# 1 – Mundo Comum

Uma nova companhia no ramo cinematográfico chamada OpFlix, empresa de pequeno porte que atua no ramo de filmes e séries, foi criada por Tadeu em 2019 na região da Paulista em São Paulo. Tadeu tem uma equipe de parceiros que desejam realizar a divulgação de novos lançamentos por ano.

Sua empresa, por ser nova, realiza a divulgação apenas pelas redes sociais, porém deseja realizar a divulgação por sua página na internet e através de um aplicativo mobile.

# 2 – Chamado para a Aventura

Tadeu lhe contratou para desenvolver um sistema web/mobile integrado onde seja possível realizar a gestão de todos os lançamentos de maneira integrada e otimizada.

Tadeu levantou os seguintes requisitos:

## Sistema Web

### Perfis de usuário:

* **Administrador**: Para o colaborador da gestão administrativa;
* **Cliente**: Clientes da empresa;

### Funcionalidades

1. O administrador poderá cadastrar qualquer tipo de **usuário (administrador ou cliente)**;
2. O administrador poderá cadastrar categorias (contendo nome);
3. O administrador poderá cadastrar o lançamento de um filme/série (contendo título, uma pequena sinopse, a categoria vinculada ~poderá ter somente uma vinculada~, tempo de duração, e se é filme ou série e a data do primeiro lançamento daquele item);
4. O cliente poderá visualizar todos os lançamentos publicados;

## Sistema mobile

### Perfis de usuário:

* **Administrador**: Para o colaborador da gestão administrativa;
* **Cliente**: Clientes da empresa;

### Funcionalidades

1. O cliente poderá visualizar todos os lançamentos;
2. O cliente poderá realizar a ordenação por data de lançamento e/ou categoria;

### Itens Extras (Web)

1. O administrador poderá alterar categorias;
2. O administrador poderá alterar os lançamentos;
3. O administrador poderá cadastrar plataformas (Netflix, Cinema, Amazon);
4. O administrador poderá indicar em qual plataforma será feito o lançamento de determinado item – um lançamento tem somente uma plataforma vinculada;
5. O cliente poderá selecionar somente por mês de lançamento;
6. O cliente poderá favoritar um filme/série;

### Itens Extras (Mobile)

1. Os usuários poderão favoritar um filme/série;
2. Os usuários poderão filtrar por plataforma/mídia/gênero;

## SPRINT 1 – Banco de Dados

A empresa OpFlix contratou você para desenvolver sua solução em software. De acordo com a cultura de desenvolvimento de softwares criadas Escola SENAI de Informática, o primeiro passo para a construção do sistema é desenvolver a modelagem do banco de dados que será utilizada pelo sistema em seus modelos conceitual, lógico e físico.

Para ajudar na construção da modelagem, Tadeu disponibilizou as planilhas onde guardava os dados administrativos da empresa, com informações referentes a: usuários cadastrados, parceiros e clientes. Você utilizará estes documentos como base para a construção do banco e dados.

### Capacidades e critérios

Os critérios estão em vermelho abaixo de cada capacidade

*Banco de Dados*

1. Identificar as características de banco de dados relacionais e não-relacionais

Definiu corretamente a importância do banco de dados

Definiu corretamente o que é banco de dados relacionais

Identificou as diferenças entre bancos de dados relacionais e não-relacionais

2. Configurar o ambiente para utilização de banco de dados relacional

Conectou-se corretamente ao banco de dados local

Liberou o acesso remoto para acesso externo

4. Elaborar diagramas de modelagem do banco de dados de acordo com a arquitetura definida (3)

Exportou o diagrama de modelo lógico de acordo com a situação problema

Exportou o diagrama de modelo conceitual de acordo com a situação problema

Exportou o diagrama de modelo físico de acordo com a situação problema

7. Documentar a estrutura do banco de dados por meio de dicionário de dados

Criou o arquivo de documentação para entrega ao cliente conforme solicitado na situação problema

Incluiu as modelagens no arquivo de documentação conforme solicitado no item 4

Descreveu corretamente as descrições das modelagens

3. Utilizar tipos de dados para definição dos atributos do banco de dados

Utilizou os tipos de dados corretos para os campos das tabelas (usuários, lançamentos, categoria)

Utilizou os tipos de dados corretos para os campos das tabelas (midias)

Utilizou o default para deixar um valor padrão para o valor na tabela de lançamentos

5. Utilizar relacionamentos entre as tabelas do banco de dados

Relacionou as tabelas de acordo com a situação problema (lançamentos com categorias)

Relacionou as tabelas de acordo com a situação problema (lançamentos com mídias, usuários com seus favoritos)

Definiu as chaves primárias de todas as tabelas de acordo com a situação problema

9. Utilizar linguagem de definição de dados (DDL)

Criou todas as tabelas de acordo com a situação problema (usuários, lançamentos, categorias)

10. Utilizar linguagem de manipulação de dados (DML)

Inseriu os dados na tabela de usuários de acordo com as colunas criadas no item 9 e especificadas na situação problema

Atualizou os dados conforme especificação do cliente

Atualizou os registros da tabela de categorias de acordo com a especificação do cliente

6. Normalizar a estrutura do banco de dados

Utilizou a primeira, a segunda e terceira formas normais do banco de dados

Utilizou a quarta e a quinta formas normais do banco de dados

13. Executar importação e exportação da base de dados

Importou os dados para o banco de dados criado de acordo a planilha recebida

Os dados importados estão de acordo com os dados da planilha recebida

Exportou a base de dados para um arquivo sql

Exportou a base de dados para um arquivo csv

11. Utilizar funções nativas do banco de dados

Mostrou a quantidade de usuários após realizar a importação do banco de dados

Calculou quantas dias faltam para o lançamento do filem/série escolhido

12. Aplicar programação em banco de dados utilizando functions, stored procedures, triggers e eventos

Criou um evento para visualizar quantos dias faltam para o lançamento de um filme/série

Criou uma função para retornar a quantidade de filmes cadastrados de uma determinada categoria

Criar um repositório no GitHub chamado 2s2019-sprint-1-bd-opflix e incluir os arquivos abaixo:

Manhã ou Tarde

M\_Documentacao\_SeuNome.docx ou T\_Documentacao\_SeuNome.docx

M\_01\_SeuNome\_DDL.sql ou T\_01\_SeuNome\_DDL.sql

M\_02\_SeuNome\_DML.sql ou T\_02\_SeuNome\_DML.sql

M\_03\_SeuNome\_DQL.sql ou T\_03\_SeuNome\_DQL.sql

M\_Diagrama.png(jpg) ou T\_Diagrama.png(jpg)

Modelos: conceitual, lógico, excel;

Nome Banco de Dados : M\_OpFlix / T\_OpFlix

## SPRINT 2 – API

De acordo com os padrões modernos no desenvolvimento de projetos de software, você e sua equipe decidiram que o software solicitado pela SP TGroup deverá ser criado em plataforma API (Application Programming Interface).

Você deverá desenvolver a API com as seguintes funcionalidades:

## Sistema Web

### Perfis de usuário:

* **Administrador**: Para o colaborador da gestão administrativa;
* **Cliente**: Clientes da empresa;

### Funcionalidades

1. O administrador poderá cadastrar qualquer tipo de **usuário (administrador ou cliente)**;
2. O administrador poderá cadastrar categorias (contendo nome);
3. O administrador poderá cadastrar o lançamento de um filme/série (contendo título, uma pequena sinopse, a categoria vinculada, tempo de duração, e se é filme ou série e a data de lançamento);
4. O cliente poderá visualizar todos os lançamentos publicados;

### Persistência de Dados

Deve ser realizada no banco de dados criado anteriormente na fase de modelagem do sistema

### Autenticação

Deve utilizar o modelo JWT (JSON Web Token) para realizar a autenticação no sistema.

### Autorização

Os endpoints devem ter autorização baseado nas funcionalidades descritas acima. Onde a funcionalidade deve ser somente feita pelo ator estabelecido nela.

Exemplo: O **administrador** poderá **cadastrar** um novo lançamento;

Administrador: O ator

Cadastrar um novo lançamento: A funcionalidade do endpoint;

### Documentação

Deve ser criado a documentação de todos os endpoints disponíveis na aplicação.

### Capacidades

Os critérios estão em vermelho abaixo de cada capacidade

*Web Back-End*

1. Identificar as características de programação backend em ambiente web

Identificou as características dos verbos HTTP (GET, POST, PUT, DELETE)

Identificou as diferenças entre um aplicativo WebAPI e um WebMVC

2. Preparar o ambiente necessário ao desenvolvimento back-end para a plataforma web

Utilizou o Visual Studio para criação do novo projeto WebAPI

Utilizou o prompt de comando para criar um novo projeto WebAPI

4. Utilizar design patterns no desenvolvimento da aplicação web

Criou uma solução com a separação de pastas correta (domínio, repositório e controles) de acordo com a situação problema e seguindo as boas práticas

Utilizou os verbos corretos na criação dos endpoints

Retornou os dados após cadastro de um novo registro (lançamentos)

10. Desenvolver sistemas web de acordo com as regras de negócio estabelecidas

Não permitiu que um lançamento fosse inserido sem uma data de lançamento

Não permitiu que um lançamento fosse inserido sem uma categoria vinculada

5. Definir os frameworks a serem utilizados no desenvolvimento da aplicação web

Os dados são salvos na base de dados utilizando um ORM

6. Utilizar interações com base de dados para desenvolvimento de sistemas web

Conectou-se a base de dados local utilizando o SQLClient

Realizou as operações de leitura, inserção, deleção e atualização (CRUD – Create, Read, Update, Delete) com o banco de dados

9. Desenvolver API (web services) para integração de dados entre plataformas

Criou os endpoints de acesso conforme situação problema

Restringiu o acesso apenas a usuários autenticados conforme situação problema

Restringiu o acesso apenas a usuários autorizados conforme situação problema

3. Definir os elementos de entrada, processamento e saída para a programação da aplicação web

Persistiu os dados enviados pela aplicação cliente conforme situação problema

Validou todos os campos com menor tamanho do que o permitido pela aplicação

Validou todos os campos requeridos enviados pela aplicação cliente

7. Transferir arquivos entre cliente e servidor por meio da aplicação web

Recebeu da aplicação cliente um arquivo do tipo imagem e salvou o caminho no banco de dados

Restringiu a aplicação para que salve apenas arquivos com o tipo de png, jpg

11. Publicar a aplicação web

Disponibilizou em ambiente online, a aplicação back-end funcional

Utilizou corretamente os arquivos de configuração em aplicação para ambientes de produção e desenvolvimento

Entregas:

Público

Login;

Cadastro de novos usuários (pode deixar um endpoint público para cadastro de novos clientes e um outro endpoint aonde administradores cadastram outros administradores);

Administrador

CRUD Lançamentos;

Cadastrar/Listar/Atualizar Categorias;

Cadastrar/Listar/Atualizar Plataformas/Mídias;

Organização

Swagger;

Postman;

Documentação (Word – continuação do arquivo da primeira sprint) – deverá conter a publicação do projeto bem como o desenvolvedor deverá abrir o projeto e realizar todo o script para início;

Projeto

Senai.OpFlix.WebApi

Desafios Extras – API

Mostrar todos os favoritos de um determinado usuário;

Usuário poderá favoritar um determinado lançamento;

Usuário poderá filtrar por data de lançamento;

Usuário poderá filtrar por plataforma;

## SPRINT 3 – Front-end

A OpFlix é uma empresa que preza pela qualidade de seus serviços assim como a interface do usuário. E hoje em dia, sabemos o quanto a interface com o usuário é importante para que o usuário tenha uma experiência satisfatória em qualquer sistema.

Você deverá desenvolver as seguintes atividades:

* Layout de baixa fidelidade;
* Layout de alta fidelidade;
* Teste de usabilidade;

Após ter desenvolvido toda a interface – API do seu sistema e seus dados estarem sendo armazenados no banco, o núcleo desenvolvimetno do SENAI de Informática necessita desenvolver a interface que o usuário fará interação.

Você utilizará o framework React para desenvolver a parte do front-end e realizar a integração com a API desenvolvida anteriormente.

## Sistema Web

### Perfis de usuário:

* **Administrador**: Para o colaborador da gestão administrativa;
* **Cliente**: Clientes da empresa;

### Funcionalidades

1. O administrador poderá cadastrar qualquer tipo de **usuário (administrador ou cliente)**;
2. O administrador poderá cadastrar categorias (contendo nome);
3. O administrador poderá cadastrar o lançamento de um filme/série (contendo título, uma pequena sinopse, a categoria vinculada, tempo de duração, e se é filme ou série e a data de lançamento);
4. O cliente poderá visualizar todos os lançamentos publicados;

### Itens Extras (Web)

1. O administrador poderá alterar categorias;
2. O administrador poderá alterar os lançamentos;
3. O administrador poderá cadastrar apenas uma plataforma (Netflix, Cinema, Amazon);
4. O administrador poderá indicar em qual plataforma será feito o lançamento de determinado item;
5. O cliente poderá selecionar somente por mês de lançamento;
6. O cliente poderá favoritar um filme/série;

### Capacidades

Os critérios estão em vermelho abaixo de cada capacidade

*Web Front-End*

1. Adequar a interface web para diferentes dispositivos de acesso

Utilizou media querys para deixar o site responsivo conforme situação problema (Mobile, iPad e Web – 600, 768, 1366, 1920)

Utilizou JavaScript para adequar o site a diferentes tipos de acesso

2. Desenvolver interfaces web interativas com linguagem de programação

Validou os campos de data de nascimento do paciente (não pode ser maior do que a data atual)

Validou os campos requeridos na aplicação cliente

Inseriu novos registros na tabela de pacientes a partir dos dados inseridos pelo usuário

4. Desenvolver interfaces web consumindo API

Enviou os dados da aplicação cliente para o servidor

Listou as informações do servidor, na aplicação cliente

Realizou a atualização dos registros em tela, após enviar uma atualização para o servidor

3. Desenvolver interfaces web utilizando frameworks

Utilizou algum framework front-end na construção do layout

Disponibilizou a aplicação em ambiente online conforme situação problema

Utilizou uma dependência externa para facilitar o desenvolvimento do software (conexão com a API)

5. Diferenciar os aspectos de aplicabilidade entre as experiências do usuário (UX) e a interface do usuário (UI)

Criou os layouts de alta fidelidade para web de acordo com a situação problema

Criou os layouts de baixa fidelidade web de acordo com a situação problema

Realizou os testes de usabilidade da aplicação web com o cliente da aplicação de acordo com a situação problema

Telas

Críticos

**Administrador**

Realizar Login;

Cadastrar novo lançamento;

Listagem de lançamentos;

Atualizar lançamentos;

Deletar Lançamentos;

Listagem de categorias;

**Cliente**

Realizar Login;

Listagem de lançamentos;

Desejáveis

Administrador

Cadastrar nova plataforma;

Cadastrar nova categoria;

Editar plataformas e categorias;

Usuário

Poderá favoritar um lançamento

## SPRINT 4 – Mobile

A OpFlix é uma empresa que preza pela qualidade de seus serviços assim como a interface do usuário. E hoje em dia, sabemos o quanto a interface com o usuário é importante para que o usuário tenha uma experiência satisfatória em qualquer sistema.

Você deverá desenvolver as seguintes atividades:

* Layout de baixa fidelidade;
* Layout de alta fidelidade;
* Teste de usabilidade;

Após um tempo de uso de sua aplicação web, Tadeu percebeu que para melhorar a usabilidade de sua plataforma, é necessário desenvolver uma aplicação mobile focando no cliente e em quais informações ele precisa visualizar.

Você deverá desenvolver um aplicativo mobile integrando com a API que foi desenvolvida anteriormente (Sprint 2) e deverá conter as seguintes funcionalidades:

## Sistema mobile

### Perfis de usuário:

* **Administrador**: Para o colaborador da gestão administrativa;
* **Cliente**: Clientes da empresa;

### Funcionalidades

1. O cliente poderá visualizar todos os lançamentos;
2. O cliente poderá realizar a ordenação por data de lançamento e/ou categoria;

### Itens Extras (Mobile)

1. Os usuários poderão favoritar um filme/série;
2. Os usuários poderão filtrar por meio de veiculação/mídia/plataforma;

### Persistência de dados

Os dados referentes ao usuário logado na aplicação mobile, deverão ser salvos localmente;

### Capacidades

Os critérios estão em vermelho abaixo de cada capacidade

*Interfaces para Dispositivos Móveis*

1. Identificar as características de programação de dispositivos móveis

Identificou as diferenças entre aplicativos híbridos e nativos

Identificou as vantagens entre aplicativos híbridos e nativos

3. Preparar o ambiente necessário ao desenvolvimento do sistema para a plataforma mobile

Instalou o node em ambiente local para desenvolvimento

Instalou o React Native

Configurou a variável de ambiente para utilizar o npm em qualquer pasta de seu computador

7. Implementar o código respeitando as características da linguagem na plataforma mobile

Utilizou código nativo para criar a SplashScreen

2. Utilizar os elementos da programação orientada a objetos em interfaces para dispositivos móveis

Utilizou polimorfismo para criação das telas

Utilizou interface para criação das telas

Utilizou abstração para criação das telas

5. Definir os elementos de entrada, processamento e saída para a codificação das funcionalidades mobile

Permitiu que somente usuários autenticados acessem o menu de navegação

Mostrou uma mensagem para o usuário após realizar login inválido

6. Projetar interfaces para dispositivos móveis

Criou os layouts de alta fidelidade para dispositivos móveis de acordo com a situação problema

Criou os layouts de baixa fidelidade para dispositivos móveis de acordo com a situação problema

Realizou os testes usabilidade da aplicação mobile com o cliente da aplicação de acordo com a situação problema

4. Interpretar os requisitos do sistema, tendo em vista a elaboração dos componentes em ambiente mobile

Criou os componentes de lista conforme situação problema

Criou os componentes de entrada de dados conforme situação problema

Telas

**Cliente**

Realizar Login;

Listagem de lançamentos;

Filtros;

## SPRINT 5 – NoSQL

Com o software para finalizar, a OpFlix decidiu visualizar em quais cidade e/ou regiões os lançamentos costumam acontecer por localização.

### Capacidades e critérios

Os critérios estão em vermelho abaixo de cada capacidade

*Banco de Dados*

1. Identificar as características de banco de dados relacionais e não-relacionais

Identificou as diferenças entre banco de dados relacionais e não-relacionais

Documentou no arquivo especificado na situação problema, a estrutura utilizada no projeto de big data

8. Configurar usuário e permissões de acesso ao banco de dados

Criou um usuário para realizar todas as operações na base de dados criada

Criou um usuário para realizar somente leitura na base de dados criada

*Web Back-End*

8. Estabelecer envio de notificações entre cliente e servidor por meio de aplicação web

Mostrar uma mensagem customizada do backend, após um novo lançamento ser incluído

9. Desenvolver API (web services) para integração de dados entre plataformas

Criou um endpoint em que o administrador cadastre novos registros

*Web Front-End*

3. Desenvolver interfaces web utilizando frameworks

Criou uma interface gráfica para realizar o cadastro de novos registros.