

PAIMON: RECOMMENDATION SYSTEM AND PRIVATE LIBRARY

PAIMON: SISTEMA DE RECOMENDAÇÕES E BIBLIOTECA PARTICULAR

Thyago Vinicius Camilo Cunha
Universidade Federal do Paraná

Gabriel Schmidt Wojciechowski
Universidade Federal do Paraná

Nickolas Ukasinski
Universidade Federal do Paraná

Rafaela Mantovani Fontana ; <https://orcid.org/0000-0001-6350-4167>
Universidade Federal do Paraná

**PAIMON:
SISTEMA DE RECOMENDAÇÕES E BIBLIOTECA PARTICULAR**

**PAIMON:
RECOMMENDATION SYSTEM AND PRIVATE LIBRARY**

*Gabriel Schmidt Wojciechowski
082.810.819-61*

*Setor de Educação Profissional e Tecnológica – UFPR Curitiba/PR
gabrielwo314@gmail.com*

*Nickolas Ukasinski
062.731.169-54*

*Setor de Educação Profissional e Tecnológica – UFPR Curitiba/PR
nickolasukasinski@ufpr.br*

*Thyago Vinicius Camilo Cunha
115.041.139-26*

*Setor de Educação Profissional e Tecnológica – UFPR Curitiba/PR
thyago.vccunha@gmail.com*

*Orientador: Professora Doutora Rafaela Mantovani Fontana
0000-0001-6350-4167*

034.217.689-78

*Setor de Educação Profissional e Tecnológica – UFPR Curitiba/PR
rafaela.fontana@ufpr.br*

RESUMO: A pandemia do coronavírus (COVID-19) fez com que a demanda de meios de entretenimento aumentasse rapidamente, devido ao período em que a população mundial estava em isolamento social. Ademais, com a grande quantidade de opções na atual era da informação, torna-se complicado encontrar os materiais de entretenimento adequados e desejados. Com base nos fatos apontados, pode-se observar a necessidade de um sistema eficiente de organização pessoal de diversas categorias de obras: jogos, livros, filmes e séries; juntamente com recomendações baseadas em interesses pessoais para auxiliar na escolha de entretenimento a ser consumida pelo usuário. Portanto, analisando o perfil de internautas do país e o dispositivo mais utilizado para navegação, foi definido como o aparelho alvo para desenvolvimento o smartphone. Então este trabalho propõe a criação de um aplicativo móvel que conta com a biblioteca pessoal do usuário e recomendações de obras de interesse individual. A análise e modelagem foram feitas utilizando diagramas baseados na linguagem UML. Durante a fase de desenvolvimento do projeto, a equipe utilizou o framework Flutter para a criação do aplicativo, com suporte para sistemas Android e iOS, de interfaces e realização de requisições ao back-end; e o framework Spring Boot para comunicação entre o app e o banco de dados, utilizando API RESTful. O planejamento e a execução foram realizados considerando o uso de ideologias da metodologia ágil que foram modificadas pelo grupo como o Scrum para a divisão do progresso do projeto em sprints e o quadro Kanban como método de organização de tarefas.

ABSTRACT: The coronavirus pandemic (COVID-19) made so the demand for entertainment media increased rapidly, considering the period the world population stayed in social isolation. In addition, with the great number of options in the current information era, it becomes complicated to find the most suitable and intended entertainment sources for each person. Based on the mentioned facts, it is possible to find the need for an efficient software in charge of the personal organization and management of different categories of works: games, books, movies and series; along with different recommendations based on the personal interests of the respective user to assist in choosing different entertainment forms to be consumed by the end user. Therefore, analyzing the profiles of distinct internet users in the country and their most used device for navigation, it was defined the target device for the development would be the smartphone. Accordingly, this project proposes the creation of a mobile application that accounts for a personal library for the current user and the recommendation of works of individual interest. The analysis and modeling were made by using different diagrams based on UML. During the development phase of the project, the team used the Flutter framework for the creation of the app, with support for Android and iOS systems, with interfaces and execution of requests for the back-end; and the Spring Boot framework for the communication between the app and the database, with a RESTful API. The overall planning and execution were performed considering the use of Agile methodologies that were modified by the team with Scrum for the division of the progress of the project in sprints, and a Kanban board as the method for organizing the group's tasks.

PALAVRAS-CHAVE: Gestão pessoal. Entretenimento. Obras. Recomendação. Aplicativo Móvel.

KEYWORD: Personal Management. Entertainment. Works. Recommendation. Mobile Application.

1 INTRODUÇÃO

A tecnologia da informação mudou a maneira como toda e qualquer forma de mídia é reproduzida (FARIAS, 2020), e com a pandemia do COVID-19, juntamente ao isolamento social, a população encontrou no entretenimento digital uma forma de lazer, em especial, no serviço de *streaming*. Um estudo realizado durante a pandemia, com uma amostra de 1260 pessoas, aponta que, dentre os brasileiros entrevistados, 42,8% assistiam diariamente a, pelo menos, um tipo de conteúdo de streaming, e outros 43,9% tinham esse hábito ao menos uma vez por semana (SEGS.COM, 2020).

Além do streaming, houve a busca por outras categorias de entretenimento, tal como o setor de videogames, no qual o game “*World of Warcraft*” bateu o recorde, em 2020, como jogo mais vendido em um período de 24h (TECHMUNDO.COM, 2020). Similarmente, em 2021 o setor de varejo de livros no Brasil apresentou um crescimento de 38,38% em vendas e compras de livros, em relação ao ano anterior. Ademais, constatou-se um aumento de 28,46% no faturamento das empresas do setor (FALAUNIVERSIDADES.COM, 2021). Os

dados ainda apontam que, durante o período de isolamento, o brasileiro voltou a se interessar pela literatura.

Neste contexto de demanda crescente do consumo de entretenimento, identificou-se que existem sites como o Letterboxd (LETTERBOXD, 2022), com detalhes e avaliações de filmes, e o MyAnimeList (MYANIMELIST.COM, 2022), focado em quadrinhos e desenhos japoneses, mas nenhum deles permite a organização pessoal e a recomendação de diversos tipos de mídia de entretenimento na mesma plataforma. Entende-se, assim, que uma plataforma unificada de gestão pessoal de interesses em entretenimento pode ser útil ao público alvo, dado o crescimento da demanda e a quantidade de opções que existem na atualidade.

1.1 OBJETIVO

Este trabalho teve como objetivo desenvolver um aplicativo para dispositivos móveis de gerenciamento e recomendação de obras de entretenimento (séries, livros, filmes e jogos).

1.1.1 OBJETIVOS ESPECÍFICOS

1. Prover a centralização dos interesses do usuário em diferentes tipos de mídia em um único aplicativo.
2. Fornecer ao usuário recomendações de obras de entretenimento.
3. Permitir a consulta de detalhes sobre uma obra de entretenimento.
4. Permitir a gestão das obras em um ambiente web para administrador.

2 METODOLOGIA

“O software distribui o produto mais importante de nossa era – a informação” (PRESSMAN e MAXIN, 2016, p 3). Para a distribuição do bem mais precioso da atualidade, são necessários métodos e processos de engenharia de modo a atingir a qualidade de software. Com esta finalidade, foram utilizadas duas diferentes metodologias de desenvolvimento Ágil: *Scrum* e *Kanban*.

Descrito por Pressman e Maxin (2016) o *Scrum* é um método de desenvolvimento ágil que possui as atividades de requisitos, análise, projeto, evolução e entrega, realizadas dentro de um período chamado *sprint*. Cada *sprint* é marcada pelo *backlog* da *sprint*, lista

de funcionalidades que devem ser entregues naquele período. A realização de reuniões diárias também é uma característica básica do *Scrum*, nas quais se debatem atividades feitas pela equipe, problemas enfrentados e o planejamento até a próxima reunião (PRESSMAN; MAXIN, 2016).

O *Kanban* é uma metodologia de gerenciamento de projetos cuja principal característica é o quadro *kanban*, instrumento visual gerado pelos conceitos de valor e fluxo dentro dos processos. O quadro permite a visualização do trabalho como um todo e em qual parte do processo cada item da divisão do trabalho se encontra. Dessa forma, ele permite a execução das tarefas de uma forma autogerenciável (PRIKLADNICKI; WILLI; MILANI, 2014, p. 119-136).

Utilizando as ferramentas apresentadas, este projeto foi desenvolvido em *sprints* conduzidas de fevereiro de 2022 a setembro de 2022. O projeto foi dividido em duas etapas. Na primeira etapa, as *sprints* tinham como objetivo a produção de elementos de análise e arquitetura do sistema. Na segunda etapa, um conjunto de histórias de usuário foi finalizado e integrado à aplicação ao final de cada ciclo, conforme descrito a seguir.

3 DESENVOLVIMENTO

Para a primeira etapa, foram concebidos os modelos *UML (Unified Modeling Language)* e as histórias de usuário que representam o escopo do software. Além disso, nesta etapa, foram feitas avaliações de diferentes tipos de algoritmos de recomendação e agrupamento, e ferramentas a serem utilizadas para o desenvolvimento do sistema.

Na segunda etapa, foi realizada a codificação do software de acordo com as definições da etapa anterior. O uso do *Scrum* intensificou-se, já que foi feito de fato o desenvolvimento com base nas histórias de usuário.

A divisão das *Sprints* da segunda etapa, com seus respectivos trabalhos, se deu da seguinte forma:

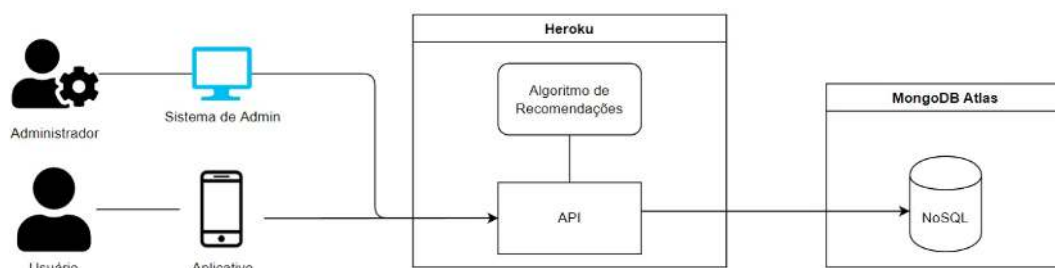
- Sprint 1: Tela principal do usuário, tela de obra e tela de pesquisa de obras.
- Sprint 2: Telas de perfil e lista pessoal, junto de seus respectivos filtros.
- Sprint 3: Tela de edição do perfil e a adição/edição das informações de uma obra dentro da lista do usuário.
- Sprint 4: Desenvolvimento do sistema de administradores, incluindo a base do sistema, tela de login, e as listas de obras e de pessoas, com pesquisa e exclusão.

- Sprint 5: Telas de criação e edição de obras e pessoas para o administrador e visualização de pessoas em uma obra para o aplicativo.
- Sprint 6: Correção de bugs no sistema, revisão de layout, finalização da documentação e preparação de dados para uso no banco de dados.

4 RESULTADOS OBTIDOS

A arquitetura do sistema resultante pode ser resumida em três componentes: um *back-end* que entrega respostas em *JSON*; um aplicativo para dispositivos móveis, que permite o acesso do usuário; e um cliente *web* para o administrador do sistema gerenciar as obras cadastradas, conforme mostra a Figura 1. O usuário acessa o aplicativo localmente no seu celular, que acessa a *API* hospedada em um servidor pelo sistema *Heroku*, possibilitando a execução do algoritmo de recomendações, e ao acesso ao banco de dados *NoSQL*, também armazenado em um servidor. O banco de dados está armazenado no sistema *MongoDB Atlas*.

FIGURA 1: Visão geral da arquitetura do sistema



FONTE: Os autores

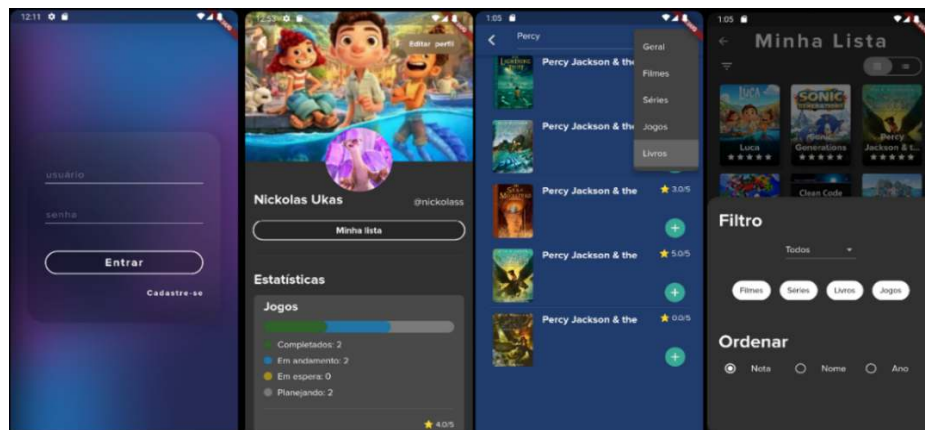
4.1 APLICATIVO PARA DISPOSITIVOS MÓVEIS

O aplicativo para dispositivos móveis é a parte central do acesso do usuário. É para ele que o desenvolvimento das APIs foi direcionado, e é através dele que pode ser visualizado o cumprimento dos objetivos. Ele foi desenvolvido utilizando a linguagem de programação *Dart* em conjunto com o *framework* para desenvolvimento de interfaces *Flutter*.

As principais funcionalidades oferecidas pelo aplicativo são: *dashboard* contendo diversas listas de obras, separadas por *tags* (etiquetas que demarcam características da obra), além de uma lista contendo recomendações para o usuário; sistema de *login* seguro utilizando *JWT*; lista pessoal de obras com o progresso do usuário para cada uma; e

personalização do perfil do usuário, com estatísticas das obras da lista. Uma amostra das telas do aplicativo pode ser visualizada na Figura 2.

FIGURA 2: Exemplos de telas presentes no aplicativo



FONTE: Os autores

4.2 CLIENTE WEB PARA ADMINISTRADOR DO SISTEMA

Além do aplicativo, foi desenvolvido um cliente web para que o administrador do sistema possa cadastrar, editar e remover obras de entretenimento e pessoas que participaram dessas obras, seja em qualquer papel, como ator, escritor, diretor, roteirista etc. Esse cliente foi desenvolvido utilizando *Flutter Web*.

Esse módulo conta com os recursos básicos de um sistema padrão de *CRUD* – *Create, Read, Update & Delete*, modelado para facilitar o gerenciamento por parte do administrador. Além disso, esse módulo também conta com uma validação de usuário, utilizando *JWT* e definindo uma permissão específica necessária para acesso, que é diferente da permissão utilizada para entrar no aplicativo.

4.3 BACK-END E SISTEMA DE RECOMENDAÇÃO

Para prover os dados a ambos os clientes, foi criado um servidor *back-end*, utilizando a linguagem *Java* e o framework *Spring Boot*, contando também com um algoritmo de recomendação de obras. Para armazenamento dos dados, foi utilizando o banco de dados não-relacional *MongoDB*, provido através do serviço *MongoDB Atlas*.

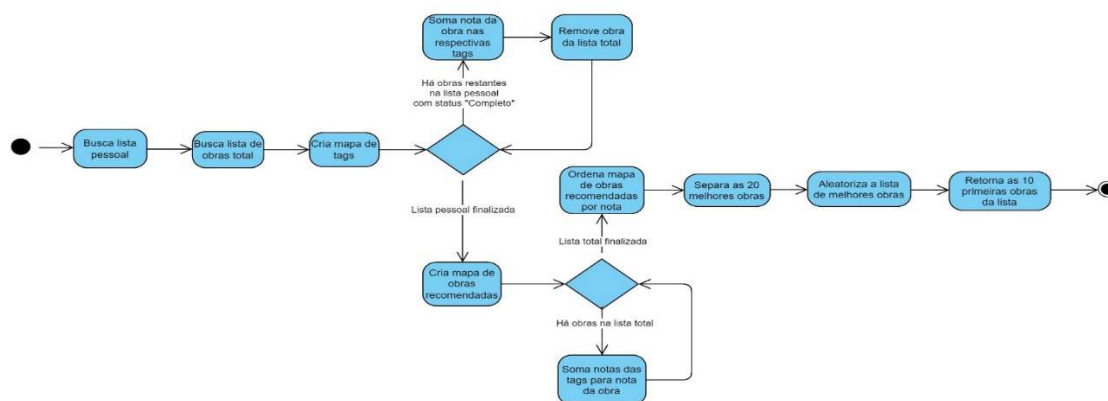
Para a criação do algoritmo de recomendação, partiu-se principalmente do conceito de recomendação baseada em conteúdo (METEREN, 2000). Conceitualmente, a filtragem

baseada em conteúdo se baseia no relacionamento entre itens específicos e um usuário, baseando-se nas suas escolhas prévias e no seu feedback sobre tais itens.

A Figura 3 mostra como o algoritmo cria as recomendações de obras. O algoritmo inicia a busca da lista pessoal de obras do usuário e da lista de todas as obras do sistema, criando um mapa de notas para cada *tag* no sistema, inicializadas com 0. Após isso, itera por toda a lista pessoal, buscando obras com o status “Completo” e uma nota atribuída, somando a nota dada a obra pelo usuário a cada *tag* dentro de tal obra. No fim da iteração, o algoritmo possui um mapa das *tags* favoritas do usuário, mantendo uma nota para cada.

Após isso, é iterada a lista de todas as obras no sistema, ignorando as já presentes na lista do usuário. Cria-se num novo mapa denominando uma nota para cada obra, identificada como a soma da nota de cada *tag* dentro da obra, sendo tal nota guardada no mapa de *tags* criado anteriormente. Por fim, o mapa de recomendações é ordenado da maior nota para a menor, e são separadas as 20 obras com maior nota. Essa lista é aleatorizada, e se retorna os 10 primeiros elementos dessa lista como obras recomendadas ao usuário. Tal método de ordenação aleatória existe para criar uma forma de variação às obras recomendadas, diversificando o resultado para o usuário sempre que o aplicativo for acessado, entregando assim uma melhor experiência.

FIGURA 3: Fluxograma do funcionamento do algoritmo de recomendação



FONTE: Os autores

5 CONSIDERAÇÕES FINAIS

Este trabalho apresentou um aplicativo para dispositivos móveis para o gerenciamento e recomendação de obras de entretenimento (séries, livros, filmes e jogos). Implementou-se um algoritmo de recomendação baseado em conteúdo capaz de recomendar obras que o usuário potencialmente terá interesse.

Em seu estado final, este trabalho ainda possui algumas limitações: a inserção de obras no banco de dados ainda não ocorre de forma automática, e o aplicativo não permite interação entre os usuários, na forma de rede social. Entende-se que isto seria interessante, como um futuro trabalho, oferecendo-se a possibilidade de listas públicas e listas específicas compartilhadas entre usuários, similares a sistemas de *playlists* de músicas.

REFERÊNCIAS

COSTA, T. S. WoW: Shadowlands bate o recorde de jogo mais vendido no PC em 24h. *In: Tecmundo. Voxel. 9 dez. 2020. Disponível em: <https://www.tecmundo.com.br/voxel/207961-wow-shadowlands-bate-recorde-jogo-vendido-pc-24h.htm>. Acesso em: 12 mar. 2022.*

FARIA, N. H. S. **A indústria do entretenimento e a expansão do streaming em meio a pandemia.** 2020. 34 f. Trabalho de Conclusão de Curso (Graduação em Comunicação Social com Habilitação em Jornalismo). Centro Universitário Do Sul De Minas, Varginha, 2020. Disponível em: <http://repositorio.unis.edu.br/handle/prefix/1371>. Acesso em: 22 fev. 2022.

LETTERBOXD. **Letterboxd.** Disponível em: <https://letterboxd.com/>. Acesso em: 10 fev. 2022.

METEREN, Robin van; SOMEREN, Maarten van. Using content-based filtering for recommendation. **Proceedings of the machine learning in the new information age: MLnet/ECML2000 workshop**, v. 30. 2000.

MICHELOTTO, G. Consumo e produção de livros aumentam durante a pandemia. *In: Fala! UFPR. Fala! Universidades.* Curitiba, 15 set. 2021. Disponível em: <https://falauniversidades.com.br/consumo-e-producao-de-livros-aumentam-durante-a-pandemia/>. Acesso em: 22 fev. 2022.

MYANIMELIST. **MyAnimeList.** Disponível em: <https://myanimelist.net/>. Acesso em: 14 fev. 2022.

PRESSMAN, R. S.; MAXIM, B. R. Engenharia de software: uma abordagem profissional. 8. ed. Porto Alegre: AMGH Editora Ltda, 2016. p. 968.

SEGS – Portal Nacional de Seguros, Saúde, Info, TI, Educação. **Nielsen/Toluna: consumo de streaming é hábito para 43% dos brasileiros durante a pandemia.** SEGS, c2020. Disponível em: <https://www.segs.com.br/demais/253698-nielsen-toluna-consumo-de-streaming-e-habito-diario-para-43-dos-brasileiros-durante-a-pandemia>. Acesso em: 22 fev. 2022.

VALE, A. Kanban. *In: PRIKLADNICKI, R.; WILLI, R.; MILANI, F. et al. Métodos ágeis para desenvolvimento de software.* Porto Alegre: Bookman, 2014. cap. 8, p. 119-145.