
PROJETO
QUIZ
Start

PROJETO REALIZADO POR:

Hugo Magalhães
José Junior
Patrícia Correia

METODOLOGIA

RESPONDER AOS PEDIDOS DOS CLIENTES DE FORMA DINÂMICA, COM GRANDE FLEXIBILIDADE E ALTA PRODUTIVIDADE?

O MODELO AGILE ASSENTA NUMA ABORDAGEM INTERATIVA. CARACTERIZA-SE POR DIVIDIR CADA PROJETO EM PEQUENAS PARTES QUE DEVEM SER COMPLETADAS SEMANALMENTE - OU EM CURTOS PERÍODOS DE TEMPO. ASSIM, TORNA-SE MAIS SIMPLES COMPREENDER PRIORIDADES E INTRODUZIR MUDANÇAS NO DESENVOLVIMENTO DO PROJETO, CASO SEJA NECESSÁRIO.

PLANEAMENTO

Como um grande formador sempre diz que um gênio já dizia...

Se eu tivesse uma hora para resolver um problema, gastaria 55 minutos pensando no problema e 5 minutos pensando nas soluções.

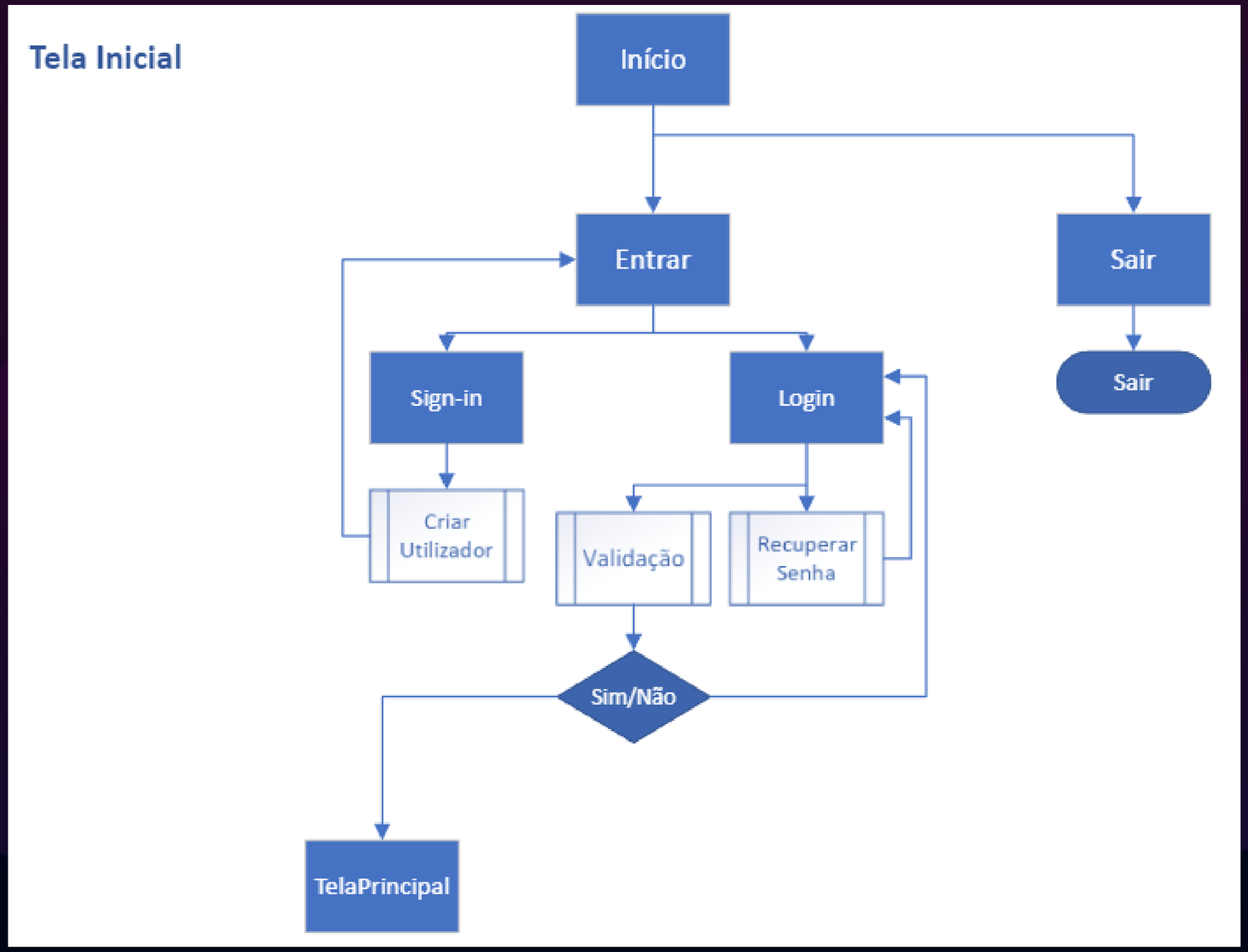
- Albert Einstein

- DEFINIÇÃO DE REQUISITOS
- FLUXOGRAMA
- DIVISÃO DE TAREFAS

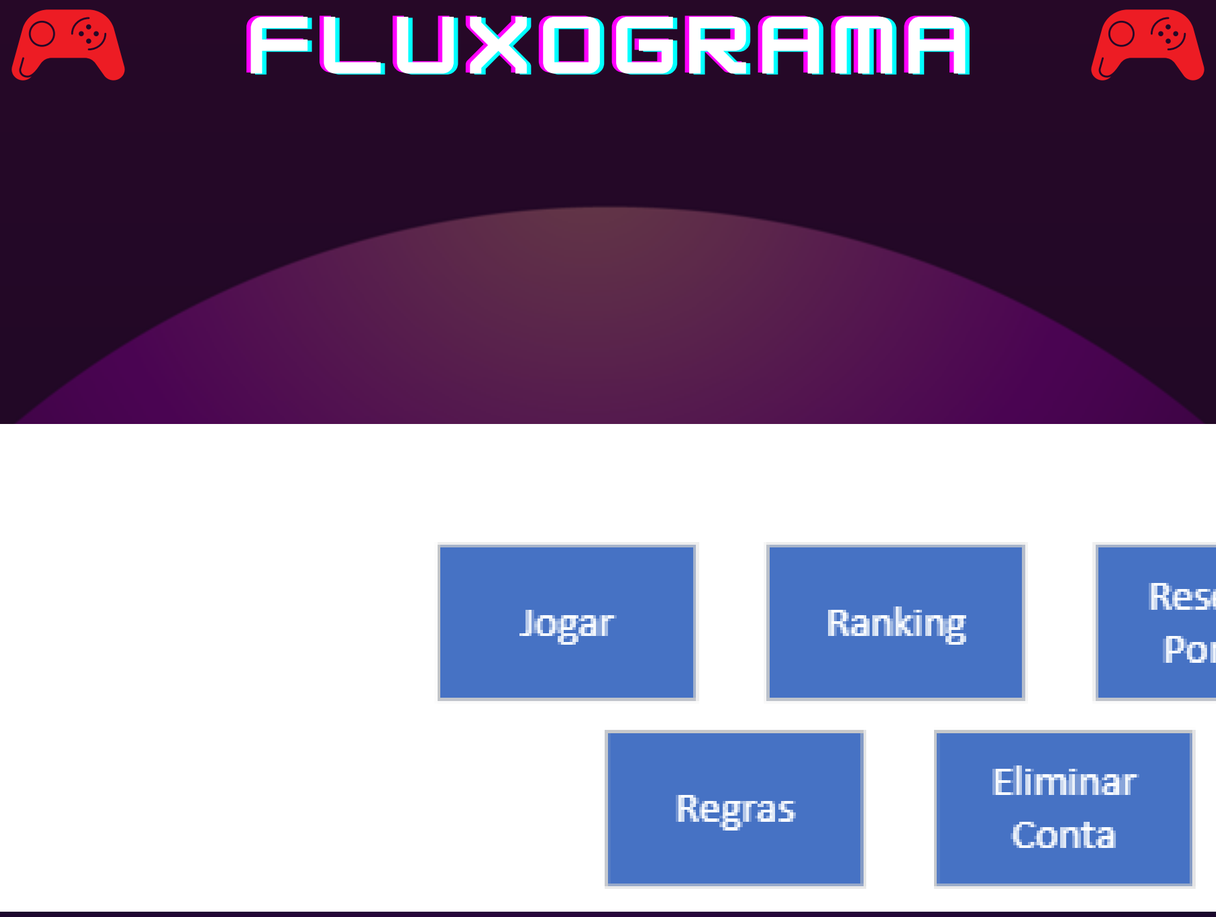
VALIDAÇÃO DE REQUISITOS

Requisitos	Grupo de Requisitos	Prioridade de Requisitos
Banco de dados TXT	Requisito funcional	Must Have
Definir Temas para questões	Requisito funcional	Must Have
Regras	Requisito não funcional	Could Have
Criar usuário	Requisito funcional	Must Have
Iniciar login	Requisito funcional	Must Have
Recuperar password	Requisito funcional	Could Have
Eliminar Conta	Requisito não funcional	Could Have
Definir tempo das questões	Requisito não funcional	Could Have
Definir pontuaçao	Requisito não funcional	Must Have
3 Dificuldades das questões	Requisito não funcional	Could Have
Tabela de lideres	Requisito não funcional	Could Have
Definir 10 questões com 4 escolhas multiplas em cada tema	Requisito funcional	Must Have
Resetar pontos	Requisito não funcional	Could Have
Escrever em C++	Requisito design	Must Have
Embelezar janelas	Requisito design	Could Have
Feedback da resposta tem de ser imediata	Requisito não funcional	Could Have

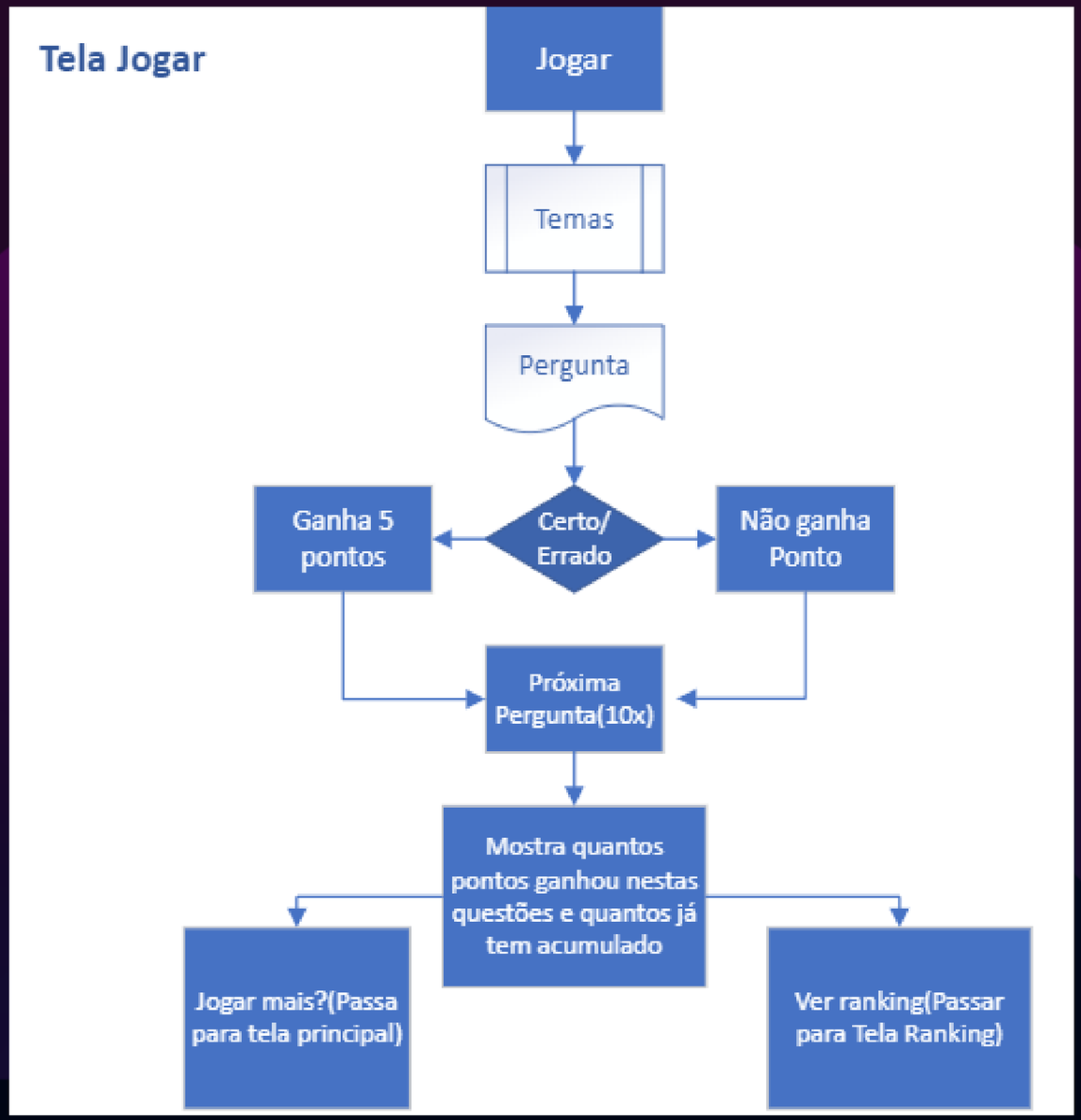
FLUXOGRAMA


Tela Inicial
Início
Entrar
Sair
Sign-in
Login
Sair
Criar Utilizador
Validação
Recuperar Senha
Sim/Não
TelaPrincipal

FLUXOGRAMA


Tela Principal
Jogar
Ranking
Reset de Pontos
Regras
Eliminar Conta

FLUXOGRAMA

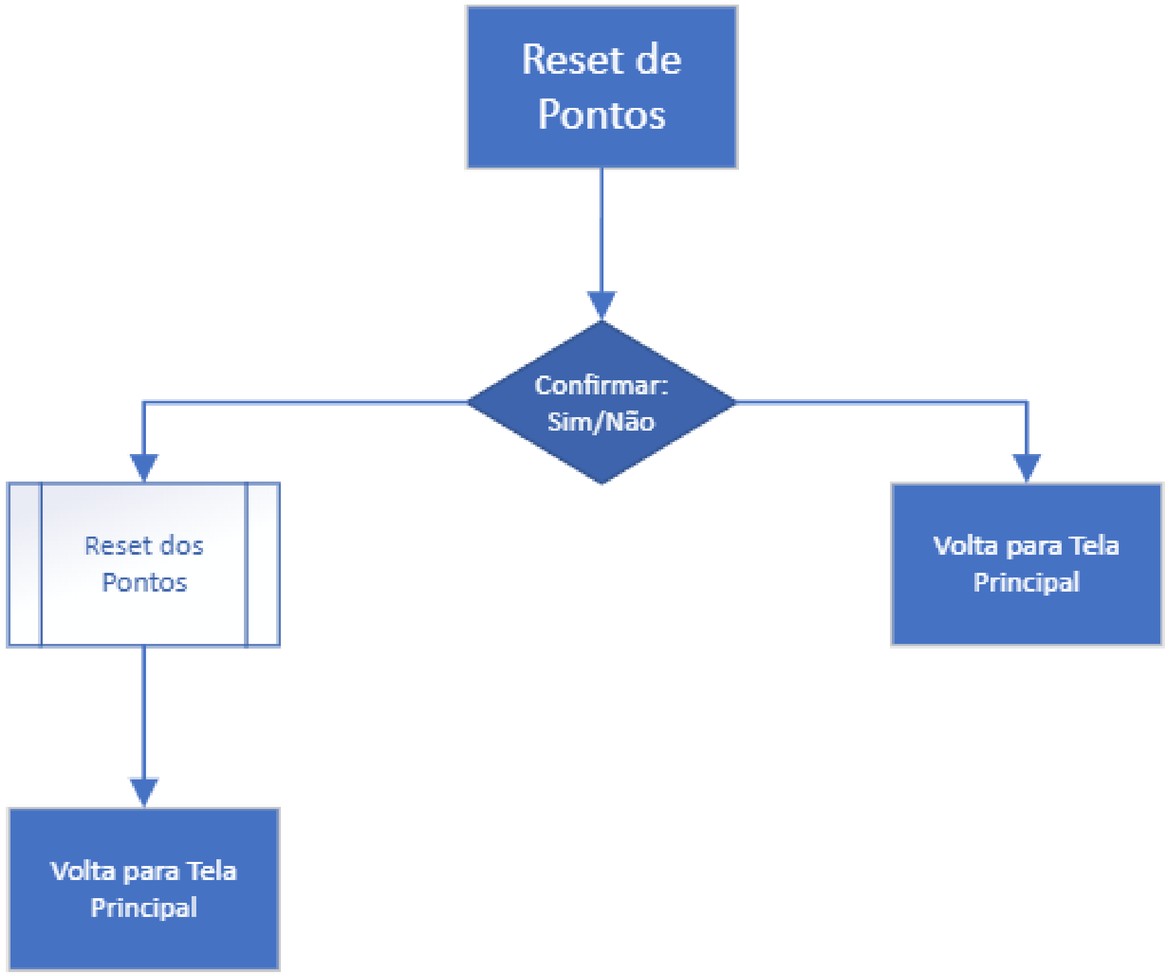

Tela Jogar
Jogar
Temas
Pergunta
Ganha 5 pontos
Certo/ Errado
Não ganha Ponto
Próxima Pergunta(10x)
Mostra quantos pontos ganhou nestas questões e quantos já tem acumulado
Jogar mais?(Passa para tela principal)
Ver ranking(Passar para Tela Ranking)

FLUXOGRAMA

Tela Ranking

FLUXOGRAMA

Tela Reset Pontos

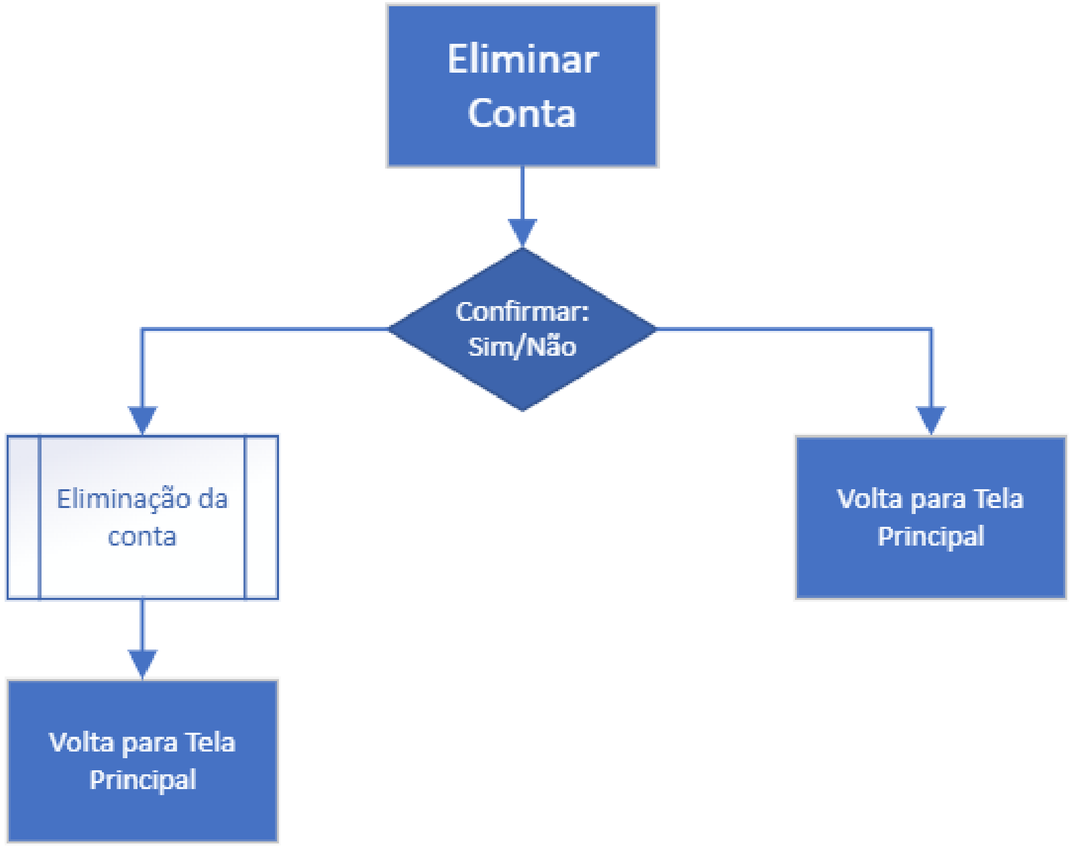


FLUXOGRAMA

Tela Regras

FLUXOGRAMA

Tela Eliminar Conta



DIVISÃO DE TAREFAS

New 0

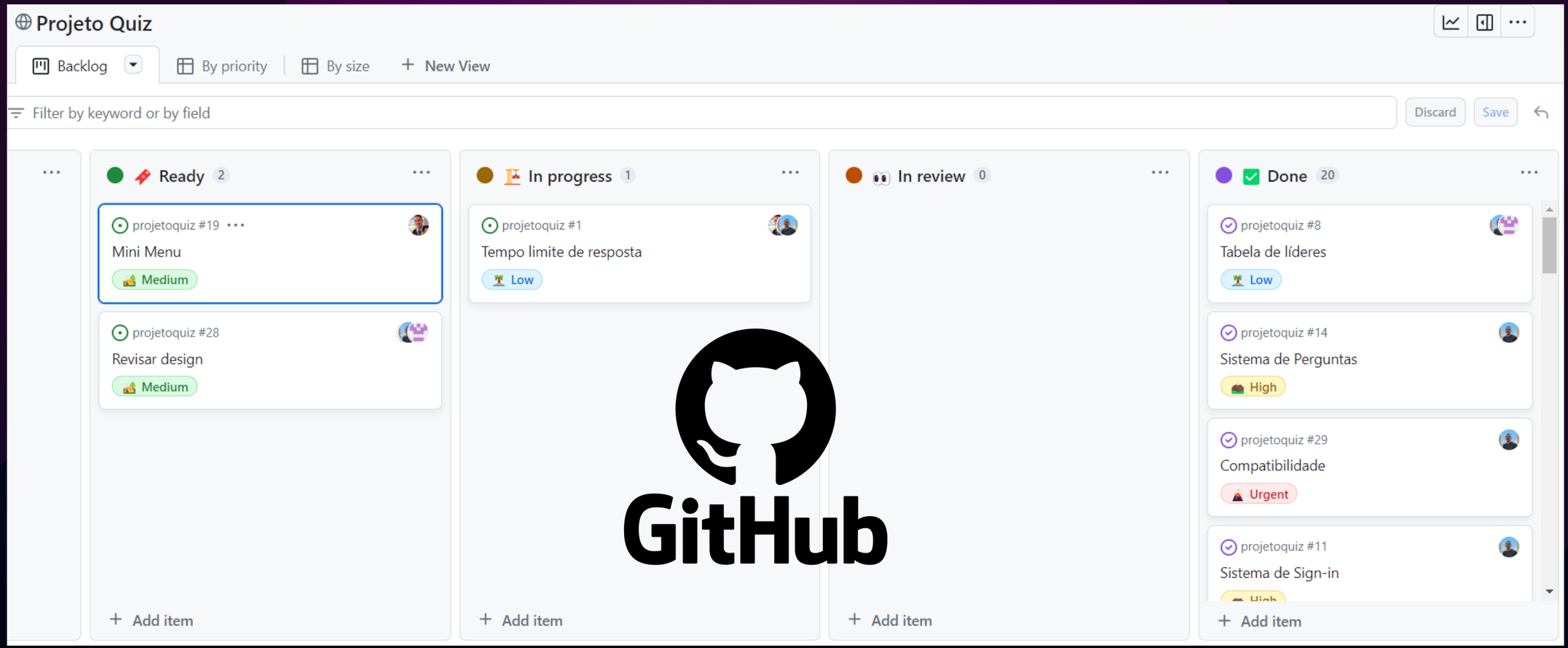
Ready 2

Backlog 0

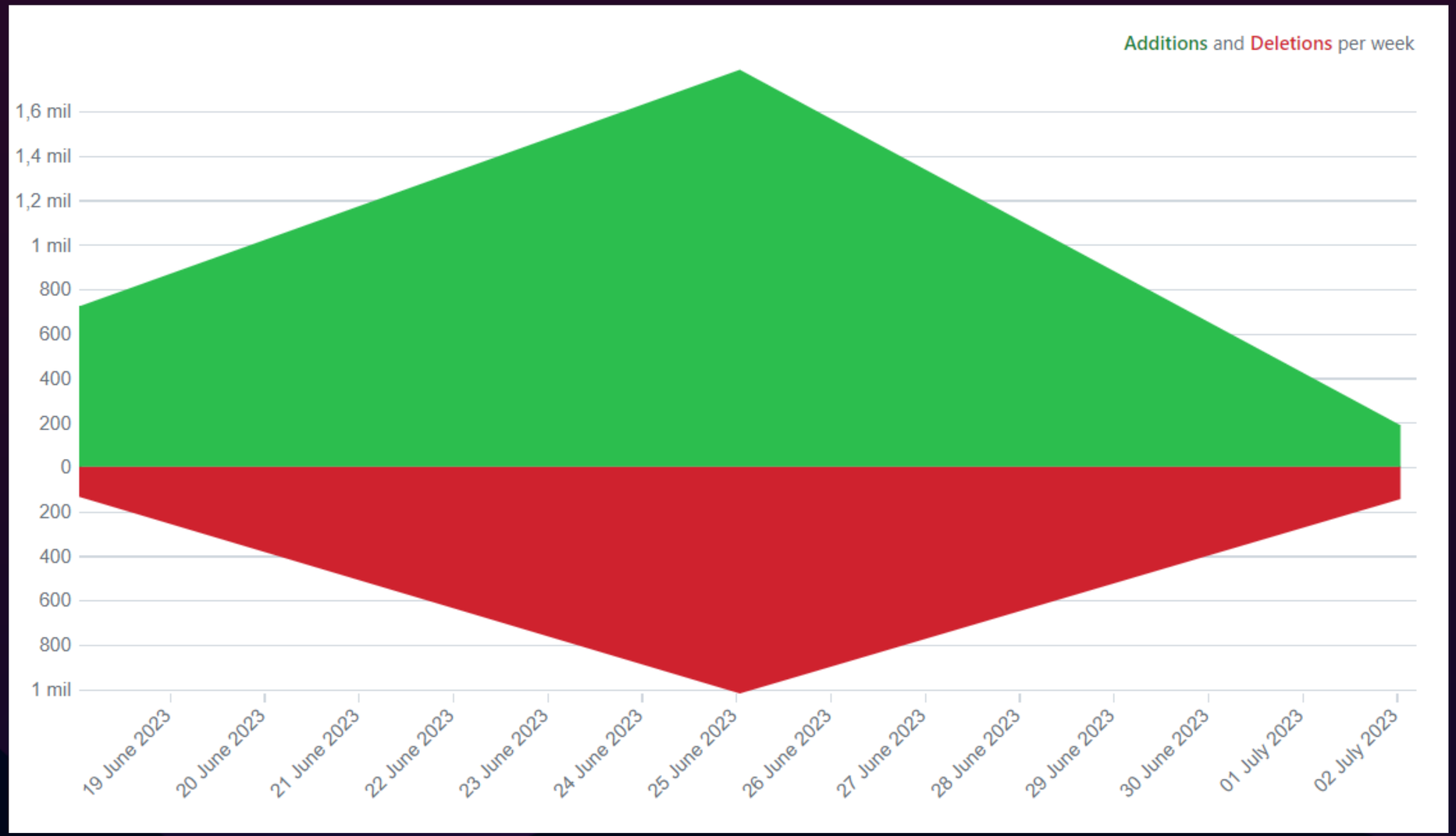
In review 0

In progress 1

Done 20



METODOLOGIA



Code frequency over the history
ofjoseevilasio/projetoquiz

DIFICULDADES

Compatibilidade - Linux vs Windows

Linux™

DIFICULDADES

```cpp
#ifdef _WIN32
    // Inclua as bibliotecas específicas do Windows aqui
    #include <conio.h>
#elif __linux__
    // Inclua as bibliotecas específicas do Linux aqui
    #include <termios.h>
#endif
```

Bibliotecas padrão, no Linux termios.h e no Windows conio.h

DIFICULDADES

```cpp
string path(string nomeArquivo) {
    //Recebe um nome de arquivo que está na pasta assets

    fs::path caminhoAbsoluto = fs::current_path() / "assets" / nomeArquivo;
    fs::path caminhoRelativo = fs::relative(caminhoAbsoluto);

    #ifdef _WIN32
        return caminhoAbsoluto.string();
    #elif defined __unix__
        return caminhoRelativo;
    #endif
}
```

Gerenciamento de caminhos (path) no Linux e no Windows

Biblioteca <filesystem>, disponivel apartir do C++ 17

BANCO DE DADOS

```cpp
struct Usuario {
    // struct para armazenar os dados do usuário
    string nomeCompleto, email, password, pergunta, resposta;
    int pontos;
    };

struct Perguntas {
    // struct para armazenar os dados de perguntas
        string tema, pergunta, respostaA, respostaB, respostaC, respostaD;
        char respostaCorreta;
    };
```

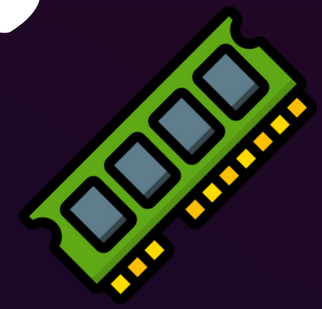
```cpp
//abrir o arquivo de database e salvar as informações coletadas
ofstream arquivo(path("database.txt"), ios::app);

if (arquivo.is_open()) {
    arquivo << usuario.nomeCompleto << "," << usuario.email << "," << usuario.password;
    arquivo << "," << usuario.pergunta << "," << usuario.resposta << "," << usuario.pontos << endl;
    arquivo.close();
    limparTela();
    load();
    cout << corLetra("verde") << "Cadastro realizado com sucesso!!!" << resetCor() << endl;
} else {
    cout << "Falha na gravação do arquivo." << endl;
}
```

>Struct para criar objeto usuário e com ofstream gravar em .txt

BANCO DE DADOS

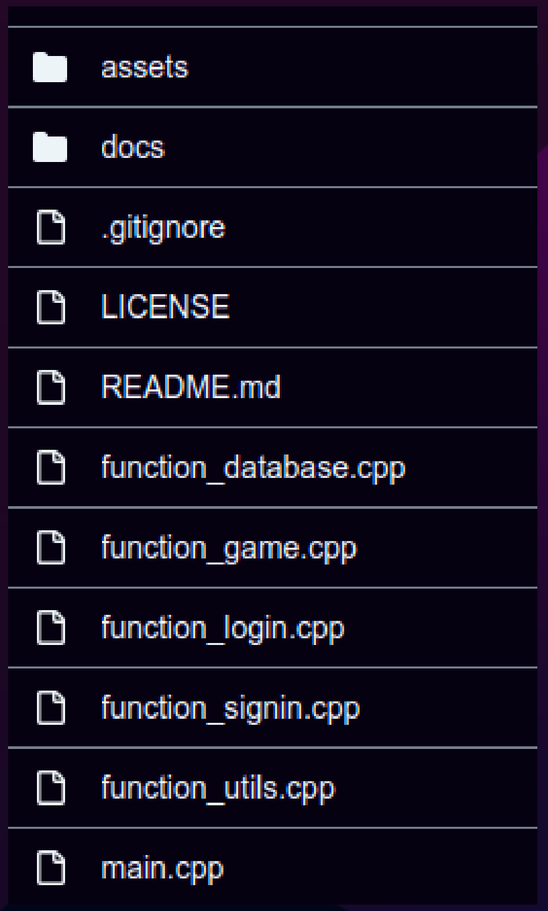

```cpp
struct Usuario {
    // struct para armazenar os dados do usuário
    string nomeCompleto, email, password, pergunta, resposta;
    int pontos;
    };

struct Perguntas {
    // struct para armazenar os dados de perguntas
        string tema, pergunta, respostaA, respostaB, respostaC, respostaD;
        char respostaCorreta;
    };
```

```cpp
vector<Usuario> databaseUsuarios () {
    // Recebe path e retorna um vector com as structs
    Usuario usuario; // instancia da struct
    vector<Usuario> usuarios; //Recebe as structs

    ifstream arquivo(path("database.txt"));

    if (arquivo.is_open()){
        string linha;

        while (getline(arquivo, linha)) {
            istringstream iss(linha);
            string dado;
            vector<string> dados;

            while (getline(iss, dado, ',')) {
                dados.push_back(dado); //A cada volta no loop dentro da linha recebe o
            }

            usuario.nomeCompleto = dados[0];
            usuario.email = dados[1];
            usuario.password = dados[2];
            usuario.pergunta = dados[3];
            usuario.resposta = dados[4];
            usuario.pontos = stoi(dados[5]);

            usuarios.push_back(usuario);
        }

        arquivo.close();

    } else {
        cout << "Falha ao abrir o arquivo." << endl;
    }

    return usuarios;
}
```



>Struct para criar objeto que recebe os dados após leitura do .txt com ifstream para ficar disponível para manipulação.

VISÃO GERAL



> main.cpp, gerenciamento das funções e interface para usuário.

> funções.cpp separados por módulos de funcionalidades que lida com as execuções chamadas pela main.cpp

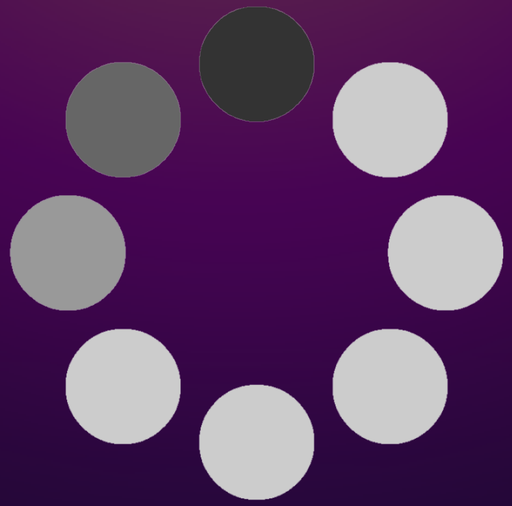
> assets onde fica o banco de dados de usuário e perguntas

> docs onde fica os arquivos formais do projeto

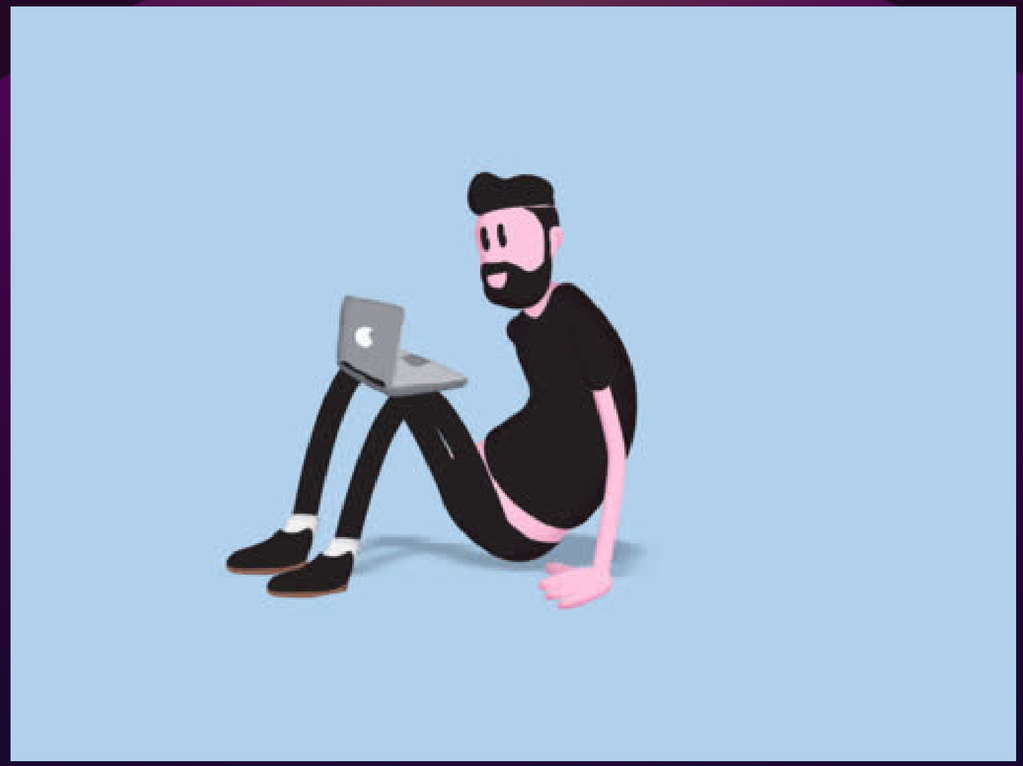
FUNCIONALIDADES

- Efetuar o Registo e Login;
- Recuperar a Senha;
- Jogar;
- Verificar o Ranking;
- Fazer o Reset dos Pontos;
- Ver Regras;
- Eliminar Conta.

CÓDIGO



RODAR O CÓDIGO




DÚVIDAS?
No
No

OBRIGADO

FIM!