

Plano de Desenvolvimento do Sistema

Plataforma de Aluguel de Carros para Motoristas de App

1. Metodologia de Desenvolvimento

- Utilizar metodologia ágil (ex: Scrum ou Kanban) com sprints curtas e entregas frequentes.
- Reuniões de alinhamento a cada início de sprint.
- Registro e acompanhamento de tarefas via ferramenta (ex: Trello, Jira, GitHub Projects).

2. Fases do Projeto

2.1. Levantamento de Requisitos

- Refinar escopo e detalhar regras de negócio junto ao cliente (você).
- Criar documentos de requisitos funcionais e não funcionais.

2.2. Arquitetura e Tecnologias

- Definir arquitetura da solução (ex: monolito ou microserviços).
- Tecnologias sugeridas:
 - Backend: Node.js + Express
 - Frontend: React (web app responsivo)
 - Banco de Dados: MySQL ou PostgreSQL
 - Armazenamento de arquivos: Amazon S3 ou local (em homologação)
 - Integrações: APIs para assinatura eletrônica, gov.br, pagamentos, notificações

2.3. Modelagem de Dados

- Criar modelo ER (Entidade-Relacionamento) do banco de dados
- Definir tabelas, relacionamentos e constraints

2.4. Prototipação

- Criar wireframes/telas básicas das principais páginas (ex: cadastro, login, listagem de carros)
- Validar layout e usabilidade antes de iniciar implementação

2.5. Implementação

- Desenvolvimento dividido por módulos:
 - Autenticação de usuários
 - Upload e análise de documentos
 - Cadastro e gestão da frota
 - Solicitação, aprovação e assinatura de contratos
 - Relatórios e painel administrativo
- Versionamento de código no GitHub

2.6. Testes

- Testes unitários (backend e frontend)
- Testes de integração
- Testes de usabilidade e validação com usuário

2.7. Implantação

- Homologação em ambiente de testes (ex: servidor cloud ou local)
- Ajustes finais e migração para produção

2.8. Treinamento e Suporte

- Criação de manuais rápidos para motoristas e admin
- Suporte inicial no pós-lançamento

3. Padrões de Código e Documentação

- Seguir boas práticas de clean code
- Documentação das APIs (Swagger ou similar)
- Comentários claros no código e READMEs em todos os repositórios

4. Segurança

- Criptografia de senhas (bcrypt)
- Proteção de rotas (JWT)
- Validação de uploads
- Backup e políticas de recuperação

5. Cronograma Inicial Sugerido

- Semana 1: Levantamento, prototipação e arquitetura
- Semana 2-4: Implementação dos principais módulos
- Semana 5: Testes e homologação
- Semana 6: Implantação e treinamento