
Especificação de Requisitos de Software

para

Mashin

Versão 1.0

**Por Bruno Gomes Mortella
Júlia Vitória Rezende Paiva**

Grupo Faz o While

23 de setembro de 2019

Sumário

Sumário	ii
1. Introdução	1
1.1 Descrição do produto	1
1.2 Público Alvo e Sugestões de leitura.....	1
1.3 Escopo.....	1
1.4 Referências.....	1
2. Descrição geral	2
2.1 Visão do Produto	2
2.2 Função do Produto	2
2.3 Classes de usuários	3
2.4 Plataformas	3
2.5 Infraestrutura de apoio	3
2.6 Premissas	3
2.7 Equipe	3
3. Requisitos	4
3.1 Cadastro de usuários	4
3.2 Mapa de aluguel.....	4
3.3 Página de aluguel	5
3.4 Avaliação durante aluguel.....	5
3.5 Perfil do usuário.....	6
3.4 Página de carona	6
4. Plano de Projeto	7
4.1 Modelo Espiral.....	7
4.2 Grupos de entrega	7
4.3 Linha do tempo	7
4. Projeto de Arquitetura	7
4. Implantação	7

1. Introdução

1.1 Descrição do produto

Mashin é um aplicativo de aluguel de carros elétricos com sistema de carona integrado, que diminuirá o preço do aluguel para o cliente que está alugando. Os veículos estarão disponíveis em estacionamentos comerciais, nas ruas e futuramente em condomínios residenciais. A empresa futuramente investirá em pontos de recarga nos locais de aluguel.

1.2 Público alvo e Sugestões de leitura

Este documento é destinado para todos os integrantes da equipe de desenvolvimento, pode ser consultado também por integrantes da equipe de marketing, além dos idealizadores do projeto. O documento tem como objetivo detalhar o produto e o processo de seu desenvolvimento afim de agilizar e manter qualidade da produção.

1.3 Escopo

O software deve ser capaz de prover o serviço de aluguel Mashin com uma interface intuitiva, contendo um mapa da região em que o cliente se localiza, além de conter uma página de carona, uma página para o perfil do cliente, na qual ele pode verificar seus dados pessoais e histórico de alugueis e uma página para realizar o pagamento.

1.4 Referências

LOGTENBERG, Ryan. et al. Comparing Fuel and Maintenance Costs of Electric and Gas Powered Vehicles in Canada. 2018. 23f. Pesquisa – 2º Institute. Disponível em:<https://www.2degreesinstitute.org/reports/comparing_fuel_and_maintenance_costs_of_electric_and_gas_powered_vehicles_in_canada.pdf>. Acesso em 20 set.2019

PRESSMAN, R.S.; MAXIM, B.R.; Engenharia de Software, uma abordagem profissional. 8ª ed. Porto Alegre.

2. Descrição geral

2.1 Visão do Produto

O Mashin tem como origem a ideia de facilitar e baratear os aluguéis de carros elétricos. Muitas pessoas acabam optando por aluguel, devido aos altos impostos, ademais, o Mashin também impactaria no preço do combustível, já que a gasolina tem um custo muito maior do que a carga elétrica. Também é preciso considerar que os custos para manter um carro elétrico operando é menos da metade que os custos de um carro movido à gasolina, levando em conta a manutenção.

2.2 Função do Produto

Usando o aplicativo Mashin o cliente deverá ser capaz de alugar carros ou pedir caronas, além de consultar seus dados pessoais, histórico de aluguel, e através do mapa, identificar zonas de uso e postos de aluguel.

Também será possível realizar o cadastro através do site.

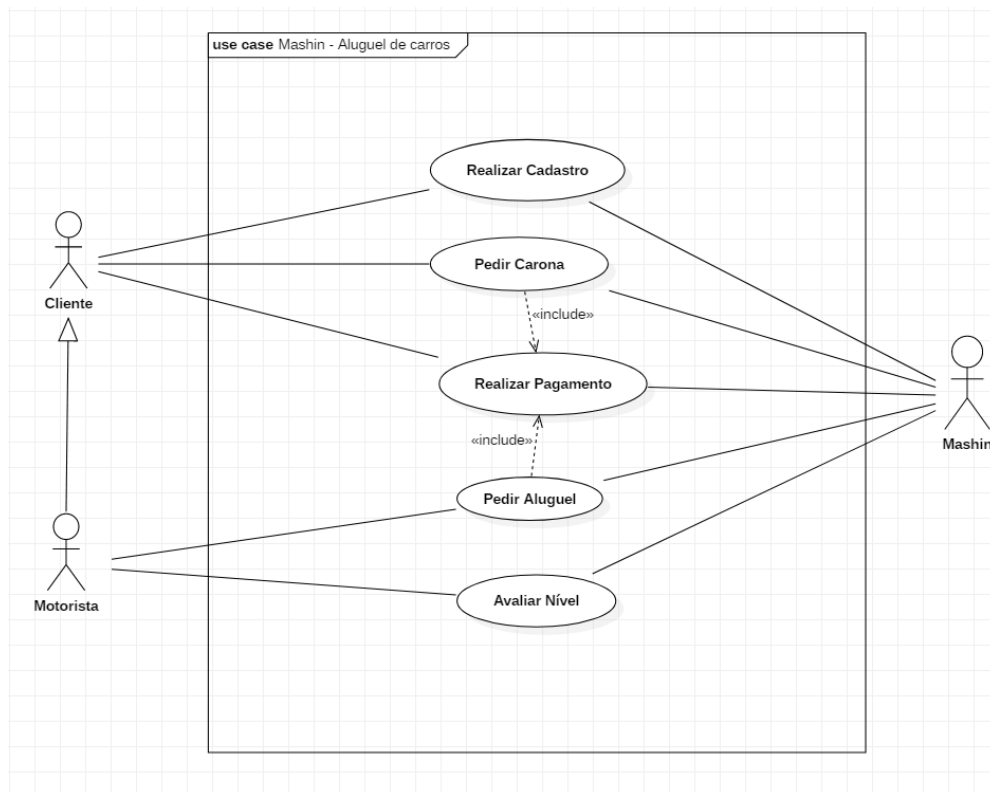


Figura 1: Diagrama de caso de uso para o aplicativo Mashin

2.3 Classes de usuários

Cliente Motorista – pessoas acima de vinte anos de idade, que possuem carteira de motorista com menos de 10 pontos. Procura uma forma simples e rápida de usar um carro, mas sem arcar com os custos de impostos e manutenção.

Cliente Carona – pessoas acima de dezesseis anos de idade, não necessitam de carteira de motorista. Procura um transporte de baixo custo e de qualidade.

2.4 Plataforma

Navegadores e smartphones com sistemas operacionais Android e iOS.

2.5 Infraestrutura de apoio

Ambiente de desenvolvimento Linux, ferramenta Star UML para criação de modelos, Visual Studio Code, React Native, Django, banco de dados PostgreSQL, controle de versão com Git, deploy e testes com Jenkins, gerência de projeto com Artia e comunicação com Slack.

2.6 Premissas

Durante o desenvolvimento do projeto deve ser levado em consideração os preços de aluguel estabelecidos, as zonas de uso em que os clientes podem circular com os carros e os pontos de abastecimento. Além disso, a avaliação da carteira de motorista dependerá de consultas à base de dados do governo.

2.7 Equipe

A equipe será composta por quatro desenvolvedores full-stack, com competência em desenvolver APIs e interfaces, que serão responsáveis por criar o software de aluguel e integra-lo ao software do cliente. A equipe também contará com dois desenvolvedores mobile, capazes de desenvolver apps para diferentes plataformas e que serão responsáveis por criar um app de fácil uso, além de um especialista em análise de risco, capaz de identificar possíveis falhas no projeto, dois especialistas em segurança, capazes de identificar falhas e realizar testes para garantir a segurança do software, e por fim, um especialista de banco de dados para criar e garantir manutenção do banco de dados.

3. Requisitos

3.1 Cadastro de usuários

3.1.1 Descrição e Prioridade

Tem como objetivo recolher os dados do cliente para realizar o cadastro no sistema, no caso de clientes que se cadastram como motoristas, estes terão a CNH avaliada pelo sistema para classificar como aptos ou não para realizar aluguéis. Prioridade nível 1 de 6.

3.1.2 Requisitos funcionais

REQ-1. Análise de usuários motoristas: verificar se possui carteira de motorista válida.

REQ-2. Usuários sem carteira de motorista ou com muitos pontos não deverão ser classificados como motoristas.

REQ-3. Classificar o nível de risco a partir dos pontos.

3.1.2 Requisitos não-funcionais

- A Carteira de motorista deve ser verificada em um banco de dados do governo.
- Análise de usuários motoristas não deverá ultrapassar 2 dias úteis.

3.2 Mapa de aluguel

3.2.1 Descrição e Prioridade

Tem como objetivo mostrar os carros próximos ao cliente que estão disponíveis para aluguel. Prioridade nível 2 de 6.

3.2.2 Requisitos funcionais

REQ-1. Mostrar carros disponíveis

RE1-2. Mostrar zonas de uso

3.2.2 Requisitos não-funcionais

- Manter sempre atualizado
- Mostrar carros em um raio de 800m do usuário

3.3 Página de aluguel

3.3.1 Descrição e Prioridade

Tem como objetivo realizar o aluguel do carro, incluindo o pagamento. Prioridade nível 3 de 6.

3.3.2 Requisitos funcionais

REQ-1. Scanner para código do carro

REQ-2. Cálculo de preço do aluguel a partir da classificação do motorista

REQ-3. Confirmação do pagamento deverá aparecer na tela para o usuário

3.2.2 Requisitos não-funcionais

- Confirmação do pagamento deverá demorar até 30 segundos
- Desbloquear o carro após a confirmação do aluguel
- Informações de aluguel deverão ser armazenadas durante um ano

3.4 Avaliação durante aluguel

3.4.1 Descrição e Prioridade

Tem como objetivo monitorar o usuário durante o aluguel. Prioridade nível 4 de 6.

3.4.2 Requisitos funcionais

REQ-1. Multar o usuário caso a zona de uso for ultrapassada

REQ-2. Multar o usuário caso o carro não for entregue nos locais determinados

REQ-3. Caso tempo limite de aluguel for ultrapassado, deverá ser cobrado por ter de uso extra

3.4.2 Requisitos não-funcionais

- Carros deverão ser rastreados 24H

3.5 Perfil do usuário

3.5.1 Descrição e Prioridade

Tem como objetivo mostrar os dados do usuário e histórico de aluguel. Prioridade nível 5 de 6.

3.5.2 Requisitos funcionais

REQ-1. Dados pessoais

REQ-2. Nível de risco

REQ-3. Carros alugados

REQ-4. Pagamentos realizados

3.6 Página de carona

3.6.1 Descrição e Prioridade

Tem como objetivo disponibilizar o serviço de carona. Prioridade nível 6 de 6.

3.6.2 Requisitos funcionais

REQ-1. Garantir opções de carona para destinos próximos, definindo um raio de distância, tanto para partida, quanto para chegada

REQ-2. Calcular quantos % do valor do aluguel o carona deverá pagar

3.6.2 Requisitos não-funcionais

- Raio para partida deverá ser de 800km
- Raio para chegada deverá ser de 3km
- O cálculo para pagamento de carona deverá ser baseado no tempo determinado durante o aluguel, sem incluir multas ou cobranças extras por tempo de uso

4. Plano de Projeto

4.1 Modelo Espiral

Para o desenvolvimento deste projeto, será adotado o modelo Espiral, criado por Barry Bohem, e que consiste em dividir as etapas do projeto em ciclos que iniciam com um planejamento onde são determinados os objetivos, soluções de problemas e restrições para o ciclo atual, seguido por uma análise de riscos, desenvolvimento e por fim uma revisão das etapas anteriores e o planejamento para o novo ciclo. Este modelo traz como vantagem a versatilidade para lidar com mudanças, além de estimativas realísticas e melhor tempo de implementação, além de que a equipe irá adquirir mais conhecimento sobre o projeto durante o desenvolvimento.

4.1.1 Ciclos

Em cada ciclo será implementado dois macro requisitos da lista citada anteriormente (tópico 3), iniciando com cadastro de usuários e finalizando com a página de carona.

4.2 Grupos de entregas

O projeto será dividido em duas etapas, ou seja, com dois grupos de entrega, seguindo a ordem de prioridade dos requisitos. O primeiro grupo de entrega será composto pela implementação do sistema de aluguel, sendo composto por 5 requisitos principais: cadastro de usuários, mapa de aluguel, página de aluguel, avaliação durante aluguel e página de usuário. Já o segundo grupo de entrega será composto pela implementação do sistema de carona, sendo este o requisito principal.

4.3 Linha do tempo

O projeto conta com os dois grupos de entrega, sendo que o primeiro grupo do sistema de aluguel levará 1050 horas para ser finalizado ou aproximadamente 6 meses e o segundo grupo do sistema de carona levará 250 horas ou aproximadamente 1 mês e meio.

5. Projeto de arquitetura

O projeto seguirá o padrão de arquitetura de microsserviços, ou seja, os requerimentos serão implementados em formatos de serviços modulares pequenos e independentes que executam tarefas exclusivas.

6. Implantação

Ocorrerão duas fases de implantação, a primeira será ao final do primeiro grupo de entrega onde uma primeira versão executável do software será disponibilizada nas lojas de aplicativos mobile, logo a instalação e desinstalação fica sobre controle do cliente e as atualizações ficarão disponíveis nas lojas. A segunda fase, onde o sistema de carona será adicionado ocorrerá através de uma atualização.