

FACULDADE UNIALFA
CURSO BACHARELADO EM ENGENHARIA DE SOFTWARE

ROGÉRIO RODRIGUES ROCHA
THIAGO PINHEIRO LEMES
GUSTAVO MORAIS DA SILVA

PROJETO DE SOFTWARE CRITICALROLL

Goiânia-GO

2024

ROGÉRIO RODRIGUES ROCHA
THIAGO PINHEIRO LEMES
GUSTAVO MORAIS DA SILVA

PROJETO DE SOFTWARE CRITICALROLL

Trabalho apresentado à disciplina Padrões Arquiteturais do Curso Bacharelado de Engenharia de Software da Universidade Unialfa - Campus Goiânia, Perimetral - como requisito parcial para a obtenção de nota.

Orientador: Prof. Me. Yuri Silva Resende

Goiânia-GO
2024

SUMÁRIO

1	INTRODUÇÃO	5
1.1	OBJETIVOS	5
1.1.1	Objetivos Gerais	6
1.1.2	Objetivos Específicos.....	6
2	REQUISITOS.....	7
2.1	REQUISITOS FUNCIONAIS (RF).....	7
2.2	REQUISITOS NÃO FUNCIONAIS (RNF)	9
3	PADRÕES UTILIZADOS.....	11
3.1	PADRÃO ARQUITETURAL MVC.....	11
3.2	PADRÃO DE PROJETO FACADE	11
4	LINGUAGENS DE PROGAMAÇÃO E FERRAMENTAS	12
5	CRONOGRAMA	15
5.1	PLANEJAMENTO E DEFINIÇÃO DO ESCOPO.....	15
5.2	DOCUMENTAÇÃO DO PROJETO	15
5.3	DESENVOLVIMENTO DO PROTÓTIPO INICIAL	15
5.4	DESENVOLVIMENTO E INTEGRAÇÃO DE CONTEÚDO	15
5.5	TESTES E VALIDAÇÃO	16
5.6	AJUSTES E POLIMENTO	16
6	CONSIDERAÇÕES FINAIS	17
	REFERÊNCIAS	18

1 INTRODUÇÃO

Nos últimos anos, o RPG de mesa tem evoluído significativamente, principalmente com o surgimento de plataformas digitais que buscam replicar a experiência imersiva desses jogos. Entre elas, o Roll20 se destaca, oferecendo um "tabuleiro virtual" completo, com inúmeras funcionalidades para mestres e jogadores. No entanto, essa gama de opções pode, por vezes, ser complexa e intimidante, principalmente para iniciantes no mundo dos jogos de RPG de mesa.

Com essa necessidade em mente, o presente projeto busca criar uma solução que seja inspirada em algumas funcionalidades do Roll20, mas que ofereça uma experiência mais simples e acessível para novos jogadores. Diferente do Roll20, que foca em uma simulação de tabuleiro completa, nosso sistema se concentrará em um ambiente onde os jogadores possam gerenciar campanhas, interagir entre si e rolar dados, sem a complexidade de um tabuleiro virtual. O foco será na criação, edição e visualização de personagens, no gerenciamento de campanhas e na comunicação entre jogadores, tornando a plataforma ideal para iniciantes que desejam se familiarizar com o mundo do RPG de mesa.

A simplicidade da interface será um dos pilares do projeto, permitindo que jogadores e mestres possam se concentrar na narrativa e nas interações, sem se perderem em funcionalidades complexas. Além disso, o sistema contará com suporte a regras customizáveis, rolagens de dados, chat integrado, e o gerenciamento completo de fichas de personagens, NPCs e eventos, tudo em um ambiente seguro e intuitivo.

1.1 OBJETIVOS

Os objetivos deste projeto estão orientados para criar uma ferramenta funcional e de fácil utilização, permitindo que mestres e jogadores iniciantes possam conduzir suas campanhas de RPG sem a necessidade de uma curva de aprendizado longa ou um excesso de funcionalidades. Esses objetivos estão divididos em objetivos gerais e específicos, conforme detalhado a seguir.

1.1.1 Objetivo Geral

O objetivo geral é desenvolver uma plataforma de RPG de mesa online, focada em jogadores iniciantes, que facilite a criação e o gerenciamento de campanhas, sem a necessidade de um "tabuleiro virtual". A proposta busca fornecer um ambiente simplificado para a condução de sessões de RPG, com funcionalidades essenciais para o progresso da campanha e interação entre jogadores.

1.1.2 Objetivos Específicos

Os objetivos específicos do projeto incluem:

- Permitir o cadastro e autenticação de senha dos usuários.
- Implementar um sistema de criação e edição de campanhas de RPG.
- Desenvolver uma interface que permita a adição, remoção e gerenciamento de jogadores nas campanhas.
- Oferecer suporte para criação e personalização de fichas de personagens, incluindo cálculo automático de atributos.
- Implementar funcionalidades de chat para comunicação entre jogadores, com suporte a mensagens privadas e em grupo.
- Criar um sistema de rolagem de dados, com suporte para diferentes tipos de dados (d4, d6, d8, etc.).
- Desenvolver uma ferramenta que permita o mestre criar e gerenciar a narrativa da campanha, incluindo NPCs e eventos.
- Proteger as informações dos usuários e garantir a integridade dos dados por meio de criptografia.

2 REQUISITOS

2.1 REQUISITOS FUNCIONAIS (RF)

[RF01] Cadastro e autenticação de usuários:

Descrição: O sistema deverá permitir que os usuários realizem o cadastro, login e logout. As informações de cadastro incluirão nome de usuário e senha. Após o login, o sistema deve permitir a autenticação segura do usuário para acessar as funcionalidades do aplicativo.

[RF02] Criar e editar campanhas:

Descrição: O sistema permitirá que os mestres de jogo possam criar e editar campanhas de RPG. Durante a criação, os mestres poderão definir o nome, descrição e sistema de regras da campanha. As campanhas criadas deverão ficar salvas e acessíveis ao mestre e aos jogadores.

[RF03] Adicionar e remover jogadores de uma campanha:

Descrição: O sistema permitirá que o mestre de jogo possa adicionar ou remover jogadores de uma campanha existente. Os jogadores adicionados terão acesso à campanha e às suas respectivas funcionalidades, como rolagem de dados e visualização de fichas de personagens.

[RF04] Configurar regras da campanha:

Descrição: O mestre poderá configurar regras personalizadas para cada campanha, como sistemas de regras pré-definidos ou customizáveis, atributos de personagens e dados utilizados. Isso inclui a definição de sistemas de fichas e o cálculo automático de atributos com base nas regras estabelecidas.

[RF05] Interface para visualização de personagens:

Descrição: O sistema disponibilizará uma interface onde os jogadores poderão visualizar as fichas de seus personagens, incluindo informações como atributos, habilidades, equipamentos e status. O mestre também terá acesso a essas fichas para consulta e edição.

[RF06] Implementação de rolagem de dados:

Descrição: O sistema deverá permitir a rolagem de dados virtuais, suportando diferentes tipos como d4, d6, d8, d10, d12 e d20. O resultado será exibido para todos os jogadores da campanha, junto ao histórico de rolagens.

[RF07] Chat de texto para comunicação entre jogadores:

Descrição: O sistema incluirá um chat integrado para que os jogadores e o mestre possam se comunicar durante as sessões de RPG. O chat deverá suportar mensagens em grupo, assim como privadas.

[RF08] Suporte para envio de mensagens privadas e em grupo:

Descrição: O chat do sistema permitirá o envio de mensagens privadas entre jogadores, além das mensagens em grupo. Esta funcionalidade facilita a comunicação específica entre indivíduos sem interferir na conversa geral.

[RF09] Criação, edição e visualização de fichas de personagens:

Descrição: O sistema deverá permitir que jogadores criem, editem e visualizem fichas de seus personagens. Essas fichas conterão informações como atributos, habilidades, equipamentos e histórico do personagem, integradas ao sistema de regras.

[RF10] Integração com o sistema de regras para cálculo automático:

Descrição: As fichas de personagens deverão estar integradas ao sistema de regras da campanha, possibilitando o cálculo automático de atributos como força, destreza e pontos de vida, com base nas definições da campanha.

[RF11] Upload e compartilhamento de imagens e documentos:

Descrição: O sistema permitirá o upload e o compartilhamento de imagens e documentos, tanto pelo mestre quanto pelos jogadores. Esse recurso é útil para o envio de mapas, fichas ou outros arquivos necessários para o andamento da campanha.

[RF12] Ferramenta para o mestre criar e gerenciar a história, NPCs e eventos:

Descrição: O sistema deverá incluir uma ferramenta que permita ao mestre criar e gerenciar a história da campanha, incluindo a criação de NPCs (personagens não jogáveis) e o planejamento de eventos importantes que ocorrerão durante o jogo.

2.2 REQUISITOS NÃO FUNCIONAIS (RNF)

[RNF01] Suporte a múltiplas campanhas simultâneas:

Descrição: O sistema deverá ser capaz de suportar a execução de múltiplas campanhas ao mesmo tempo, com diferentes mestres e jogadores, sem comprometer a performance ou a experiência de jogo.

[RNF02] Tempo de resposta rápido:

Descrição: O sistema deverá ter um tempo de resposta rápido, especialmente nas interações críticas como rolagens de dados e atualizações de fichas de personagens, garantindo uma experiência de usuário fluida.

[RNF03] Interface amigável e intuitiva:

Descrição: A interface deverá ser projetada de forma a ser fácil de usar e entender, com uma curva de aprendizado reduzida, ideal para iniciantes no RPG de mesa e jogadores com pouca experiência em sistemas digitais.

[RNF04] Proteção dos dados dos usuários:

Descrição: Todos os dados dos usuários, incluindo informações pessoais e fichas de personagens, deverão ser protegidos por criptografia, garantindo a privacidade e a segurança das informações.

[RNF05] Controle de acesso:

Descrição: O sistema deverá implementar mecanismos de controle de acesso para garantir que apenas usuários autorizados possam editar ou modificar campanhas e personagens, evitando alterações não permitidas.

[RNF06] Acessibilidade em múltiplas plataformas:

Descrição: O sistema deverá ser acessível via navegador web em diferentes dispositivos, como desktops, tablets e smartphones, oferecendo uma experiência consistente e responsiva em todos os tamanhos de tela.

[RNF07] Código bem documentado e modular:

Descrição: O código do sistema deverá ser escrito de maneira modular e bem documentada, facilitando futuras manutenções, expansões e adaptações do sistema, seja para novas funcionalidades ou para outras plataformas.

[RNF08] Alta disponibilidade:

Descrição: O sistema deverá ter alta disponibilidade, garantindo que os jogadores possam acessar suas campanhas e continuar o jogo sem interrupções ou quedas no serviço.

[RNF09] Portabilidade para outras plataformas:

Descrição: O sistema deverá ser desenvolvido de forma a permitir sua portabilidade futura para outras plataformas, como aplicativos móveis nativos, sem grandes modificações na estrutura do código.

3 PADRÕES UTILIZADOS

3.1 PADRÃO ARQUITETURAL MVC

O Model-View-Controller (MVC) é um padrão arquitetural amplamente utilizado no desenvolvimento de aplicações, especialmente naquelas que envolvem interfaces gráficas e web. Esse padrão divide a aplicação em três componentes principais: o Modelo, responsável pela lógica de dados e regras de negócios; a Vista, que lida com a interface do usuário e a apresentação das informações; e o Controlador, que faz a intermediação entre o Modelo e a Vista, recebendo entradas do usuário e processando as ações necessárias. A ideia principal por trás do MVC é a separação de responsabilidades, permitindo que cada uma dessas partes seja desenvolvida, modificada e mantida de maneira independente, o que facilita a escalabilidade e a manutenção do sistema.

O principal objetivo do MVC é proporcionar uma arquitetura modular e organizada, permitindo maior flexibilidade na implementação e manutenção de aplicações complexas. Ao manter a lógica de apresentação separada da lógica de negócios e das interações do usuário, o padrão MVC facilita o desenvolvimento de sistemas com diferentes camadas, melhora a reutilização de código e reduz o acoplamento entre os componentes. Essa abordagem resulta em um sistema mais sustentável a longo prazo, onde mudanças na interface, por exemplo, podem ser feitas sem a necessidade de alterar o núcleo do sistema ou sua lógica de funcionamento.

3.2 PADRÃO DE PROJETO FACADE

O padrão de projeto Facade (Fachada) é um dos padrões estruturais definidos pela Gang of Four (GoF) e tem como principal objetivo fornecer uma interface simplificada para um conjunto complexo de classes ou subsistemas. Ao utilizar esse padrão, um único ponto de acesso é oferecido para a interação com várias partes do sistema, ocultando a complexidade interna das classes que compõem o subsistema. A Fachada serve como uma "capa" para o sistema, facilitando a comunicação e interação entre módulos de alto nível e componentes internos, além de reduzir o número de interações diretas que um cliente precisa ter com os componentes internos do sistema.

O objetivo principal do padrão Facade é simplificar a utilização de subsistemas complexos e aumentar a manutenibilidade do código. Ele promove o encapsulamento de complexidades, tornando mais fácil para o cliente interagir com o sistema sem precisar compreender toda a sua estrutura interna. Além disso, o uso da Facade contribui para reduzir o acoplamento entre os componentes do sistema, uma vez que as alterações nos subsistemas internos podem ser feitas sem impactar diretamente o cliente, desde que a interface da Fachada permaneça a mesma.

4 LINGUAGENS DE PROGRAMAÇÃO E FERRAMENTAS

Para o desenvolvimento das atividades do projeto elaborado serão utilizadas as seguintes linguagens de programação e ferramentas de desenvolvimento:

- **Javascript:** JavaScript é uma linguagem de programação amplamente utilizada para o desenvolvimento web. É uma linguagem de alto nível e interpretada, que permite a criação de interações dinâmicas e responsivas no lado do cliente. No contexto deste projeto, JavaScript será fundamental para implementar funcionalidades interativas na interface do usuário e gerenciar a comunicação entre o frontend e o backend. Além disso, com o uso de bibliotecas e frameworks como React, a linguagem facilitará a criação de uma interface modular e eficiente.
- **HTML:** HTML é a linguagem padrão de marcação utilizada para estruturar e exibir conteúdo na web. Cada página e interface do sistema será desenvolvida com HTML, que fornece a estrutura básica para a exibição de informações e elementos de interação. O HTML permite organizar o conteúdo de maneira hierárquica e semântica, garantindo que os navegadores e dispositivos possam renderizar as páginas corretamente e que o sistema seja acessível e navegável.
- **CSS3:** CSS3 é a versão mais recente da linguagem de folhas de estilo em cascata, usada para estilizar e dar um layout visual ao HTML. No projeto, o CSS3 será utilizado para definir a aparência do sistema, incluindo cores, fontes, espaçamentos e efeitos visuais, garantindo que a interface seja amigável e visualmente atraente. Com CSS3, é possível implementar um design responsivo, adaptando a interface a diferentes tamanhos de tela e dispositivos, como desktops, tablets e smartphones.
- **Banco de Dados PostgreSQL:** PostgreSQL é um sistema de gerenciamento de banco de dados relacional de código aberto, reconhecido por sua robustez, confiabilidade e conformidade com o padrão SQL. No projeto, o PostgreSQL será utilizado para armazenar e

gerenciar dados críticos do sistema, como informações de usuários, campanhas, histórico de rolagens e arquivos compartilhados. A escolha do PostgreSQL permite realizar consultas complexas e garantir a integridade dos dados, além de oferecer escalabilidade para suportar múltiplas campanhas e vários jogadores simultâneos.

- **Node.js:** Node.js é uma plataforma de execução JavaScript no lado do servidor, amplamente utilizada para criar aplicações rápidas e escaláveis. Baseado no mecanismo V8 do Google Chrome, o Node.js permite lidar com operações assíncronas e de alta concorrência, tornando-o ideal para sistemas que precisam gerenciar múltiplas conexões simultaneamente, como um chat ou sistema de campanhas online. No projeto, o Node.js será essencial para implementar a lógica do backend, gerenciar solicitações HTTP e conectar o sistema ao banco de dados PostgreSQL.

5 CRONOGRAMA

O cronograma previsto para o desenvolvimento deste trabalho foi definido nas seguintes fases:

5.1 PLANEJAMENTO E DEFINIÇÃO DO ESCOPO

A fase de Planejamento e Definição do Escopo inclui as atividades de:

- Definição dos objetivos e escopo do site.
- Planejamento da aparência geral.
- Reuniões de brainstorming para definição de tópicos e mecânicas.

A data de início para a realização da atividade é dia 18/09/2024.

A data de termino da realização da atividade é dia 22/09/2024.

5.2 DOCUMENTAÇÃO DO PROJETO

A fase de Documentação do Projeto inclui as atividades de:

- Redação da introdução, objetivos e justificativa.
- Desenvolvimento da metodologia e cronograma.
- Revisão e formatação conforme normas ABNT.

A data de início para a realização da atividade é dia 28/09/2024.

A data de termino da realização da atividade é dia 04/10/2024.

5.3 DESENVOLVIMENTO DO PROTÓTIPO INICIAL

A fase de Desenvolvimento do Protótipo Inicial inclui as atividades de:

- Implementação do sistema básico de Rolagem e mostra de imagens.
- Criação de um cenário e tela inicial.
- Testes iniciais de utilização, mecânicas e aplicações.

A data de início para a realização da atividade é dia 18/09/2024.

A data de termino da realização da atividade é dia 22/09/2024.

5.4 DESENVOLVIMENTO E INTEGRAÇÃO DE CONTEÚDO

A fase de Desenvolvimento e Integração de Conteúdo inclui as atividades de:

- Criação dos sistemas de rolagem avançados, inclusão de sistema de criação de tokens e demonstração de personagens em layers na tela.

- Desenvolvimento de FrontEnd e telas de controle de mapa, fichas e sistemas.
- Implementação de sistemas e chat com histórico.
- Inserção de Sistemas de RPG para criação automática.

A data de início para a realização da atividade é dia 05/10/2024.

A data de termino da realização da atividade é dia 25/10/2024.

5.5 TESTES E VALIDAÇÃO

A fase de Testes e Validação inclui as atividades de:

- Testes internos com a equipe e conhecidos.
- Correção de bugs e ajustes.
- Testes beta com grupos de usuários pela internet - Coleta e análise de feedback.

A data de início para a realização da atividade é dia 26/10/2024.

A data de termino da realização da atividade é dia 05/11/2024.

5.6 AJUSTES E POLIMENTO

A fase de Ajustes e Polimento inclui as atividades de:

- Refinamento de gráficos e animações.
- Implementação de melhorias na interface de usuário (UI).
- Correção final de bugs.

A data de início para a realização da atividade é dia 06/11/2024.

A data de termino da realização da atividade é dia 16/11/2024.

6 CONSIDERAÇÕES FINAIS

O desenvolvimento deste sistema de RPG de mesa online, inspirado na plataforma Roll20, teve como principal objetivo criar uma solução simplificada e acessível, voltada para iniciantes no universo dos jogos de RPG. Com foco na facilidade de uso e na implementação de funcionalidades essenciais para o gerenciamento de campanhas e personagens, o projeto visa suprir a necessidade de uma plataforma menos complexa, sem a exigência de um "tabuleiro virtual" robusto, mas que ainda ofereça as ferramentas necessárias para mestres e jogadores interagirem de maneira fluida e imersiva.

Durante o processo de desenvolvimento, foram enfrentados desafios técnicos e estruturais relacionados à criação de uma interface intuitiva, capaz de suportar múltiplos usuários e campanhas simultâneas. No entanto, o uso de padrões arquiteturais como o MVC e o padrão de projeto Facade permitiu modularizar o sistema, facilitando sua manutenção e escalabilidade. Acreditamos que a solução proposta atenderá de forma eficiente às necessidades dos jogadores e mestres iniciantes, proporcionando uma plataforma estável, acessível e segura. Para o futuro, esperamos expandir as funcionalidades, considerando a implementação de novos recursos e a portabilidade para outras plataformas, garantindo a evolução contínua do sistema.

REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **ABNT NBR 6023:** informação e documentação: referências: elaboração. Rio de Janeiro: ABNT, 2018.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **ABNT NBR 6024:** informação e documentação: numeração progressiva das seções de um documento: apresentação. Rio de Janeiro: ABNT, 2012.

<https://roll20.net/>

https://bdm.unb.br/bitstream/10483/22294/1/2018_BrunoFreirePereira_tcc.pdf

<https://www.unialfa.com.br/arqs/mestradoadm/documentos/22.pdf>

<https://www.devmedia.com.br/introducao-ao-padrao-mvc/29308>

<https://refactoring.guru/pt-br/design-patterns/facade>

<https://www.devmedia.com.br/o-padrao-facade-aplicado/12683>