

# **Documento de Post-mortem**

**Nome do Projeto: NatureLens - Aplicativo de Identificação de Plantas e Animais**

## **Objetivos:**

**Analisar o ciclo de vida do projeto, identificando o que funcionou bem e os desafios encontrados.**

**Discutir lições aprendidas e ações de melhoria para futuros projetos.**

## **Revisão do Projeto/Eventos**

**Descrição: O NatureLens é um aplicativo que usa inteligência artificial e APIs de biodiversidade para identificar espécies de plantas e animais a partir de imagens enviadas pelos usuários.**

**Objetivos e Metas: Criar uma plataforma funcional e responsiva, permitindo que os usuários carreguem imagens e recebam informações detalhadas sobre as espécies, integrando IA através do Google Vision Pro e dados do GBIF.**

## **O que Funcionou Bem**

**Integração da API do Google Vision Pro: A integração com a API foi bem-sucedida, permitindo a análise de imagens e o retorno de informações precisas sobre plantas e animais.**

**Uso de Next.js para o Frontend: A escolha do Next.js facilitou o desenvolvimento de um frontend rápido e responsivo, além de simplificar a criação de rotas e páginas dinâmicas.**

**Autenticação com Google OAuth: A implementação da autenticação via Google proporcionou uma experiência de login simplificada para os usuários.**

## **O que Não Funcionou**

**Problema 1: Dificuldades para realizar testes no frontend devido à falta de familiaridade com ferramentas como Jest e Cypress.**

**Problema 2: Dificuldades para fazer o deploy da aplicação por falta de experiência em ferramentas como Vercel ou Heroku, além de limitações no uso do Git e GitHub.**

**Problema 3: A organização do repositório no GitHub ficou aquém do esperado, não seguindo as melhores práticas devido à falta de conhecimento na ferramenta.**

**Problema 4: O time teve dificuldades com a gestão do tempo, focando mais em outros projetos acadêmicos, o que impactou o progresso do NatureLens.**

## **Causas das Falhas**

**Causa 1: Falta de experiência com ferramentas de teste, como Jest e Cypress, dificultou a execução de testes automatizados no frontend.**

**Causa 2: Falta de familiaridade com Git e GitHub prejudicou a organização do código e o fluxo de trabalho em equipe, além de inviabilizar o deploy.**

**Causa 3: A gestão do tempo foi prejudicada pela prioridade dada a outros projetos, deixando o desenvolvimento do NatureLens em segundo plano.**

### **Ações de Melhoria**

**Ação 1: Realizar treinamentos e workshops sobre ferramentas de teste (Jest, Cypress) e deploy (Vercel, Heroku) para adquirir habilidades técnicas.**

**Ação 2: Estabelecer uma rotina de estudos e práticas sobre Git e GitHub para melhorar o controle de versão e colaboração em equipe.**

**Ação 3: Implementar uma melhor gestão de tempo no futuro, definindo prioridades claras e equilibrando as demandas de diferentes projetos.**

### **Lições Aprendidas**

**Lição 1: A importância de ter familiaridade com ferramentas de teste e deploy desde o início do projeto, para garantir que as fases finais sejam executadas com eficiência.**

**Lição 2: O uso adequado do Git e GitHub é essencial para a organização do código e o trabalho colaborativo, e**

**aprender essas ferramentas antes de iniciar o projeto teria evitado problemas.**

**Lição 3: A gestão do tempo é um fator determinante no sucesso de qualquer projeto. Ter um cronograma bem definido e respeitar prazos é essencial, especialmente em projetos acadêmicos que envolvem várias disciplinas.**

**Conclusão**

**Resumo: O projeto NatureLens foi uma experiência valiosa que proporcionou aprendizado em várias áreas, como integração de APIs, autenticação e uso de frameworks modernos. No entanto, a falta de experiência em testes, deploy e controle de versão resultou em alguns desafios que impactaram o cronograma do projeto.**