RELATORIO DE EXTENSÃO

* **1. Introdução  
  Este relatório documenta o processo de desenvolvimento do aplicativo móvel Guerra Reboque, criado para oferecer um serviço rápido e eficiente de solicitação de reboque de veículos. O principal objetivo do Guerra Reboque é conectar motoristas a prestadores de serviços de reboque por meio de uma plataforma digital, otimizar o tempo de resposta e facilitar a comunicação entre todas as partes envolvidas.**
* **O nome “Guerra Reboque” foi escolhido para transmitir uma ideia de resolução rápida e eficaz de situações de emergência no trânsito, como a necessidade urgente de um serviço de reboque, que é frequentemente associado a momentos de pressão e urgência.**
* **2. Objetivos do Projeto**
* **Desenvolver um aplicativo intuitivo e fácil de usar para motoristas e prestadores de serviços de reboque.**
* **Garantir que o processo de solicitação, localização, e pagamento fosse simples e rápido, minimizando a frustração dos usuários.**
* **Implementar funcionalidades como geolocalização em tempo real, integração com sistemas de pagamento digital, e notificações push para manter os usuários informados.**
* **Prover uma plataforma eficiente para a gestão dos atendimentos dos motoristas de reboque, permitindo um controle ágil das ordens de serviço e prazos de chegada.**
* **3. Planejamento e Definição de Requisitos  
  O desenvolvimento do Guerra Reboque foi precedido de um levantamento detalhado de requisitos, levando em consideração as necessidades de motoristas que precisavam de um reboque urgente e as necessidades dos prestadores de serviço para gerenciar esses chamados de forma eficiente. As funcionalidades e características mais importantes do aplicativo foram:**
* **Cadastro de Usuários: Permitir que motoristas e empresas de reboque se cadastrassem facilmente no sistema. Texto

  Descrição gerada automaticamente**

**Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente**

* **Notificações Push: Informar o status da solicitação, como a confirmação de recebimento, o tempo estimado de chegada, e outras atualizações em tempo real.**
* **Sistema de Pagamento: Integração com plataformas de pagamento como cartões de crédito e Pix para facilitar o pagamento diretamente pelo app.**
* **Avaliação e Feedback: Possibilitar que os usuários avaliem o serviço prestado, ajudando na manutenção da qualidade e confiabilidade da plataforma.**
* **4. Arquitetura e Tecnologias Utilizadas  
  Para garantir uma experiência fluida e multiplataforma, o Guerra Reboque foi desenvolvido com as seguintes tecnologias:**
* **Frontend: Flutter, uma framework de desenvolvimento móvel que permite a criação de um único código-base para Android e iOS, tornando o desenvolvimento mais rápido e eficiente.**
* **Backend: Node.js com o framework Express, para construção de APIs robustas e escaláveis, capazes de processar múltiplas solicitações simultâneas sem perda de performance.**
* **Banco de Dados: MongoDB, escolhido pela flexibilidade e escalabilidade, permitindo o armazenamento eficiente de dados de usuários, ordens de serviço, históricos de transações, e feedbacks.**
* **Geolocalização: API do Google Maps para localização precisa dos veículos e dos prestadores de serviço, além de otimizar rotas e tempos de resposta.**
* **Pagamentos: Integração com Stripe e PayPal, garantindo segurança e conveniência nos pagamentos dos serviços.**
* **Notificações Push: Firebase Cloud Messaging para envio de alertas em tempo real, mantendo os usuários sempre atualizados.**
* **5. Desenvolvimento e Desafios Enfrentados  
  O desenvolvimento do Guerra Reboque trouxe uma série de desafios técnicos e operacionais que precisaram ser superados para garantir que o aplicativo funcionasse da melhor forma possível:**
* **Geolocalização em Áreas Remotas: Um dos maiores desafios foi garantir que a geolocalização funcionasse bem, mesmo em áreas onde a conexão de dados é fraca ou inexistente. Para mitigar esse problema, investimos em otimizações no código e na utilização de múltiplas fontes de dados para melhorar a precisão da localização.**
* **Segurança no Pagamento: Implementar um sistema de pagamento digital seguro foi uma prioridade, pois lidamos com informações sensíveis dos usuários. O processo de integração com o Stripe e o PayPal foi cuidadosamente planejado para garantir que todas as transações fossem criptografadas e realizadas de maneira segura.**
* **Escalabilidade do Sistema: À medida que o número de usuários e pedidos aumentava, a infraestrutura de backend precisou ser ajustada para lidar com mais dados simultâneos. O MongoDB e a arquitetura de APIs escaláveis ajudaram a garantir que o sistema permanecesse rápido e confiável, mesmo em momentos de alto tráfego.**
* **6. Testes e Validação  
  A fase de testes foi essencial para garantir a qualidade do Guerra Reboque. Os testes foram realizados nas seguintes etapas:**
* **Testes Unitários: Focaram em validar cada módulo individualmente (por exemplo, geolocalização, pagamentos, notificações) para garantir que cada funcionalidade estivesse operando corretamente.**
* **Testes de Integração: Verificaram a interação entre os diversos componentes do sistema, como a comunicação entre o frontend e o backend, a precisão dos dados de geolocalização e a integração com os sistemas de pagamento.**
* **Testes de Usabilidade: Focaram na experiência do usuário, realizando testes com motoristas e clientes reais para verificar a facilidade de uso da interface, a clareza nas interações e o fluxo de navegação. Os feedbacks foram analisados para aprimorar a interface e a experiência geral.**
* **7. Lançamento e Monitoramento  
  Após o término dos testes, o Guerra Reboque foi lançado nas plataformas Google Play e Apple Store. Durante os primeiros dias após o lançamento, o aplicativo foi monitorado de perto para identificar e corrigir qualquer bug ou problema de desempenho. A equipe técnica ficou de plantão para oferecer suporte imediato e implementar correções rápidas.**
* **8. Resultados e Conclusões  
  O lançamento do Guerra Reboque foi bem recebido pelos usuários, com destaque para a interface intuitiva e a eficiência no processo de solicitação de reboque. Os motoristas elogiaram a rapidez na localização do prestador de serviços mais próximo e o sistema de pagamentos simples e seguro. As funcionalidades de notificação também foram muito apreciadas, pois mantiveram os usuários informados sobre o status da solicitação em tempo real.**
* **Os prestadores de serviços de reboque também se beneficiaram do aplicativo, com uma gestão mais eficiente dos chamados e um aumento na visibilidade de seus serviços.**
* **9. Próximos Passos e Melhorias  
  Embora o Guerra Reboque tenha sido um sucesso no lançamento, há várias melhorias planejadas para futuras versões:**
* **Expansão de Funcionalidades de Pagamento: Adicionar suporte para carteiras digitais como Google Pay, Apple Pay, e pagamento via QR Code.**
* **Otimização de Algoritmos de Roteamento: Aperfeiçoar a busca pelo reboque mais próximo, considerando mais variáveis como tráfego e condições da estrada.**
* **Suporte a Mais Idiomas e Regiões: Expandir o aplicativo para outras regiões, oferecendo suporte a múltiplos idiomas e ajustando os serviços às necessidades locais.**
* **O Guerra Reboque tem grande potencial de crescimento, e o feedback dos usuários será essencial para sua evolução contínua.**
* **Otimização da geolocalização em áreas de difícil acesso: Trabalhar na melhoria da precisão da localização em áreas mais remotas.**
* **Suporte a mais idiomas e locais: Expandir a disponibilidade do aplicativo para outras regiões e países.**

**O projeto foi um sucesso em termos de atender às necessidades dos usuários e fornecer uma solução moderna para serviços de reboque. O aplicativo continuará evoluindo conforme as demandas do mercado e os avanços tecnológicos.**

**:Plano de Projeto: Desenvolvimento de um Aplicativo para Reboque:**

**\*Objetivo do Projeto:\***

**Desenvolver um aplicativo para serviços de reboque, que permita aos usuários solicitar reboques de maneira eficiente e rápida, acompanhar o status do serviço e interagir com motoristas e prestadores de serviço.**

**\*Duração do Projeto:\***

**4 meses (16 semanas)**

**\*Participantes e Papéis:\***

**1. \*Líder de Projeto (ANDEW): Responsável por coordenar o progresso, definir as prioridades e garantir que o cronograma seja cumprido.**

**2. \*Product Owner (LUCAS): Responsável por definir os requisitos do aplicativo, priorizar as funcionalidades e garantir que o produto final atenda às necessidades dos usuários.**

**3. \*Desenvolvedor Front-end (ANDRÉ): Responsável pela interface do usuário (UI) e experiência do usuário (UX), garantindo um design intuitivo e atraente.**

**4. \*Desenvolvedor Back-end (ALYSSON): Responsável pela infraestrutura do servidor, banco de dados, APIs e lógica de negócios.**

**5. \*Especialista em QA (Quality Assurance - QA):(PEDRO) Responsável por testar o aplicativo em todas as etapas, identificar bugs e garantir que o produto seja funcional e seguro.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SEMANA** | **FASE** | **Líder de Projeto (ANDEW)** | **Product Owner (LUCAS)** | **Desenvolvedor Front-end (ANDRÉ)** | **Desenvolvedor Back-end (ALYSSON)** | **Especialista em QA (Quality Assurance) (PEDRO)** |
| **1-2** | **Planejamento Inicial e Levantamento de Requisitos** | Definir escopo e cronograma | Levantar e documentar requisitos funcionais e não funcionais | Analisar ideias de UI | Analisar requisitos técnicos para a infraestrutura | Revisar requisitos e planos de teste iniciais |
| **3-4** | **Design e Prototipagem** | Aprovar protótipos de alta fidelidade | Definir funcionalidades principais | Desenvolver Wire frames e Mockus | Iniciar estruturação do banco de dados e API | Planejar teste de usabilidade e de performance |
| **5-6** | **Desenvolvimento – Primeira Iteração** | Acompanhar o progresso e ajustar cronograma | Validar funcionalidades essenciais | Implementar o design de telas principais | Criar API de solicitação de reboque e integração com o banco de dados | Iniciar testes manuais de funcionalidades básicas |
| **7-8** | **Desenvolvimento – Primeira Iteração** | Acompanhar o progresso | Definir fluxos de usuários e priorizar features secundárias | Finalizar a integração de UI com APIs | Finalizar lógica do back-end (usuários, pagamento) | Testar funcionalidades implementadas, focando no fluxo principal |
| **9-10** | **Testes e Ajustes** | Ajustar cronograma conforme progresso | Validar e ajustar funcionalidades com base no feedback do PO | Ajustar UI com base nos testes | Corrigir bugs no back-end e integrar novas funcionalidades | Realizar testes de integração, usabilidade e bugs críticos |
| **11-12** | **Desenvolvimento – Iteração 2 (Novas Funcionalidades)** | Acompanhar progresso do desenvolvimento | Incluir funcionalidades adicionais (ex: chat com motorista, avaliação do serviço) | Implementar novas telas (ex: perfil do usuário, histórico de reboques) | Implementar novos serviços e funcionalidades no back-end (ex: notificações em tempo real) | Testar novas funcionalidades e corrigir erros encontrados |
| **13-14** | **Refinamento e Polimento Final** | | Validar o produto e organizar a entrega | Revisar e aprovar todas as funcionalidades | Polir design e usabilidade final, corrigir detalhes de UI | Realizar otimizações no back-end | Testar versões beta, garantir a estabilidade e desempenho do app |
| **15-16** | **Ajustes e Finalização do Projeto** | Planejar o lançamento e campanhas de marketing | Recolher feedback de usuários internos e externos | Ajustar UI para a versão final, testar em diferentes dispositivos | Finalizar configurações de servidor e banco de dados para produção | Testar a versão final em múltiplos dispositivos e cenários reais |

**:Fases do Projeto:**

**\*Fase 1: Planejamento Inicial e Levantamento de Requisitos (Semana 1–2)\***

**- Líder de Projeto (ANDREW):**

**- Definir o escopo, metas e cronograma do projeto.**

**- Coordenar a comunicação entre os membros da equipe.**

**- Product Owner (LUCAS):**

**- Levantar e documentar requisitos essenciais e complementares do aplicativo (ex: como o usuário solicita o reboque, rastreamento do status, integração com formas de pagamento).**

**- Priorizar funcionalidades para a primeira versão (MVP).**

**- Desenvolvedor Front-end (ANDRÉ):**

**- Analisar os requisitos de design do aplicativo e propor ideias para a UI/UX.**

**- Desenvolvedor Back-end (ALYSSON):**

**- Analisar os requisitos técnicos e definir a arquitetura do servidor e banco de dados.**

**- Especialista em QA (PEDRO):**

**- Revisar os requisitos e começar a planejar os testes.**

**\*Fase 2: Design e Prototipagem (Semana 3–4)\***

**- ANDREW:**

**- Acompanhar o progresso do design e garantir que esteja alinhado com os requisitos.**

**- LUCAS:**

**- Validar as funcionalidades e fluxos do aplicativo.**

**- ANDRÉ:**

**- Criar wireframes e mockups de telas principais (ex: tela de solicitação de reboque, tela de rastreamento).**

**- ALYSSON:**

**- Iniciar a modelagem do banco de dados e planejamento da API.**

**- PEDRO:**

**- Planejar os testes de usabilidade e performance.**

**\*Fase 3: Desenvolvimento – Primeira Iteração (MVP) (Semana 5–8)\***

**- ANDREW:**

**- Garantir que o projeto siga dentro do cronograma e faça ajustes conforme necessário.**

**- LUCAS:**

**- Validar funcionalidades com base em prioridades.**

**- ANDRÉ:**

**- Implementar o design das telas principais e integrar com as APIs do back-end.**

**- ALYSSON:**

**- Desenvolver a API de solicitação de reboque e integração com o banco de dados.**

**- PEDRO:**

**- Realizar testes manuais e funcionais, focando nas funcionalidades básicas.**

**\*Fase 4: Testes e Ajustes – MVP (Semana 9–10)\***

**- ANDREW:**

**- Acompanhar o progresso dos testes e ajustes necessários.**

**- LUCAS:**

**- Validar e ajustar funcionalidades com base no feedback recebido.**

**- ANDRÉ:**

**- Ajustar o design com base nos testes e feedbacks.**

**- \*ALYSSON:\***

**- Corrigir problemas no back-end e integrar novas funcionalidades.**

**- PEDRO:**

**- Testar o MVP completo e corrigir erros encontrados.**

**\*Fase 5: Desenvolvimento – Iteração 2 (Novas Funcionalidades) (Semana 11–12)\***

**- ANDREW:**

**- Acompanhar o progresso das funcionalidades extras e garantir que o cronograma esteja sendo seguido.**

**- LUCAS:**

**- Validar e priorizar novas funcionalidades (ex: chat com motorista, avaliações).**

**- ANDRÉ:**

**- Implementar novas telas e funcionalidades baseadas nas necessidades do usuário.**

**- ALYSSON:**

**- Desenvolver e integrar novos serviços, como notificações em tempo real e atualizações de status.**

**- PEDRO:**

**- Testar novas funcionalidades e garantir que não haja regressões.**

**\*Fase 6: Refinamento e Polimento Final (Semana 13–14)\***

**- ANDREW:**

**- Validar que todas as funcionalidades essenciais estão implementadas e funcionando corretamente.**

**- LUCAS:**

**- Validar e garantir que o aplicativo atenda a todos os requisitos do usuário.**

**- ANDRÉ:**

**- Finalizar o design para a versão final e corrigir problemas de usabilidade.**

**- ALYSSON:**

**- Realizar otimizações no back-end.**

**- PEDRO:**

**- Testar versões beta e garantir que o aplicativo seja estável e de alto desempenho.**

**\*Fase 7: Preparação para Lançamento (Semana 15–16)\***

**- ANDREW:**

**- Planejar o lançamento e as campanhas de marketing.**

**- LUCAS:**

**- Recolher feedback de usuários internos e externos para ajustes finais.**

**- ANDRÉ:**

**- Ajustar a UI para diferentes dispositivos e telas.**

**- ALYSSON:**

**- Preparar o servidor e o banco de dados para o lançamento.**

**- PEDRO:**

**- Testar a versão final em múltiplos dispositivos e cenários reais.**

**Riscos e Mitigações:**

**Atrasos no desenvolvimento:**

**- Mitigação: Manter um cronograma detalhado e revisões semanais para ajustes.**

**Problemas de integração entre front-end e back-end\*:**

**- Mitigação: Realizar testes contínuos.**

|  |
| --- |
| **Grupo de Alunos** |
| **< André Dos Reis Figueiredo >**  **Matrícula: < 202304328731 >** |
| **< Andrew Guedes Reis >**  **Matrícula: <202309136007>** |
| **<Alysson Saraiva Barros>**  **Matrícula: <202309135981>** |
| **< Lucas Straub Fernandes Silva >**  **Matrícula: < 202309137216 >** |
| **<Pedro Luiz Nascimento Dos Santos>**  **Matrícula: <202308602276 >** |