

## PROJETO PICMoney

### Requisitos da disciplina Modelagem de Software e Arquitetura de Sistemas

#### INTEGRANTES DO PROJETO e RA'S

|                              |   |          |
|------------------------------|---|----------|
| Bruno De Brito Faccio        | - | 25028271 |
| José Antônio Da Silva Aguiar | - | 25028150 |
| Luiza Sardinha Alde          | - | 25028264 |
| Maria Luisa Silva Lima       | - | 25028139 |

São Paulo  
2025

## SUMÁRIO

|   |          |
|---|----------|
| <b>1 INTRODUÇÃO</b>   | <b>3</b> |
| <b>2. DOCUMENTO DE ABERTURA DO PROJETOS</b>                 | <b>4</b> |
| 2.1 – Project Charter                                       | 4        |
| 2.2 – Histórias do Usuário                                  | 5        |
| <b>3. DESIGN SPRINT – Ideação e prototipação do desafio</b> | <b>6</b> |
| 3.1 Desafio   | 6        |
| 3.2 Entender Mapear   | 6        |
| 3.3 Ideação – desenho da solução (trilha do usuário)        | 6        |
| 3.4 Prototipagem  | 6        |
| <b>4. REQUISITOS DE SISTEMA</b>                             | <b>6</b> |
| 4.1 REQUISITOS FUNCIONAIS DE SOFTWARE                       | 6        |
| 4.2 REQUISITOS NÃO FUNCIONAIS DE SOFTWARE                   | 7        |
| <b>5. CASOS DE USO</b>                                      | <b>8</b> |
| <b>6. DIAGRAMA DE CLASSE</b>                                | <b>8</b> |
| <b>7. ARQUITETURA DO SISTEMA</b>                            | <b>8</b> |
| <b>8. REFERÊNCIAS BIBLIOGRÁFICAS</b>                        | <b>8</b> |

## 1 INTRODUÇÃO

Este projeto consiste no desenvolvimento de uma aplicação desktop gamificada para aumentar o engajamento dos usuários. O foco é facilitar a descoberta e a exploração de informações sobre a localização, e a categoria de cupons oferecidos pela PICMoney.

- Ver documento do [Projeto PI](#) – Gamificação PICMoney.

### BMJL PROJECT

**Nome da Instituição:** PicMoney

**Objetivo da Aplicação:**

ADS1 - O objetivo desta gamificação é atender aos requisitos da empresa, criando um ambiente interativo e educativo. A proposta é engajar os usuários em temas de **história** e **matemática**, utilizando o aprendizado como entretenimento. A premiação dos usuários consiste em vouchers de cupons de desconto, obtidos através de pontuação ou conclusão de marcos.

**Desafio:**

Desenvolver uma aplicação desktop gamificada que promova a interação do usuário com a localização e os tipos de cupons oferecidos pela PICMoney. O projeto será desenvolvido mediante a aplicação prática dos conhecimentos obtidos através das disciplinas do semestre.

### Personas a Serem Atendidas:

- **Usuário Final:** Jovens 15+.  
O jogador deverá aplicar suas habilidades e conhecimentos lógicos na resolução de enigmas para alcançar o objetivo principal: a conquista de cupons PICMoney;
- **Product Owner:** PICMoney  
A empresa por meio da gamificação, busca expandir o seu público alvo de maneira mais divertida e instrutiva.

**Recursos:** <https://www.picmoney.shop/>



## 2. DOCUMENTO DE ABERTURA DO PROJETOS

### 2.1 – Project Charter

#### Prefácio

Stakeholder principal, CEO PICMoney

#### Introdução

O Jogo foi desenvolvido com objetivo de recompensar o usuário por sua evolução educativa, atendendo os critérios do cliente. A aplicação utiliza uma tela de cadastro/login que solicita ao usuário um username e senha. Em seguida, o usuário é direcionado a uma tela inicial que possui três componentes:

- **Novo Jogo:** O início da experiência, oferecendo a possibilidade de escolha do usuário entre dois temas e três dificuldades (Fácil, Médio e Difícil);
- **Ranking:** Exibe a pontuação do usuário e dos outros jogadores que utilizaram o mesmo computador;
- **Sair:** Executa o comando para fechar o jogo.

#### Glossário

*Username: O Nome do Usuário ou Apelido (Nickname).*

*Ranking: Lista que guarda informações ou marcos como pontuações e tempo.*

*Login: É a entrada feita após o usuário informar os dados corretos já pré cadastrados no sistema.*

*StakeHolder: Todas as partes interessadas no sistema ou programa.*

*CRUD: Criar, Ler, Atualizar e Deletar (Create, Read, Update and Delete)*

*Desktop: Parte da interface gráfica de sistemas operacionais que exibe, no vídeo, representações de objetos, também é um sinônimo para “Computador”.*

*Vouchers: Premiações que proporcionam descontos ou alguns bônus.*

## Definição de requisitos de usuário

- **Requisitos funcionais:** Cadastro e autenticação de usuários, sistemas de pontuação, banco de dados, ranking de usuários, escolha de níveis de dificuldade, palavras cruzadas, informações visuais de acertos e erros;
- **Requisitos não funcionais:** Aplicação offline, tempo de resposta inferior a 1 segundo, utilização de banco relacional, interface intuitiva e compatível com desktop, estrutura de código modular.

## Arquitetura do sistema

A arquitetura do Sistema foi desenvolvida de forma simples e intuitiva. O foco é a jornada do usuário, na seguinte sequência:

- **Acesso:** O usuário iniciará com o seu cadastramento criando o seu username e senhas para a autenticação;
- **Menu principal:** Após a efetivação do cadastramento, o usuário será direcionado para a tela principal, que vai oferecer a opção de escolha entre “Novo jogo”, “Ranking” e “Sair”;
- **Configuração do jogo:** Ao selecionar a opção “Novo Jogo” o usuário vai poder escolher entre dois temas, sendo eles “Matemática” e “História”. E a opção de definição de dificuldades entre “Fácil”, “Intermediário” e “Difícil”;
- **Jogabilidade:** O objetivo do jogo é o usuário acertar 10 palavras no menor tempo possível. Tendo a possibilidade de ganhar pontos, e com esses pontos receber seus cupons. Ao finalizar o jogo, ele será direcionado a uma tela de ranking;

- **Final do jogo:** O usuário terá a visibilidade do ranking obtido por ele e pelos demais jogadores. Em seguida, o usuário terá duas opções: “Jogar novamente” ou “Tela Principal” onde poderá recommençar o jogo.

## Especificação de requisitos do sistema

### Requisitos Funcionais.

**Tela de login:** O Sistema deve guardar e registrar os acessos do usuário;

**Sistema de pontuação:** O sistema deve coletar e guardar os pontos que o usuário alcançou com base no seu tempo;

**Validação:** O sistema deve validar as letras e palavras inseridas pelo usuário;

**Dificuldade:** O sistema deve atrelar a recompensa do jogador ao tempo em que ele levou para resolver a tela de jogo;

**Ranking:** O sistema deve coletar a pontuação dos jogadores com base em seu tempo e gerar um relatório com todos os jogadores e suas pontuações.

### Requisitos Não Funcionais.

**Jogo:** O jogo deve funcionar offline sem a necessidade de que o jogador esteja conectado a internet;

**Interface:** O jogo deverá funcionar apenas em desktops;

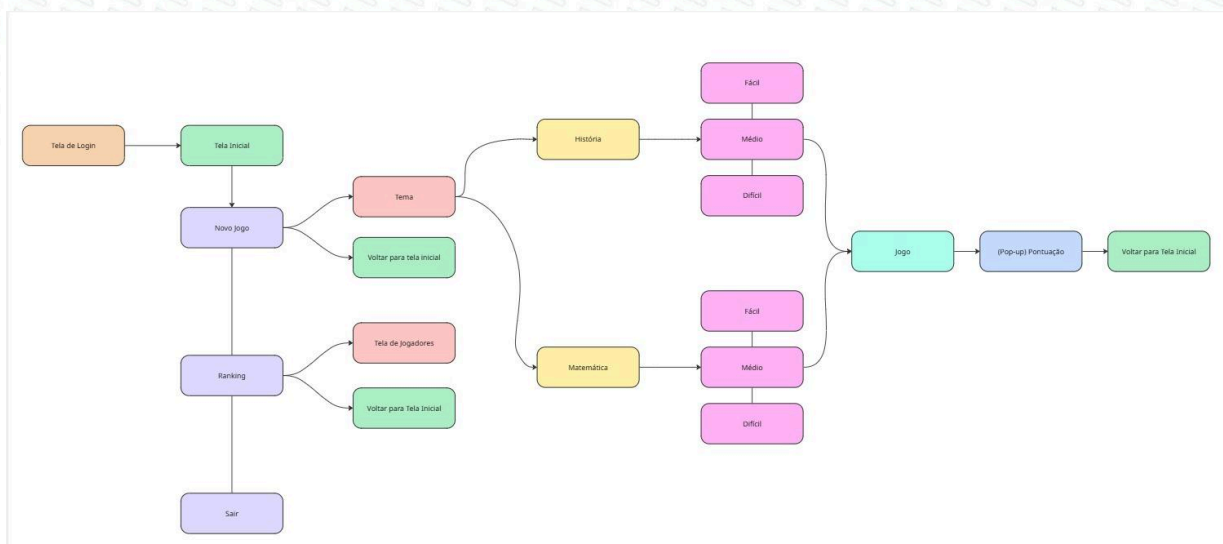
**Tempo:** O tempo de resposta do jogo deverá ser de 1 segundo ou menos em qualquer aplicação;

**Bancos de Dados:** O Sistema deverá guardar apenas o nome do usuário, pontuação e seu tempo;

**Sistema:** O sistema deve seguir um fluxo claro entre as telas do programa.

### Modelos do sistema





## Evolução do sistema

O jogo tem potencial, e requer uma evolução em aspectos funcionais e gráficas. Porém, um dos pontos de evolução a se destacar é a memória de armazenamento de jogadores e suas pontuações, nível técnico de evolução das funcionalidades como “Novo Jogo” e “Ranking”. Entre algumas melhorias no banco de dados e código nativo da aplicação que precisam de melhorias.

## Apêndices

### Requisitos Mínimos:

- **Sistema Operacional:** Windows 10, macOS 10.15 (Catalina), Ubuntu 20.04;
- **Processador:** Intel Core i3 de 2.0 GHz ou equivalente AMD;
- **Memória RAM:** 4 GB;
- **Placa de Vídeo:** Compatível com OpenGL 3.3;
- **Armazenamento:** 250 MB de espaço livre.

### Requisitos Recomendados:

- **Sistema Operacional:** Windows 11, macOS 12 (Monterey) ou superior, Ubuntu 22.04 ou superior;
- **Processador:** Intel Core i5 de 2.5 GHz ou equivalente AMD;
- **Memória RAM:** 8 GB;

- **Placa de Vídeo:** Dedicada com 1 GB de VRAM (NVIDIA GeForce, AMD Radeon);
- **Armazenamento:** 500 MB de espaço livre.

**Banco de Dados:**

Tela de login > ID\_Usuario > Senha.

Tela Inicial > ID\_Novo Jogo > Matemática > História.

Dificuldade > Fácil > Médio > Difícil.

Ranking > ID\_Cupom > Tempo > Pontuação.

## 2.2 – Histórias do Usuário

**1° usuário:** Ana tem 34 anos, mora em Belo Horizonte e vive uma rotina corrida entre trabalho, casa e filhos . Com um orçamento apertado, ela adora economizar e ama cupons de desconto, mas o problema é que ela não sabe onde achar esses cupons. Já tentou alguns sites, mas os códigos não atuam da maneira correta. Ela quer economizar sem perder tempo, e sem cair em pegadinhas.

**2° usuário:** Lucas tem 15 anos. Está no ensino médio e passa boa parte do tempo online. Gosta de jogos, tecnologia e de comprar roupas estilosas, mas sempre com o dinheiro contado. já ouviu falar em cupons de desconto e vê muita gente usando, mas não sabe de que forma conseguir.

## 3. DESIGN SPRINT – Ideação e prototipação do desafio

### 3.1 Desafio

O desenvolvimento de uma aplicação desktop gamificada que possibilite aos usuários a exploração de dicas de localização e a categoria de cupons oferecidos



pela PICMoney. O projeto será desenvolvido mediante a aplicação prática dos conhecimentos obtidos através das disciplinas do semestre.

## 3.2 Entender Mapear

1 - **Tetris**, com ênfase em somas matemáticas simples que ensinam cálculos simples.

1.1 - Inicialmente pensamos em criar um jogo baseado na mecânica do Tetris, criando uma estrutura que criaria cálculos matemáticos aleatórios e os transformaria em contas simples dentro de cada bloco.

Descartamos a ideia devido a sua complexidade, tendo em vista que neste primeiro semestre não possuímos um conhecimento amplo em programação, para criar estruturas avançadas em um prazo limitado.

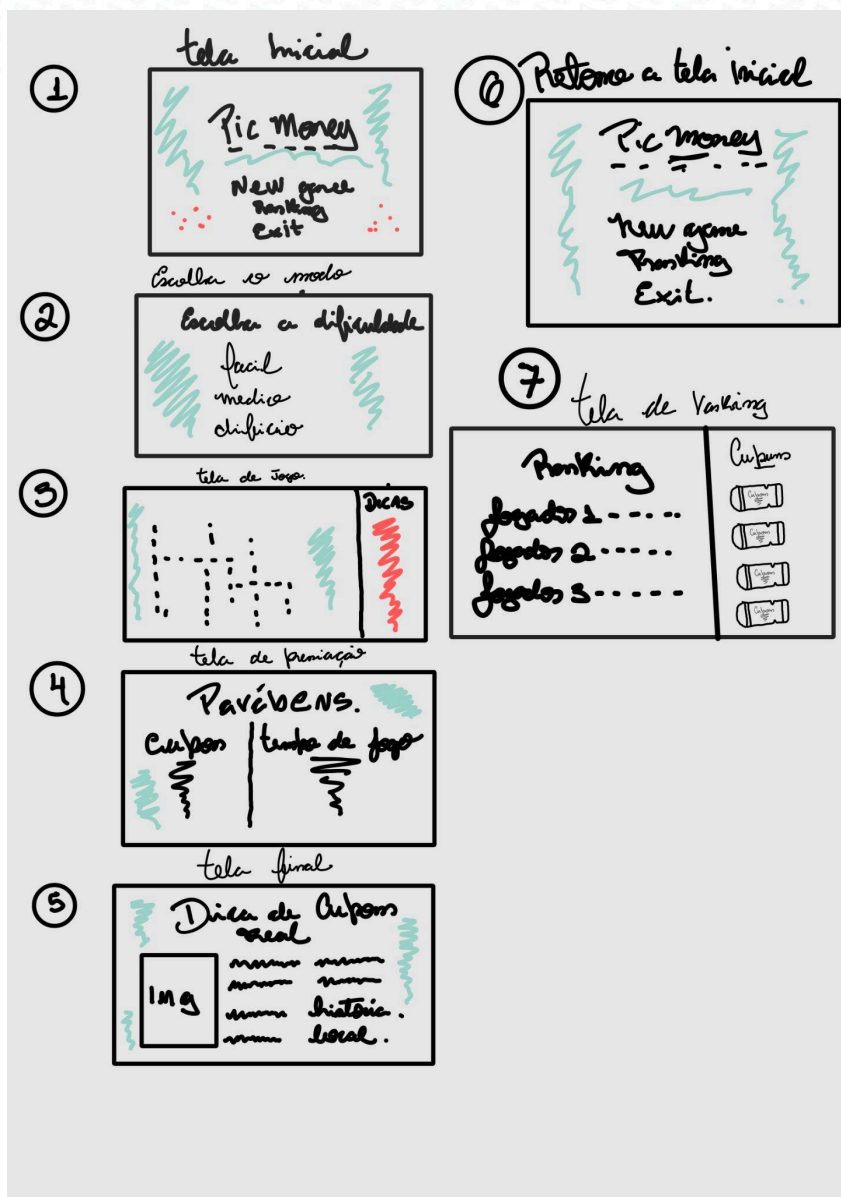
2 - **Palavras Cruzadas**, com ênfase em ensinar mais sobre a história das imigrações em São Paulo.

2.1 - Essa é a nossa proposta atual, optamos em criar o jogo que possua uma estrutura simples. O sistema vai possuir algoritmos de premiação que concedem ao usuário de acordo com a sua pontuação e no tempo de resolução. O jogo de palavras cruzada terá dois temas, divididos entre **História** e **Matemática**, que possibilitará três níveis (Fácil, Médio e Difícil). A complexidade das palavras varia de acordo com o nível, com um padrão de dez palavras por tela.

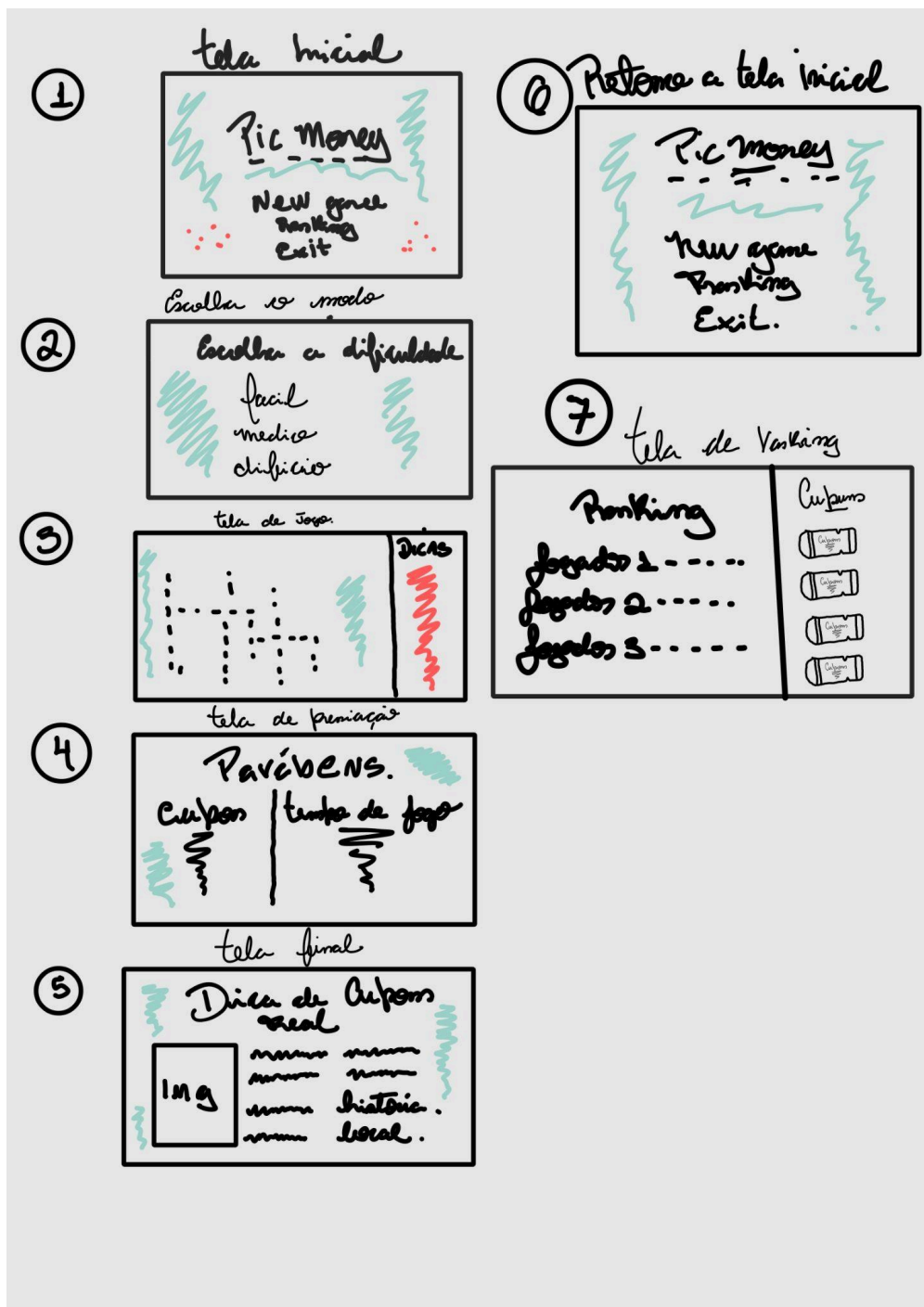
### Novo Desafio

Criar um jogo clone de outro jogo que seja inovador mas que respeite toda a qualidade que o jogo original possuía, por conta do jogo ser um clone ele deve respeitar a jogabilidade do jogo, todos os aspectos e também todos requisitos funcionais e não funcionais aproveitando deles para criar novas funcionalidades.

### 3.3 Ideação – desenho da solução (trilha do usuário)



### 3.4 Prototipagem





## 4. REQUISITOS DE SISTEMA

### 4.1 REQUISITOS FUNCIONAIS DE SOFTWARE

- Sistema de pontuação por tempo, ou seja, quanto menos tempo demorar para resolver a “cruzadinha” maior é a sua pontuação;
- Escolha de nível de dificuldade.;
- Ranking de usuários com as maiores pontuações;
- Finalização do jogo.

| RFS 01           |  |
|------------------|--|
| <b>Função</b>    | Cadastro de Usuário.   |
| <b>Descrição</b> | Realizar um CRUD (Create, Read, Update e Delete),<br>Sendo obrigatórios os campos.   |
| <b>Entradas</b>  | Nome, senha.   |
| <b>Fonte</b>     | Do usuário.  |
| <b>Saídas</b>    | Informar cadastro realizado, ou se já houver, dar cadastro “já existente”.   |
| <b>Ação</b>      | <ul style="list-style-type: none"><li>- O usuário informa dados de login (nickname e senha), apresenta a tela para inserir as informações, exigir que todos os campos sejam preenchidos e gravar o registro.</li><li>- Para deletar um usuário exigir confirmação.</li><li>- Sempre que o usuário confirmar uma transação de CRUD, retornará uma mensagem de sucesso ou falha.</li></ul> |

| RFS 02           |   |
|------------------|---|
| <b>Função</b>    | Indicar se as palavras estão certas ou erradas.   |
| <b>Descrição</b> | Realizar verificação.   |
| <b>Entradas</b>  | As palavras.  |
| <b>Fonte</b>     | Do usuário.   |
| <b>Saídas</b>    | Cor verde ou vermelho.  |
| <b>Ação</b>      | Ao término do jogo o usuário irá verificar se suas respostas estão certas ou erradas, sendo indicado pelas cores verde (corretas) e vermelha (erradas). |

## 4.2 REQUISITOS NÃO FUNCIONAIS DE SOFTWARE

- A aplicação deverá ser feita em modo offline;
- Interface desktop;
- Tempo de resposta 1 segundo;
- O Sistema deve armazenar, usuário, pontos e tempo;
- Fluxo de interface fixo. Tela de login > Tela inicial > Escolher tema > Escolher dificuldade > Tela de Jogo > Tela de Ranking > Tela Inicial.

## 5. CASOS DE USO

Apresentar 3 casos de uso do sistema

## 6. DIAGRAMA DE CLASSE

## 7. ARQUITETURA DO SISTEMA

## 8. REFERÊNCIAS BIBLIOGRÁFICAS

- SOMMERVILLE, I. **Engenharia de Software**. 11ª Edição. São Paulo: Pearson Addison-Wesley, 2017.
- <https://www.picmoney.shop/>