

Aprendizagem por Projetos Integrados 2023-1

Cliente:	IONIC Health	
Período / Curso:	4º ADS – Análise e Desenvolvimento de Sistemas	
Professor M2:	Arley Ferreira de Souza	arley.souza@fatec.sp.gov.br
Professor P2:	Juliana Pasquini	pasquini.juliana@fatec.sp.gov.br
Contato do Cliente:		
Tema do Semestre		
Sistema Web com Regra de Negócio complexa em ambiente de produção (Deploy). Preparação e configuração de ambientes para práticas de engenharia também fazem parte do escopo.		
Conhecimentos exigidos no semestre		
<ul style="list-style-type: none">• Construir Modelo Conceitual de banco de dados utilizando uma ferramenta de modelagem• Converter Modelo Conceitual de banco de dados em Modelo Lógico de BD.• Normalizar as tabelas de um banco de dados.• Implementar Projeto de Banco de Dados Relacional em termos de seus Principais Objetos (Schema, Tabelas, Views).• Implementar Consultas em um banco de dados, utilizando junções, subconsultas e agrupamentos .• Implementar manutenção de dados utilizando DML.• Implementar FrontEnd SPA (Single Page Application) com React.• Implementar Aplicação usando NodeJS e características técnicas mais avançadas.• Implementar Aplicação usando ORM TypeORM.• Implementar junção de padrões como o MVC que é a junção do Observer + Strategy + Composite• Implementar outros padrões aproveitando a semelhança de conceitos entre eles, por exemplo: State, Adapter e Bridge possuem uma arquitetura semelhante ao Strategy.		
Desafio (problema)		
<p>Temos atualmente dores relacionadas a integração de funcionalidades necessárias para o fluxo de trabalho da Ionic Health. Precisamos de uma plataforma capaz de receber e gerenciar chamados internos dentro de uma sprint de projeto, sendo possível configurar uma árvore de decisão para redirecionar demandas de acordo com a classificação (hotfix ou feature).</p> <p>Para uma feature, a plataforma precisa ser capaz de direcioná-la para um comitê de aprovação, onde poderemos configurar quais etapas serão necessárias para avaliação da viabilidade do projeto. Para uma hotfix, é necessário que a correção seja escalonada diretamente na sprint do produto (sem a necessidade de aprovação).</p> <p>Exemplo de fluxo:</p> <ul style="list-style-type: none">➔ Abertura de uma nova feature:1) Solicitante envia as informações por um formulário2) Plataforma escalona para o comitê de aprovação<ul style="list-style-type: none">2.1) CSO e RT: preenchem um documento de análise de risco		

2.2) Plataforma analisa risco (em caso de alto risco, a solicitação é descartada para uma pasta de demandas rejeitadas)

2.3) CTO e Head: Alinhamento estratégico (Approved ou Disapproved). Caso seja aprovado, o head precisa ser capaz de definir um nível de prioridade a solicitação e qual time será responsável.

3) Plataforma direciona demanda para o time designado.

4) Controle de status (New, On Holding, Done).

5) Após a conclusão, a solicitação fica salva no banco.

➔ Abertura de uma hotfix:

Para uma hotfix, o fluxo será idêntico ao item 2.3 de “Abertura de uma nova feature” em diante.

Requisitos **apresentados** para os alunos

Requisitos Funcionais:

- Cadastros de grupo de usuários.
- Permissão por grupo e por usuário
- Usuários devem ter distinção por times.
- Fila de chamados por time
- Configuração e personalização de formulário de chamados.
- Árvore de decisão para escalonamento de chamados.
- Cadastro de soluções para problemas conhecidos.
- Formulário de abertura de chamado com indicação de interessados.

Requisitos Não Funcionais:

- React
- NodeJS

Tecnologias Desejáveis:

- MySQL/Oracle
- BrModelo
- MySQL Workbench/SQLDeveloper
- Vertabelo
- Heroku
- TypeScript