cadastrar um novo usuário ou entrar no *Play Football*. A Figura 16 mostra a tela de *login*.

¹ Código do protótipo disponível em: <https://github.com/UserWesley/PlayFootball.git>



Figura 16. Tela de login

Ao efetuar o *login* no *Play Football*, o usuário será direcionado a tela de carregamento, onde será possível iniciar um novo campeonato, ou carregar um campeonato existente previamente criado por este usuário. A Figura 17 mostra a tela de carregamento.



Figura 17. Tela de carregamento, dando opção para usuário iniciar um novo campeonato ou carregar um existente

Se o usuário escolher a opção de novo campeonato, o protótipo irá gerar: 16 times com nomes aleatórios, 52 jogadores por time com atributos aleatórios e os 48

jogos da fase de grupos, permitindo o usuário selecionar seu time. A Figura 18 apresenta a estrutura de novo campeonato.



Figura 18. Estrutura de um novo campeonato

Após a inicialização de um novo campeonato, o usuário terá acesso aos dados de todos os seus jogadores, com visualização da tabela do campeonato, e de todos os jogos ocorridos ou que ocorrerão. O usuário tem como objetivo somar o maior número possível de pontos dentro do seu grupo, com o intuito de se classificar para a fase eliminatória.

Dentro do campeonato existem 4 grupos formados por 4 times cada, onde cada time joga duas vezes com os times de seu grupo, sendo um jogo em casa e o outro fora. Após o término da fase de grupos, a ideia é que os dois primeiros colocados de cada grupo irão avançar para a fase eliminatória, na qual as partidas serão geradas por meio de sorteio. A Figura 19 mostra a tela da partida.

Conforme exposto na Figura 19, o time que estiver destacado em azul pertence ao usuário, enquanto o outro ao adversário (computador). O botão vermelho disponibiliza a ação disponível ao usuário naquele momento, ao clicar nele será disponibilizado a seleção de jogada, jogadores e formação, com isto, os dados serão confrontados com o do adversário gerando assim um acontecimento, como touchdown, perda de bola entre outros. Cada acontecimento gasta um tempo, e quando o tempo acaba a partida finaliza. Ao clicar sobre o botão “Todos Acontecimentos” da cor azul o usuário terá a visualização de todos os eventos do jogo. O campo da cor preta contém dados referentes aquela partida. O time que mais pontuar sairá vencedor desta partida.

No decorrer da partida o usuário terá disponível as seguintes jogadas: atacar, defender, *kickoff* e retorno de *kickoff*. Quando o usuário realizar o *kickoff* ou retorno de *kickoff* são somadas todas as habilidades de 11 jogadores, em seguida será realizado um sorteio de número aleatório entre o número mínimo, que sempre será 0, e o número

máximo, que será a soma das habilidades dos dois times. Com o número gerado ocorre um resultado composto por tempo, avanço, acontecimento e próxima jogada do usuário. A Figura 20 mostra a análise para jogadas de *kickoff* e retorno *kickoff*.



Figura 19. Tela da partida entre duas equipes



Figura 20. Funcionamento da jogada de *kickoff* contra retorno de *kickoff*

Quando o usuário atacar ou defender, ele selecionará a jogada ataque (formação, estratégia, linha) ou defesa (formação, estratégia, marcação), em seguida será gerado um resultado que terá impacto direto no sorteio de um número aleatório, que gerará um resultado em relação ao tempo, avanço, acontecimento e definição da próxima jogada do usuário. A Figura 21 mostra a análise entre a jogada de ataque contra defesa.

6. Conclusão

Com o término deste projeto, o *Play Football* tornou-se funcional podendo servir como base para implementação das *releases* 2 e 3 e de outras novas funcionalidades, assim como para melhorias das funcionalidades existentes. Diante disto, sugere-se como trabalhos futuros a aplicação de testes de funcionalidades e da jogabilidade.

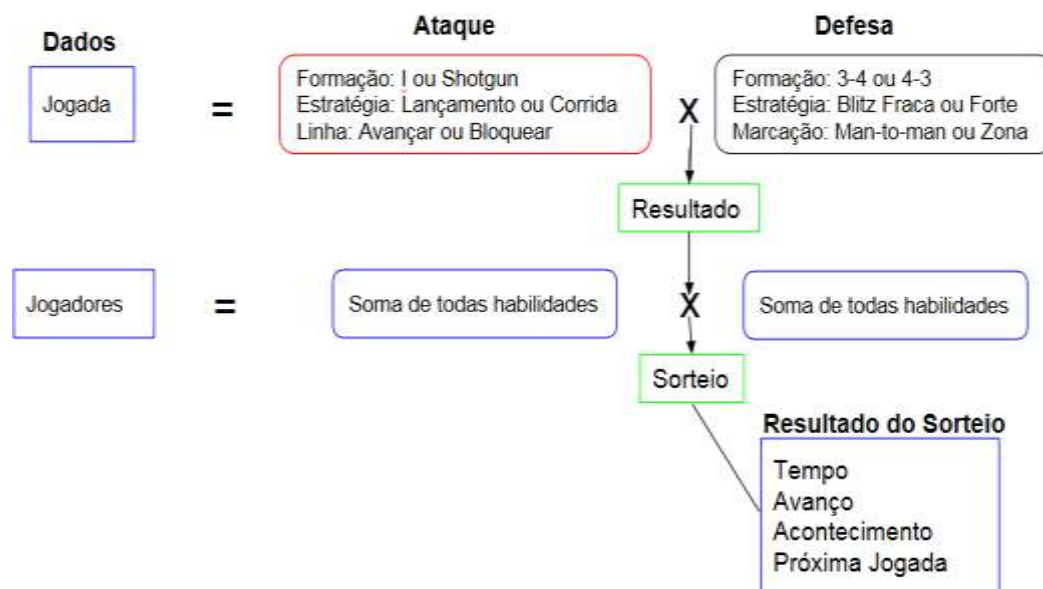


Figura 21. Funcionamento das jogadas de ataque contra defesa

Para elaborar este projeto foram aplicadas diversas matérias aprendidas no decorrer do curso: linguagem de programação, banco de dados, engenharia de software, análise orientada a objetos, desenvolvimento web e desenvolvimento de jogos.

Referências

- ABRAGAMES. **Plano diretor da promoção da indústria de desenvolvimento de jogos eletrônicos no Brasil:** diretrizes básicas. 2004. Disponível em: <http://www.abragames.org/uploads/5/6/8/0/56805537/plano_diretor_da_promocao_da_industria_de_desenvolvimento_de_jogos_eletronicos_no_brasil.pdf>, Acesso em: 10 out. 2018.
- BAIXAKI. *NFL Showdown: Football Manager*, Disponível em: <<https://www.baixaki.com.br/windows-8/nfl-showdown-football-manager.htm>>, Acesso em: 27 ago. 2018.
- CHACON, S.; STRAUB, B. *Pro Git*. 2. ed. Apress, 2018. Disponível em: <<https://git-scm.com/>>, Acesso em: 10 out. 2018.
- BEZERRA, E. **Princípios de Análise e Projeto de Sistemas com UML**. 3. ed. Rio de Janeiro: Elsevier, 2015.
- BRASFOOT. **Site Oficial**. Disponível em: <<http://www.brasfoot.com/>>, Acesso em: 27 ago. 2018.
- DUARTE, L. **Scrum e Métodos Ágeis: Um Guia Prático**. 5. ed. LuizTools, 2016.
- FEDERAÇÃO GAÚCHA DE FUTEBOL AMERICANO. **Mapeamento de público e renda da temporada 2018 do campeonato gaúcho de futebol americano**. Disponível em: <<http://fgfa.com.br/mapeamento-de-publico-e-renda-da-temporada-2018-do-campeonato-gaucha-de-futebol-americano/>>, Acesso em: 27 ago. 2018.
- _____. **Mapeamento de público e renda dos playoffs da temporada 2018 do campeonato gaúcho de futebol americano**. Disponível em:

<<http://fgfa.com.br/mapeamento-de-publico-e-renda-dos-playoffs-da-temporada-2018-do-campeonato-gaucha-de-futebol-americano/>>, Acesso em: 26 ago. 2018.

_____. **Mapeando o consumo de futebol americano através das transmissões ao vivo pelo Facebook – Campeonato Gaúcho 2018.** Disponível em: <<http://fgfa.com.br/mapeando-o-consumo-de-futebol-americano-atraves-das-transmissoes-ao-vivo-pelo-facebook-campeonato-gaucha-2018/>>, Acesso em: 26 ago. 2018.

IBOPE REPUCOM. **NFL tem mais de três milhões de fãs brasileiros, aponta pesquisa do IBOPE Repucom.** Disponível em: <<http://www.iboperepucom.com/br/releases/nfl-tem-mais-de-tres-milhoes-de-fas-brasileiros-aponta-pesquisa-do-ibope-repucom/>>, Acesso em: 28 ago. 2018.

_____. **IBOPE Repucom:** Mais de 14 milhões de internautas brasileiros são fãs de futebol americano. Disponível em: <<http://www.iboperepucom.com/br/releases/ibope-repucom-mais-de-14-milhoes-de-internautas-brasileiros-sao-fas-de-futebol-americano/>>, Acesso em: 28 ago. 2018.

_____. **Futebol Americano tem mais de 15 milhões de fãs no Brasil, aponta IBOPE Repucom.** Disponível em: <<http://www.iboperepucom.com/br/releases/futebol-americano-tem-mais-de-15-milhoes-de-fas-no-brasil-aponta-ibope-repucom/>>, Acesso em: 28 ago. 2018.

MENDES, G. **Artigo:** Futebol americano no Brasil, barreiras e desafios de um esporte amador. Disponível em: <<https://blogs.oglobo.globo.com/mvp/post/artigo-futebol-americano-no-brasil-barreiras-e-desafios-de-um-esporte-amador.html>>, Acesso em: 26 ago. 2018.

MCLAUGHLIN, B.; POLLICE, G. **Use a cabeça:** análise e projeto orientado ao objeto. 2 ed. Rio de Janeiro: Alta Books, 2007.

NIEDERAUER, J. **Desenvolvendo Websites com PHP.** 2. ed. São Paulo: Novatec, 2011.

NEWZOO, **Brazil Games Market 2018.** Disponível em: <<https://newzoo.com/insights/infographics/brazil-games-market-2018>>, Acesso em: 26 ago. 2018.

OLIVEIRA, F. **Artigo: Game Design:** Criando projetos de jogos digitais. Disponível em: <<http://www.fabricadegogos.net/posts/artigo-game-design-criando-projetos-de-jogos-digitais/>>, Acesso em: 14 nov. 2017.

PRESSMAN, Roger S. **Engenharia de software:** Uma abordagem profissional. 7. ed. Porto Alegre: AMGH: Bookman, 2011.

SANTOS, L. et al. Análise do Processo de Inserção e Desenvolvimento do Futebol Americano em Curitiba/PR. ALESDE, Curitiba, v. 7, n. 1, p. 25-38, junho de 2016.

SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. **Sistema de banco de dados.** 6. ed. Rio de Janeiro: Elsevier, 2012.

SILVEIRA, P. et al. **Introdução à Arquitetura e Design de Software:** uma visão sobre a plataforma Java. Rio de Janeiro: Elsevier, 2012.

TECHTUDO. **Review Madden NFL 18.** Disponível em : <<https://www.techtudo.com.br/review/madden-nfl-18.html>>, Acesso em: 26 ago. 2018.

WEILLER, T. A. *Game Start*: Lições de game design para seu videogame. São Paulo, 2015.