

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CÂMPUS CORNÉLIO PROCÓPIO
DIRETORIA DE GRADUAÇÃO E EDUCAÇÃO PROFISSIONAL
DEPARTAMENTO DE COMPUTAÇÃO
ENGENHARIA DE SOFTWARE**

CAIO THIZIO, DAFNY GARCIA, RAFAEL SANTOS

MEMORY-CLOUD

DOCUMENTO DE PROJETO DE SOFTWARE

CORNÉLIO PROCÓPIO

2018

SUMÁRIO

1	INTRODUÇÃO	2
1.1	CONTEXTO	2
1.2	JUSTIFICATIVA	2
1.3	PROPOSTA	3
1.4	ORGANIZAÇÃO DO DOCUMENTO	3
2	DESCRIÇÃO GERAL DO SISTEMA	5
2.1	OBJETIVOS(GERAIS E ESPECÍFICOS)	5
2.2	LIMITES E RESTRIÇÕES	5
2.3	DESCRIÇÃO DOS USUÁRIOS DO SISTEMA	5
3	DESENVOLVIMENTO DO PROJETO	7
3.1	TECNOLOGIAS E FERRAMENTAS	7
3.1.1	Linguagem de Programação	7
3.1.2	Sistema Gerenciador de Banco de Dados	7
3.1.3	Ferramentas	7
3.1.3.1	Desenvolvimento	7
3.1.3.2	Documentação	7
3.1.3.3	Gerenciamento	8
3.2	METODOLOGIA DE DESENVOLVIMENTO(CICLO DE VIDA E EQUIPE)	8
3.3	CRONOGRAMA PREVISTO	9
4	REQUISITOS DO SISTEMA	10
4.1	REQUISITOS FUNCIONAIS	10
4.2	REQUISITOS NÃO-FUNCIONAIS	10
4.3	DIAGRAMAS DE CASOS DE USO	11
5	ANÁLISE DO SISTEMA	12
5.1	ARQUITETURA DO SISTEMA	12
5.1.1	Infraestrutura	12
5.2	MODELO DO BANCO DE DADOS	13
5.2.1	Modelo Conceitual	13
5.2.2	Modelo Lógico	13
5.2.3	Dicionário de Dados	13
5.3	DIAGRAMA DE CLASSES	14
5.4	DIAGRAMA DE ATIVIDADES	14
6	IMPLEMENTAÇÃO	15
6.1	PROTÓTIPO DE TELAS	15
6.2	DESCRIÇÃO DO CÓDIGO	15
7	CONSIDERAÇÕES FINAIS	16
	REFERÊNCIAS	17

1 INTRODUÇÃO

1.1 CONTEXTO

Ferramentas para persistência de progressos de jogos Unity, no cenário atual, são pouco acessíveis para desenvolvedores. Serviços nesse segmento são caros, além de existirem poucas variedades. Nenhuma destas ferramentas possuem um código open source. Desenvolvedores de jogos de pequenas empresas, sofrem atualmente para que seus progressos de jogos possam ser salvos, enquanto trabalham em melhorias ou a continuidade de seus jogos.

As ferramentas gratuitas no mercado atualmente, trabalham com jogos na Steam ou com jogos completos, não sendo possível que o desenvolvedor teste e ainda assim mantenha seu progresso, sem a versão final.

Das ferramentas mais utilizadas, temos PlayFab e GameSparks. Para questão de comparação, usaremos a PlayFab. A ferramenta oferece:

- Multiplataforma de autenticação, para que os players possam jogar em vários dispositivos.
- Armazenamento de informações do jogador e estado do jogo na nuvem para que os dados nunca sejam perdidos.
- Compartilhamento de vários dispositivos e jogos.
- Compartilhamento com outros jogadores.
- Escolha do nível de acesso do cliente do jogo.
- Estatísticas do jogador para placares, matchmaking e segmentação. Suporte.

Para que as empresas possam utilizar esses serviços, o preço de início do pacote Professional é de US\$299/mês.

1.2 JUSTIFICATIVA

Com o crescente mercado de jogos e interesse de novos desenvolvedores tanto em startups quanto individuais, fazer parte da comunidade desenvolvedora de jogos vem sendo um desafio. Ferramentas como PlayFab têm acesso limitado por possuírem código fechado, alta complexidade de desenvolvimento e configuração e assinaturas dispendiosas.

O projeto de uma ferramenta de persistência de jogos será de muita importância tanto para usuários de jogos Unity quanto para desenvolvedores. A API de outras ferramentas, além de muito custosas são muito complicadas. A API a ser desenvolvida é mais simplificada, pois possui conceitos similares à API da própria Unity, além da vantagem de custo. Com esta nova abordagem, usuários normais poderão interagir gratuitamente com amigos do Facebook através de rankings. Aos desenvolvedores, o benefício virá pela tentativa de trazer um tempo menor de

desenvolvimento, tendo como um dos requisitos uma curva menor de aprendizado de uso da ferramenta.

A implementação do projeto teria como primeiro impacto ser uma alternativa open source aos serviços pagos. Por ser de código livre, cada desenvolvedor poderá personalizar o back-end seguindo o gosto pessoal e estilo de programação, além de poder personalizar o próprio servidor, de acordo com as preferências. Sobre finanças, reforça-se que é uma ferramenta open source e os usuários não precisariam pagar para fazer uso dela.

1.3 PROPOSTA

O projeto será uma ferramenta de servidor que guardará o progresso de jogos Unity. O cliente estará usando o Facebook como identificador de usuários, criando rankings e cada jogo será salvo somente uma vez para que progresso atual possa prescrever o anterior. O usuário deverá logar primeiramente no facebook para que o jogo seja salvo.

Para o desenvolvedor, teremos uma API simplificada onde o usuário desenvolvedor irá configurar na própria plataforma Unity o cliente e poderá usar o servidor personalizado ou disponibilizado por padrão. O ranking que será mostrado ao cliente também poderá ser personalizado pelo desenvolvedor.

1.4 ORGANIZAÇÃO DO DOCUMENTO

O documento está estruturado em tópicos, ou capítulos, onde cada tópico aborda uma ideia central e divide esta ideia em subtópicos.

O tópico seguinte aborda a Descrição Geral do Sistema, aonde serão apresentados objetivos gerais e específicos do sistema, bem como algumas de suas restrições e os usuários que poderão se beneficiar do produto.

O terceiro tópico trata do Desenvolvimento do Projeto, abordando algumas das tecnologias e ferramentas utilizadas para implementação do projeto; alguns dos métodos e processos utilizados para que o projeto se desenvolva, bem como os artefatos gerados durante os mesmos; traçar um plano de metas que ajudará na elaboração de um cronograma de desenvolvimento.

O quarto tópico trata dos requisitos envolvidos no projeto. Os requisitos serão levantados e classificados como funcionais ou não-funcionais e por meio de sua interpretação será possível a elaboração de diagramas de caso de uso para melhor visualização dos stakeholders e ações que os mesmos poderão performar no produto.

O quinto tópico trata da arquitetura do sistema. Analisar a arquitetura do software, fazer um mapeamento dos dados e como eles se distribuem no modelo lógico e apresentar diagramas de classes e de atividades.

O sexto tópico trata da implementação. Por meio de protótipos de tela e detalhamento da codificação, o documento retratará alguns aspectos do que foi desenvolvido no projeto.

O sétimo e último tópico discutirá os resultados da aplicação do produto, impactos e considerações finais.

2 DESCRIÇÃO GERAL DO SISTEMA

2.1 OBJETIVOS(GERAIS E ESPECÍFICOS)

Como foi descrito na seção anterior, um desenvolvedor de jogos Unity quando necessita de uma ferramenta de persistência de progresso online para algum de seus projetos, deve optar por restritos serviços que têm alto custo financeiro e possuem seu código fechado.

Como alternativa a esta situação, o objetivo geral deste projeto pode ser definido como: desenvolver uma solução open-source para o salvamento de progresso em jogos que utilizam a Engine Unity. Uma Engine nada mais é que uma biblioteca, um pacote de funcionalidades que são disponibilizadas para facilitar o desenvolvimento de um jogo e impedir que sua criação tenha seja feita do zero.

Por meio da delimitação do objetivo geral, é possível traçar alguns objetivos específicos que visam trazer mais portabilidade ao projeto:

- A ferramenta deve ter seu código aberto e totalmente personalizável, para que desenvolvedores possam utilizá-la da forma que melhor atender suas demandas;
- De forma a unificar um pouco do processo, usuários de jogos que utilizem a ferramenta deverão iniciar sessão na rede social Facebook para que o servidor colete as informações;
- Informações geradas pelos progressos dos jogos estarão disponíveis em um website para que desenvolvedores e usuários interajam com a ferramenta, por meio da criação de rankings.

2.2 LIMITES E RESTRIÇÕES

Alguns dos limites do projeto inicial contemplam:

- O sistema será monoplataforma para autenticação, utilizando apenas o Facebook para logar-se. Outras formas de autenticação podem ser implementadas posteriormente;
- O sistema restringe seu funcionamento a jogos que utilizam a engine Unity para salvamento de progressos;
- A única forma de interação entre usuários será, inicialmente, feita por meio dos rankings. Outras formas de interação deverão ser consideradas no futuro, como compartilhamento de informações e progressos entre usuários.

2.3 DESCRIÇÃO DOS USUÁRIOS DO SISTEMA

Os principais usuários finais do projeto deverão ser:

Desenvolvedor O desenvolvedor de jogos com a engine Unity será o principal usuário da ferramenta. Por meio dela, ele deverá ser capaz de implementar, dentro de seus projetos, formas para manter a persistência dos progressos realizados dentro dos jogos. Como a ferramenta possui código aberto, o desenvolvedor poderá adaptá-la da forma que melhor atender suas demandas. Ele poderá contar com bibliotecas predefinidas para evitar que o jogo perca seu progresso toda vez que seja reiniciado.

Jogador O jogador de jogos que utilizam a ferramenta para persistir as informações de progresso poderá acompanhar seus avanços por meio de um site, cuja autenticação se dá por meio de login pelo Facebook. No site será possível acompanhar um ranking com amigos que também joguem games que utilizam a ferramenta.

3 DESENVOLVIMENTO DO PROJETO

3.1 TECNOLOGIAS E FERRAMENTAS

3.1.1 Linguagem de Programação

Servidor NodeJS(ES8) e GraphQL

Cliente C# e GraphQL

3.1.2 Sistema Gerenciador de Banco de Dados

MongoDB para persistência e Redis para cache.

3.1.3 Ferramentas

3.1.3.1 Desenvolvimento

- Unity3D 2017.0f3
- Visual Studio Code
- JetBrains Rider
- Kitematic
- GraphQL App
- GitHub
- Heroku
- MongoDB Atlas
- RedisLabs
- Travis-CI

3.1.3.2 Documentação

- Astah
- MySQL Workbench
- TeXstudio
- GitHub
- Google Drive
- Cloudcraft

3.1.3.3 Gerenciamento

- GitHub(boards, projects, issues, milestones)
- Discord

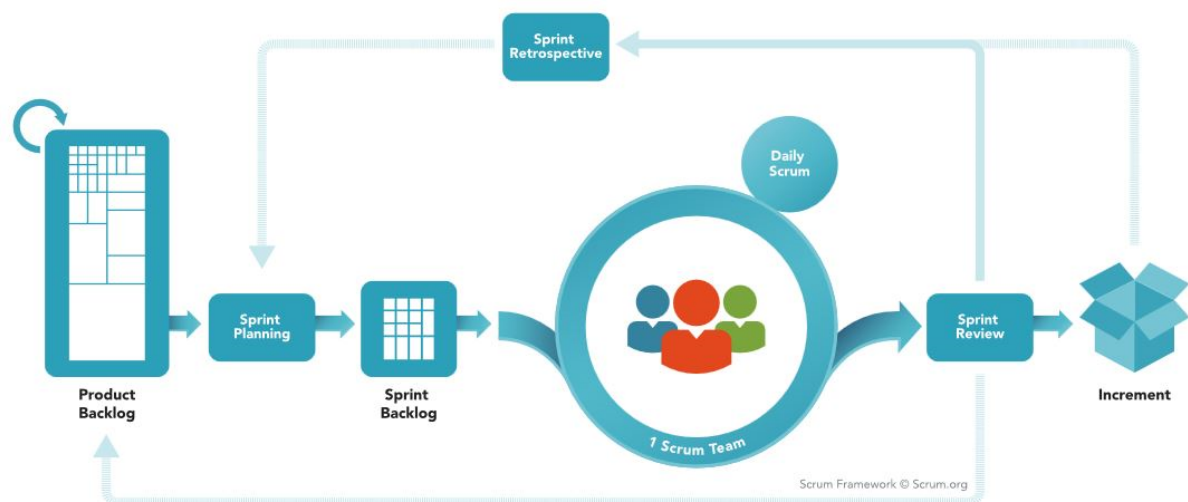
3.2 METODOLOGIA DE DESENVOLVIMENTO(CICLO DE VIDA E EQUIPE)

O desenvolvimento será iterativo e incremental a equipe é composta por:

Product Owner asd

Development Team asd

Scrum Master asd



Fonte: (SCRUM, 2018)

Product Backlog funcionalidades ordenadas de acordo com sua importância.

Sprint Planning reunião para definir o Sprint Backlog.

Sprint Backlog lista de funcionalidades a serem implementadas no Sprint.

Sprint fase de implementação do que foi definido no Sprint Backlog.

Daily Scrum reuniões diárias para deixar o time a par do desenvolvimento, realizadas via Discord.

Sprint Review reunião para definir .

Sprint Retrospective reunião de retrospectiva do Sprint, para definir o que pode ser melhorado no processo.

3.3 CRONOGRAMA PREVISTO

4 REQUISITOS DO SISTEMA

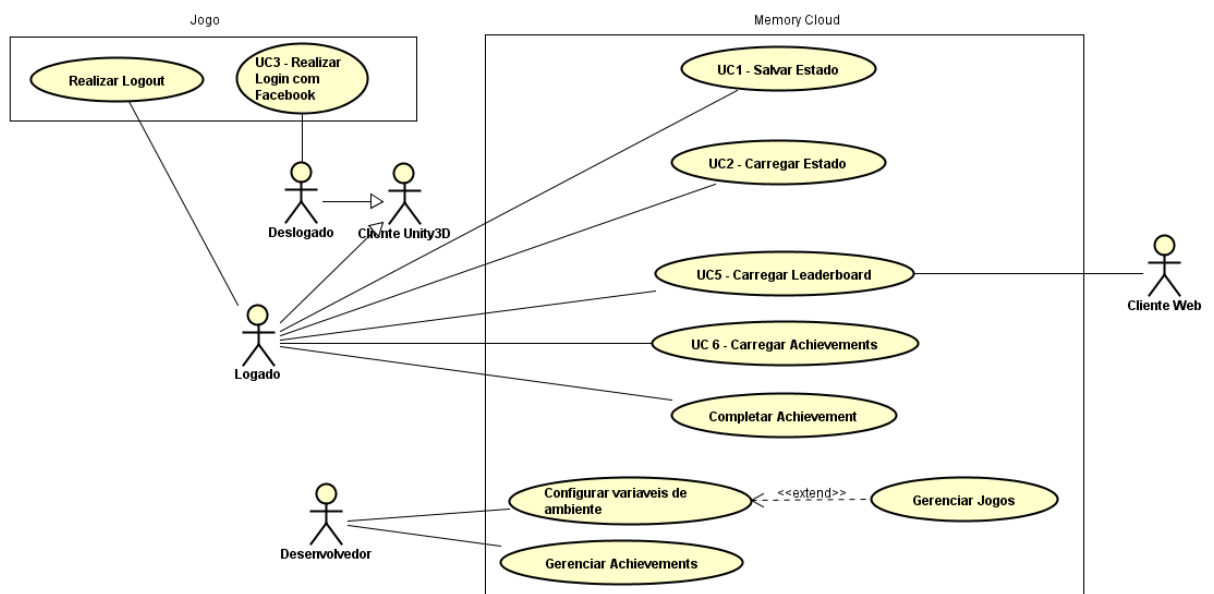
4.1 REQUISITOS FUNCIONAIS

ID	Funcionalidade	Prioridade
RF1	Salvar inteiros, floats, booleans e strings.	Essencial
RF2	Carregar inteiros, floats, booleans e strings.	Essencial
RF3	Usar o Facebook para realizar login.	Essencial
RF4	Suportar múltiplos jogos.	Essencial
RF5	Gerar Leaderboards a partir dos inteiros.	Importante
RF6	Sistema de achievements.	Desejável

4.2 REQUISITOS NÃO-FUNCIONAIS

ID	Requisito	Categoria
RNF1	O sistema deve ser configurável através de variáveis de ambiente.	Usabilidade
RNF2	Save deve ser feito em uma requisição ao servidor.	Usabilidade
RNF3	Load deve ser feito em uma requisição ao servidor.	Usabilidade
RNF4	Os Leaderboards devem ficar no cache por 120 segundos.	Usabilidade

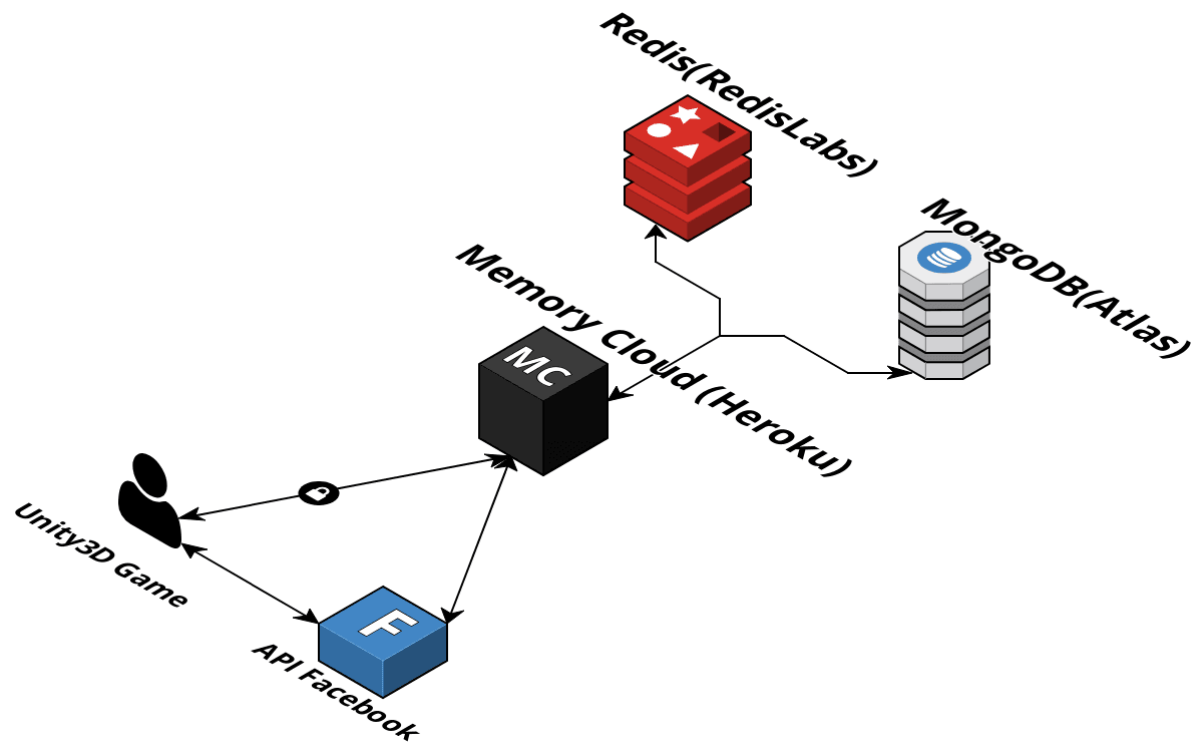
4.3 DIAGRAMAS DE CASOS DE USO



5 ANÁLISE DO SISTEMA

5.1 ARQUITETURA DO SISTEMA

5.1.1 Infraestrutura



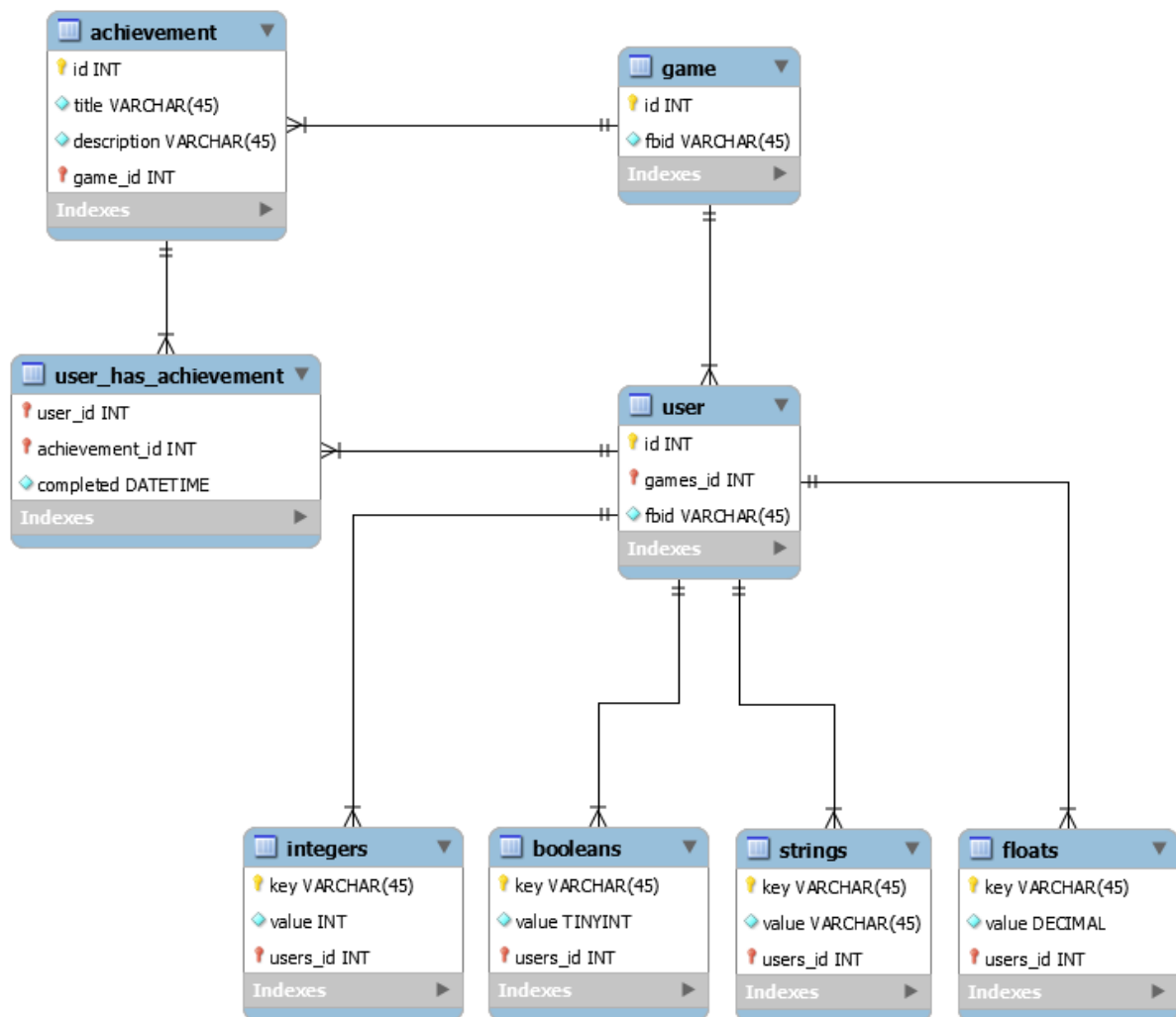
Fonte: <https://cloudcraft.co>

5.2 MODELO DO BANCO DE DADOS

5.2.1 Modelo Conceitual

O sistema gerenciador de banco de dados usado é não relacional, os dados são guardados como documentos em formato JSON.

5.2.2 Modelo Lógico

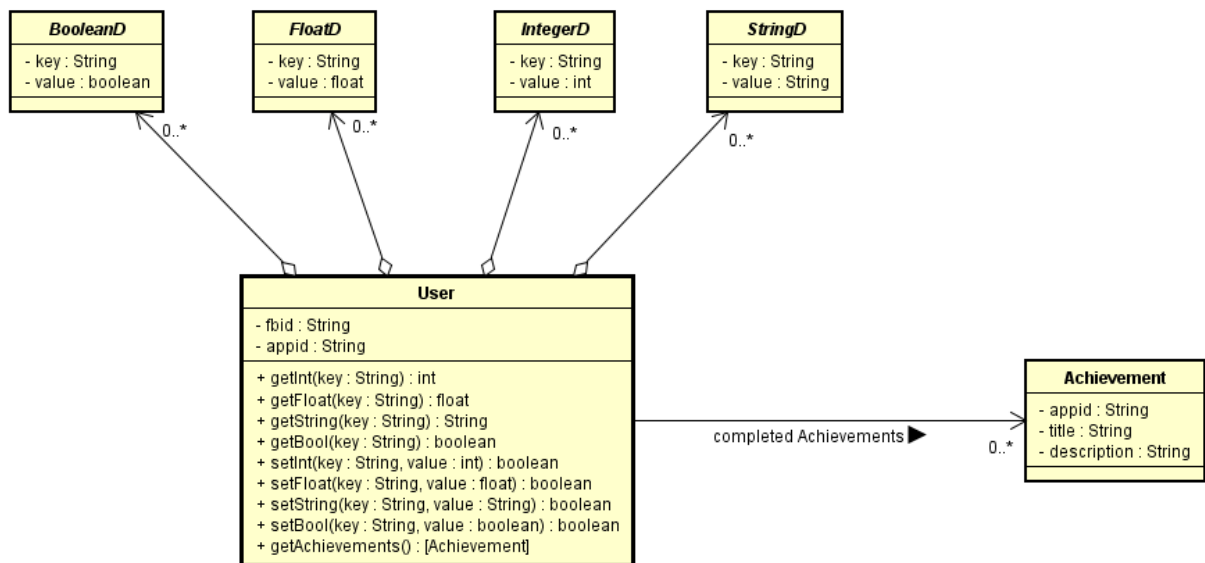


Fonte: MySQL Workbench

5.2.3 Dicionário de Dados

- 🔑 Chave Primária
- 💎 Campo Obrigatório
- 🔗 Chave Estrangeira

5.3 DIAGRAMA DE CLASSES



5.4 DIAGRAMA DE ATIVIDADES

6 IMPLEMENTAÇÃO

6.1 PROTÓTIPO DE TELAS

6.2 DESCRIÇÃO DO CÓDIGO

7 CONSIDERAÇÕES FINAIS

REFERÊNCIAS

SCRUM. **Scrum Framework**. 2018. Disponível em: <<https://www.scrum.org/resources/what-is-scrum>>. Acesso em: 9 de abril de 2018. Citado na página 8.