**Proposta de Projeto Integrador**

**Data: 28/08/2024 Grupo: .Netos**

1. **Nome Projeto:** Pé na trilha
2. **Nome Usuário no GitHub:** julio-sa
3. **Grupo de Alunos:**

|  |  |  |
| --- | --- | --- |
| **RA** | **Nome** | **e-mail** |
| **0030482223004** | **Andre Daisuke Motoda** | **andre.motoda@fatec.sp.gov.br** |
| **0030482223031** | **Gabriel Telo Mariano** | **gabriel.mariano8@fatec.sp.gov.br** |
| **0030482223002** | **Julio Henrique Silva de Assis** | **julio.assis01@fatec.sp.gov.br** |
| **0030482223044** | **Leonardo José de Camargo** | **leonardo.camargo18@fatec.sp.gov.br** |
| **0030482323006** | **Ricardo Gomes Marques** | **ricardo.marques5@fatec.sp.gov.br** |

1. **Compreensão do Problema**

O ecoturismo é uma atividade fundamental para a conservação ambiental, proporcionando uma conexão direta entre as pessoas e a natureza, enquanto promove a conscientização sobre a importância da preservação de ecossistemas. No entanto, existe uma lacuna de informação significativa sobre locais ideais para a prática dessa atividade. Muitos ecoturistas deixam de conhecer trilhas e parques por não terem acesso a informações detalhadas sobre esses locais.

Nesse contexto, a tecnologia desempenha um papel crucial. O desenvolvimento de plataformas digitais pode fornecer aos usuários dados sobre trilhas, parques e atividades, facilitando o planejamento de visitas e promovendo o ecoturismo sustentável. A utilização de um aplicativo como o Pé na Trilha visa resolver essa lacuna, conectando usuários a guias e parques, além de criar um canal eficiente de disseminação de informações sobre o ecoturismo.

O Pé na Trilha, desenvolvido inicialmente para dispositivos móveis, focava em conectar ecoturistas, guias e gestores de parques, facilitando a busca de trilhas e atividades. O sistema original já permitia funcionalidades essenciais como a marcação de trilhas visitadas. No entanto, há espaço para melhorias e expansões.

1. **Proposta de Solução de Software e Viabilidade**

A proposta deste projeto é a partir de uma aplicação já existente, o “Pé na trilha”, desenvolvida para web, mas *mobile-first*; ou seja, não possui responsividade adequada para desktops e notebooks. O aplicativo visa conectar Ecoturistas, guias licenciados e gestores de parque a fim de promover o ecoturismo.

Como ideias para incrementação da aplicação, propõe-se novas funcionalidades, tais como:

Como possíveis incrementos, o grupo propõe:

* Adaptação da responsividade para telas de computadores de mesa e notebooks.
* Conexão com a base de dados do CNUC - [Plataforma CNUC — Ministério do Meio Ambiente e Mudança do Clima (www.gov.br)](https://www.gov.br/mma/pt-br/assuntos/biodiversidade-e-biomas/areas-protegidas/plataforma-cnuc-1).
* Criar uma funcionalidade para que o visitante possa oficializar a visita ao parque NO CASO DE TRILHAS MONITORADAS (visitante marca como visitado -> o guia aprova -> dados da visita vão para uma base de dados).
* Adicionar tela para os gerentes dos parques poderem consultar o histórico de visitação MONITORADA.
* Sistema de ranqueamento de dificuldade e necessidade de expertise do visitante para execução da trilha. (Já possui nível de dificuldade no projeto original, pode ser adaptado para uma classificação por ícone) guias???
* Sistema de pontuação que o visitante recebe à cada trilha concluída.
* Criação de um avatar para o usuário dentro da aplicação, com a pontuação o usuário não só aumenta o nível, como pode trocar as pontuações por recompensas para o avatar (estilo Samsung AR Zone). (pode implementar na tela "Meu Perfil")
* Sistema de status do avatar (Ex.: Observação, Resistência, Conhecimento e afins).
* Página de avaliações dos guias, com comentários e sistema de estrelas (0 a 5, estilo Uber).
* Sistema de avaliação do parque em diversos quesitos, organização, limpeza, conservação, imersão proporcionada por eventos, recepção dos funcionários do parque.
* Chat estilo Desk, em que o visitante pode entrar em contato pela plataforma com o parque e com os guias para solicitar informações adicionais.
* Insígnias para parques e trilhas conquistados que demonstram o "prestígio" do aventureiro (estilo Pokémon).
* Sistema de “*Side Quests*” que o visitante pode realizar para obter recompensas exclusivas e ganhar pontos adicionais (tirar foto de um ponto específico, de uma planta específica, realizar um quis liberado pelo guia em um determinado momento da trilha, conseguir uma foto da fauna local, etc.).
* Criação de equipes para "*Raids*", geralmente para eventos sazonais, criação de tarefas simples que os usuários devem completar em conjunto para melhoras suas pontuações.

1. **Visão Geral dos Pré-Requisitos**

O sistema pode manter o layout proposto pelo grupo anterior, mas será necessário ajustar uma responsividade para a aplicação web, além da implementação dos tópicos acima mencionados.

O sistema **Pé na Trilha**, conforme desenvolvido pelo grupo anterior, apresentou uma aplicação com foco em plataformas móveis, oferecendo informações sobre trilhas e parques ecológicos. O layout e as funcionalidades básicas implementadas já atendem a muitos dos requisitos essenciais para a prática de ecoturismo, como busca por parques, visualização de guias e atividades, além da possibilidade de marcar parques como "visitados".

Contudo, para que a aplicação seja aprimorada e ofereça uma experiência mais rica e acessível, alguns ajustes são necessários, especialmente no que diz respeito à responsividade da interface para dispositivos móveis e web. Além disso, serão implementadas novas funcionalidades para expandir a experiência do usuário, como a gamificação, avaliações de parques e guias, entre outras descritas anteriormente.

**Requisitos a serem implementados:**

**Requisitos Funcionais:**

1. **Responsividade**:

A aplicação deve ser totalmente responsiva, garantindo uma interface otimizada tanto para dispositivos móveis quanto para a web.

1. **Gamificação**:

Implementar um sistema de pontuação, insígnias e conquistas. Os usuários devem ganhar pontos ao completar trilhas e desafios.

1. **Perfil Personalizado com Avatares**:

Usuários devem ter perfis personalizáveis com avatares que evoluem conforme acumulam pontos e completam trilhas.

1. **Sistema de Avaliação**:

Adicionar um sistema de avaliação com notas de 0 a 5 estrelas para trilhas, guias e parques, permitindo também comentários dos usuários.

1. **Histórico de Atividades**:

O sistema deve manter um histórico de trilhas e atividades completadas pelo usuário, com base nesse histórico, oferecer sugestões personalizadas.

1. **Integração com APIs de Geolocalização e Clima**:

A aplicação deve se conectar a APIs de geolocalização e informações climáticas, ajudando o usuário a planejar trilhas de forma mais precisa.

1. **Sistema de Recompensas**:

Usuários devem acumular pontos que poderão ser trocados por recompensas ou melhorias visuais no avatar.

1. **Acessibilidade**:

A aplicação deve seguir padrões de acessibilidade, garantindo que pessoas com deficiência visual e auditiva possam utilizar todas as funcionalidades.

1. **Sistema de Ranqueamento e Nível de Dificuldade**:

Trilhas devem ser classificadas com base em nível de dificuldade e requisitos técnicos, oferecendo uma visão clara para o ecoturista.

1. **Chat para Atendimento**:

Implementar uma funcionalidade de chat para que usuários possam tirar dúvidas com os gestores dos parques ou guias em tempo real.

1. **Criação de Times e Eventos Sazonais**:

Usuários devem poder formar equipes para participar de desafios e eventos sazonais, promovendo a interação em grupo.

1. **Conexão com Bases de Dados de Trilhas**:

A aplicação deve se integrar a bases de dados para fornecer informações mais detalhadas e atualizadas sobre trilhas e parques.

**Requisitos Não Funcionais:**

1. **Desempenho**:

O sistema deve garantir tempos de resposta rápidos, mesmo com a adição de novas funcionalidades, mantendo uma boa performance em dispositivos móveis e web.

1. **Segurança**:

Implementar autenticação segura para todos os usuários, garantindo que dados pessoais e históricos de atividades estejam protegidos.

1. **Compatibilidade**:

A aplicação deve ser compatível com os principais navegadores (Google Chrome, Safari, Edge, Firefox) e plataformas móveis (Android e iOS).

1. **Disponibilidade**:

A aplicação deve estar disponível 24 horas por dia, 7 dias por semana, com mínima possibilidade de interrupções.

1. **Conceitos e Tecnologias Envolvidos**

Como será aproveitado o projeto anterior, considera-se, a princípio, as mesmas ferramentas utilizadas pelo grupo anterior.

Banco de dados:

* PostgresSQL (16.0.0): por ser conhecido por sua robustez, riqueza de

recursos e desempenho; e por dois dos membros do projeto já terem

familiaridade com a linguagem;

*Back-end*:

* Node.js (20.12.2): ambiente de execução JavaScript que permite executar aplicações desenvolvidas com a linguagem de forma autônoma, sem depender de um navegador. Com ele, é possível criar praticamente qualquer tipo de aplicações web, desde servidores para sites estáticos e dinâmicos, até APIs e sistemas baseados em micro serviços.
* Docker (3.8): plataforma open source que facilita a criação e administração de ambientes isolados. Nele você consegue criar, implantar, copiar e migrar de um ambiente para outro com maior flexibilidade.
* Prisma (5.3.1): ORM (Object-Relational Mapping) que ajuda na criação de aplicações de forma mais rápida e com menos erros, contando com um kit de ferramentas de banco de dados open source, como PostgreSQL e MySQL. Além disso, o Prisma também suporta as linguagens Javascript e Typescript.
* Insomnia (2023.5.8): Framework Open Source para desenvolvimento/teste de API Clients. Pode ser usado para envio de requisições REST, SOAP.
* Fastify13 (4.24.3): é um framework web para Node.js que se destaca pela sua velocidade e eficiência. Ele foi projetado para ser rápido, leve e fácil de usar, oferecendo um desempenho excepcional para construir aplicativos web escaláveis.

*Front-end*:

* Node.js (20.12.2): ambiente de execução JavaScript que permite executar aplicações desenvolvidas com a linguagem de forma autônoma.
* Next.js (14.2.1): estrutura da web de desenvolvimento front-end React de código aberto que permite funcionalidades como renderização do lado do servidor e geração de sites estáticos para aplicativos da web baseados em React.

1. **Situação atual (estado-da-arte)**

**Falar da aplicação existentes**

**Aqui falar das coisas que podem melhorar**

A aplicação existente já possui bastantes funcionalidades, como as telas abaixo:

* Login
* Logout
* Recuperar senha
* Pesquisar parques
* Pesquisar eventos
* Registrar conta
* Gerenciar perfil
* Gerenciar vínculos com parques
* Gerenciar parques visitados
* Validar gerentes de parques
* Gerenciar parque
* Gerenciar dicas e boas práticas
* Gerenciar atividade
* Gerenciar infraestrutura
* Gerenciar evento
* Gerenciar guia
* Validar guia

Porém, como já mencionado, não possui responsividade adequada para desktops e notebooks. Além disso, a aplicação não permite a comunicação entre as partes envolvidas (visitante, guia, gestor de parque e administrador do sistema) por meio dela mesma, da aplicação propriamente dita.

Referente à funcionalidade do visitante poder marcar um parque como visitado e excluir essa marcação, entende-se que é possível criar outra funcionalidade para que o visitante afirme que concluiu uma atividade monitorada, e o guia responsável confirmar, também na aplicação; isso geraria uma base de dados disponível para os gestores dos parques bem como para os administradores do sistema. Evidentemente, essa funcionalidade não busca substituir o atual sistema adotado nas unidades de conservação, onde os dados são anotados manualmente em papel sulfite, e armazenados em arquivos. A ideia é apenas complementar, caso haja interesse da gerência dos parques.

Pensou-se também na possibilidade de tornar a aplicação mais comercializável, por meio da gamificação, e parcerias. Isso não só a tornaria mais atrativa, como também permitiria maior difusão entre possíveis usuários.

Na situação atual, a aplicação já possibilita o envio de e-mails para as partes envolvidas. Seria interessante usar essa funcionalidade para a implementação do “help-desk” (mencionado acima). Atualmente, a comunicação dos parques com os visitantes é apenas via telefone e, principalmente, e-mail. Para que os servidores da administração do parque e os guias possam se comunicar com os usuários via aplicação, seria interessante que existisse um sistema de notificação, como acontece entre o Outlook e o Teams, da Microsoft (quando um usuário recebe uma mensagem no Teams fora do horário de uso, um e-mail é enviado com a informação de que possui mensagens não lidas).

Algumas aplicações encontradas no mercado são:

* Parques SP (Guia Virtual de apoio ao visitante com informações sobre os Parques Estaduais Carlos Botelho, Ilha do Cardoso, Intervales, Caverna do Diabo, Turístico do Alto Ribeira (PETAR) e Ilhabela, [Parques SP – Apps no Google Play](https://play.google.com/store/apps/details?id=br.gov.sp.sma.parquessp&hl=pt_BR&gl=US))
* BoRa Parque Nacional do Iguaçu (Aprenda mais sobre as Cataratas e a biodiversidade da Mata Atlântica de um jeito leve e divertido. Colecione suas figurinhas e prêmios no álbum. Uma experiência acessível com audiodescrição e língua de sinais brasileira - Libras., [BoRa Parque Nacional do Iguaçu – Apps no Google Play](https://play.google.com/store/apps/details?id=br.com.fubaea.bora.pni&hl=pt&gl=US))
* BoRa Parque Ecológico (Aprenda mais sobre os animais do parque de um jeito leve e divertido. Colecione suas figurinhas e prêmios no álbum. Uma experiência acessível com audiodescrição e vídeos em Libras., [BoRa Parque Ecológico – Apps no Google Play](https://play.google.com/store/apps/details?id=br.com.fubaea.bora.pesc&hl=pt_BR&gl=US))
* Parques do Brasil (NÃO DISPONÍVEL: reúne informações sobre as principais unidades de conservação (UCs) do país, aquelas responsáveis pelo maior fluxo de visitantes. Pela ferramenta, é possível pesquisar informações sobre as UC mais próximas do usuário, incluindo orientações sobre como chegar, atrativos, descrição das trilhas, atividades disponíveis, o bioma da unidade, as principais espécies protegidas, condições de acessibilidade e preços de ingressos., [MMA lança APP Parques do Brasil — Instituto Chico Mendes de Conservação da Biodiversidade (www.gov.br)](https://www.gov.br/icmbio/pt-br/assuntos/noticias/ultimas-noticias/mma-lanca-app-parques-do-brasil))

1. **Estimativa de custo do projeto**

Para calcular o custo do projeto "Pé na Trilha", utilizamos uma base de valores compatível com o mercado, considerando a remuneração média de desenvolvedores full stack e outros custos associados ao desenvolvimento de uma aplicação de ecoturismo. Baseamos o cálculo em um salário médio de **R$ 5.000,00** por desenvolvedor, conforme referência de sites como **Glassdoor**.

Além dos custos com desenvolvimento, incluímos os valores estimados para a integração com APIs de geolocalização, hospedagem em servidores cloud, e funcionalidades de gamificação, que são diferenciais essenciais do projeto. Para calcular os custos de APIs, tomamos como base valores disponíveis em serviços amplamente utilizados no mercado, como **Google Maps** e **Mapbox**.

A publicação do aplicativo nas lojas de aplicativos como **Play Store** e **App Store** também foi considerada, com base nas taxas de licenciamento padrão dessas plataformas. Além disso, custos com hospedagem, licenças e testes para assegurar a qualidade da aplicação são considerados. Também foi previsto um valor para gastos imprevistos, visto que projetos dessa natureza podem exigir ajustes durante o desenvolvimento.

**Tabela 1 – Estimativa de custos do projeto "Pé na Trilha"**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Descrição** | **Custo Mensal (R$)** | **Tempo** | **Custo Total (R$)** | **Observações** |
| **Salário dos Desenvolvedores** | R$ 25.000,00 | 6 meses | R$ 150.000,00 | R$ 5.000,00 por desenvolvedor (5 desenvolvedores) |
| **Integração com APIs** | R$ 1.500,00 | 6 meses | R$ 9.000,00 | Integração com APIs de geolocalização e bases de dados públicas de trilhas |
| **Desenvolvimento de gamificação** | R$ 2.000,00 | 3 meses | R$ 6.000,00 | Implementação de gamificação (avatares, sistema de pontos e side quests) |
| **Hospedagem em servidores cloud** | R$ 1.200,00 | 6 meses | R$ 7.200,00 | Hospedagem em servidores cloud para manter a aplicação online |
| **Desenvolvimento de chat e sistema de mensagens** | R$ 1.500,00 | 6 meses | R$ 9.000,00 | Implementação de chat para interação entre usuários e gestores de parques |
| **Teste e validação de funcionalidades extras** | R$ 1.000,00 | 3 meses | R$ 3.000,00 | Testes e validação de usabilidade e performance das funcionalidades extras |
| **Licença para publicação nas lojas de aplicativos** | - | Único | R$ 500,00 | Licenças para publicação nas plataformas Play Store e App Store |
| **Gastos imprevistos** | - | - | R$ 10.000,00 | Reserva para custos não planejados |
| **Total Estimado (Aprox.)** | - | - | R$ 194.700,00 | Valores variáveis podem surgir |

1. **Glossário**

* **Aplicação Móvel**: Software desenvolvido para ser executado em dispositivos móveis como smartphones e tablets.
* **API (Application Programming Interface)**: Conjunto de rotinas, protocolos e ferramentas que facilitam a comunicação entre diferentes sistemas, permitindo a integração de funcionalidades externas, como dados de geolocalização e clima.
* **Avatar**: Representação gráfica do usuário dentro da aplicação, que pode ser personalizada e melhorada conforme o usuário avança no sistema de pontuação.
* **Back-end**: Parte do sistema responsável pelo processamento de dados e pela lógica de negócios, funcionando nos bastidores da aplicação. Desenvolvido com Node.js.
* **Base de Dados**: Repositório de informações onde serão armazenados os dados de trilhas, usuários, avaliações, entre outros. Utiliza o PostgreSQL como sistema de gerenciamento de banco de dados.
* **Chat**: Ferramenta de comunicação dentro do aplicativo que permite aos usuários se conectarem com gestores de parques ou guias para obter informações adicionais.
* **Docker**: Plataforma que permite criar e administrar ambientes isolados para desenvolvimento, garantindo consistência entre ambientes de desenvolvimento e produção.
* **Ecoturismo**: Modalidade de turismo voltada para a exploração de áreas naturais, com foco na conservação ambiental e no aprendizado sustentável.
* **Fastify**: Framework web utilizado para construir APIs de forma rápida e eficiente, oferecendo alto desempenho em aplicações Node.js.
* **Gamificação**: Técnica de design que usa elementos de jogos, como pontuação, níveis e recompensas, para engajar os usuários e aumentar a motivação na utilização da aplicação.
* **Geolocalização**: Tecnologia que permite determinar a localização geográfica de um dispositivo, usada no aplicativo para mostrar trilhas e pontos de interesse próximos ao usuário.
* **Insígnias**: Condecorações digitais que os usuários ganham ao completar trilhas ou atingir metas específicas, exibindo suas conquistas na plataforma.
* **Next.js**: Framework front-end baseado em React que oferece renderização no servidor e geração de páginas estáticas para melhorar o desempenho e a experiência do usuário.
* **Node.js**: Ambiente de execução que permite o uso de JavaScript para o desenvolvimento do lado do servidor, garantindo escalabilidade e alta performance.
* **ORM (Object-Relational Mapping)**: Ferramenta que facilita a interação entre a aplicação e o banco de dados, transformando dados relacionais em objetos da linguagem de programação. Utiliza o Prisma.
* **Prisma**: ORM que facilita a manipulação do banco de dados PostgreSQL, permitindo a criação e gestão de consultas de forma eficiente e segura.
* **Raids**: Eventos sazonais ou missões em grupo onde os usuários formam equipes para completar desafios coletivos e aumentar suas pontuações.
* **Side Quests**: Missões paralelas oferecidas durante as trilhas que permitem aos usuários ganhar pontos extras e recompensas por completar tarefas, como tirar fotos de espécies locais ou responder quizzes.
* **Trilhas**: Percursos predefinidos em áreas naturais destinados à prática de ecoturismo, com diferentes níveis de dificuldade, permitindo a interação dos usuários com o ambiente natural