São Paulo, 18 de julho de 2024

**Relatório Técnico - FIAPTask**

**Link do repositório: https://github.com/FIAPflix/fiaptask**

**por:** Ariel Fortes e Bruna Baria

**Tech Challenge - FIAPTask**

O Tech Challenge - FIAPTask é um projeto destinado à conclusão de curso da pós -graduação Coding & Tech Journey From User to Creator. Neste quinto módulo que compõem a grade do curso, desenvolvemos um sistema de gerenciamento de tarefas utilizando no front-end reativo com as tecnologias JavaScript, React,CSS e Booststrap, no back-end utilizamos a plataforma Firebase para produção de API's, gerenciamento de banco de dados e autenticação. Desta forma, criamos um dashboard simples e intuitivo.

**Funcionalidades**

Cadastro de Membros do Time: Permitir que o usuário registre membros do time, armazenando informações relevantes como nome, email e função;

Gestão de Tarefas: Os usuários podem criar tarefas, definindo descrição, prazo, prioridades e o tempo estimado para conclusão;

Atribuição de Tarefas: Associar tarefas aos membros do time, permitindo que mais de uma tarefa seja atribuída a um membro;

Cálculo de Alocação de Tempo: A aplicação deve calcular automaticamente o total de horas alocadas para cada membro do time, considerando todas as tarefas que lhe foram atribuídas;

Notificações de Prazo: Implementar um sistemas de notificações, utilizando o firebase, para alertar sobre tarefas que estão se aproximando do prazo final.

Tecnologias utilizadas

Node.js, Express, Firebase, JavaScript, React, HTML, CSS, Bootstrap.

**Principais arquivos e pastas**

server.js: Ponto de entrada da aplicação;

routes/: Conténs as rotas das API's;

view/: contém as páginas da aplicação.

**Pré-requisitos**

Node.js Instalado na máquina local ou em um servidor.

npm express: npm install express

npm EJS: npm install ejs

npm dotenv: npm install dotenv

npm express-session: npm install express-session

npx create-react-app .

npm firebase: npm install --save firebase

npm install firebase-admin

npm nodemon: npm install -g nodemon

npm jquery: npm install jquery

**Instalação**

1. Clone ou faça o download deste repositório ou acesse o diretório do kit de arquivos baixados referentes à este projeto.
2. Instale as dependências npm acima citadas;
3. Executar o servidor com npm run;
4. Configurar as variáveis de ambiente;
5. Acesse o Dashboard no navegador para utilizar o FIAPtask através do localhost.

**Utilização**

Após as devidas instalações npm, acesse o arquivo index.html localizado no diretório de views. Através desta página, você poderá:

1. Realizar autenticação de usuário - através de uma conta google;
2. Acessar a Homepage;
3. Consultar os time, tarefas ou colaboradores cadastrados;
4. Atualizar/Excluir times, tarefas ou Colaboradores;
5. Cadastrar novos times, tarefas ou colaboradores.

Em Home: Ao clicar em "Times", "Tarefas" ou "Colaboradores" no menu nav-bar, ou nos botões destacados no corpo da na homepage, você será redirecionado(a) à página de consulta de dados.

Para Consultas: Nesta página, identificada como consultaColaborador.html, consultaTime.html ou consultaTarefa.html, você poderá:

1. Utilizar o menu nav-bar para navegação entre as demais páginas do projeto;
2. Visualisar todas as tarefas, times e colaboradores já cadastrados nas respectivas telas acessadas;
3. Clicar no botão de "lápis" azul ou "lixeira" vermelha para editar o cadastro da linha selecionada - será direcionado(a) página de edições;
4. Nesta página também há informações no rodapé sobre os desenvolvedores da FIAPtask.

Para Edições:

Ao selecionar a opção de "lixeira", a linha escolhida será excluída; Ao selecionar a opção de "lápis" você poderá:

1. Na edição de tarefa: Um ID, o nome da tarefa e a descrição serão pré-carregados e não será possível editá-los, apenas os campos de data, prioridade, tempo para conclusão e responsável;
2. Na edição de colaboradores: O nome, o email e o cargo serão pré-carregados e não será possível editá-los, apenas o campo de "ativo";
3. Na edição de time: O nome será pré-carregado e não será possível editá-lo, apenas o status. Nessa etapa você poderá visualizar as tarefas atribuídas a este time e os colaboradores vinculados; Em qualquer um destas páginas haverá um botão "cadastrar" ou "salvar" para submeter o formulário e um botão "voltar" para retornar à página anterior. Você poderá utilizar o menu nav-bar para navegação entre as demais páginas do projeto, bem como encontrar informações no rodapé sobre os desenvolvedores da FIAPtask

Para Novos Cadastros: Ao clicar em "+ Novo", a direita do menu superior nav-bar, um menu dropdown será exposto com opções para clicar em "colaborador", "time" e "tarefa". Ao selecionar qualquer uma dessas opções, você será direcionado(a) para a tela de novo cadastro. Nesta página, identificada como novaTarefa.html, novoColaborador.html e novoTime.html você poderá:

1. Utilizar o menu nav-bar para navegação entre as demais páginas do projeto;
2. Visualizar um ID automaticamente gerado para esse cadastro;
3. Para nova tarefa, os campos possível para preenchimento são: nome da tarefa, descrição da tarefa, data, prioridade, quantidade de horas para conclusão e responsável.
4. Para novo colaborador, os campos possíveis para preenchimento são: nome completo, email, cargo e um menu dropdown para informar se está ativo, afastado ou desligado;
5. Para novo time, os campos possíveis para preenchimento são: nome do time e status;

Para a todas as páginas, haverá um botão "cadastrar" para submeter o formulário e um botão "voltar" para retornar à página anterior;

Nesta página também há informações no rodapé sobre os desenvolvedores da FIAPtask.

**Notas dos Desenvolvedores**

Utilizamos as tecnologias acima descritas afim de desenvolver e entregar o desafio pré definido. A proposta inicial era desenvolver um modelo de arquitetura MVC em JavaScript, HTML, CSS e Node.js para lidar com os CRUDs necessários ao projeto. Desenvolver um database e API's, bem como configurar o autenticador Firebase. Encontramos dificuldade em alocar as funcionalidades de acordo com sua respectiva proposta de model e controller. Nesse sentido utilizamos um modelo de middleware e rotas para estabelecer a conexão entre o frontend e o backend. Escolhemos manter os diretórios de controller e model, pois planejamos inicialmente utilizá-los. Ficará para uma implementação futura essa melhoria. A etapa de comunicação entre a aplicação e o firebase, autenticação e consulta de dados foi particularmente desafiadora, consumindo horas relevantes de trabalho, mas felizmente superada e implementada. A entrega de algumas outras funcionalidades planejadas, entretanto, ficará para uma implementação futura. Tais como a consulta e CRUD do Perfil da pessoa logada, o CRUD de times,os testes unitários e o menu sandwich. O envio de notificação ao aproximar-se da data limite de conclusão da tarefa não foi implementado pois a ferramenta Firebase solicita a alteração de plano gratuito para o "blaze". Neste caso, optamos por não configurar possíveis cobranças para esse projeto.

Disponibilizaremos os arquivos de prototipagem deste projeto. É possível perceber as adaptações entre as ideias planejadas e executadas. Optamos por não ajustar o protótipo para observarmos justamente as adaptações e eventualmente, modificar o projeto até alcançar o planejado, futuramente.