

# Arquitetura de Software

Alunos: André kovalski, Gabrielle Louise e Leonardo Knight.

# TRANSPORTE



## BIG IDEA

Transporte

## ESSENTIAL QUESTION

Como facilitar a busca  
de serviços de  
transporte para o  
estudante?

## CHALLENGE

Conectar usuários  
(estudantes,  
trabalhadores, etc) e  
motoristas.

# Questões Guias



- Qual a importancia do software? (Domínio, prioridade média)

Conectar passageiros e motoristas com a finalidade de facilitar a comunicação entre ambos para que passageiros possam encontrar o melhor serviço.

- Em qual formato será o software? (mobile, web)  
(Funcionalidade, prioridade alta)

Já que a pessoa deverá conseguir acessar o serviço em qualquer lugar, um software mobile seria o ideal, mas uma página web também pode ser bom para o usuário

- Como saber se quem está digirindo, é o motorista aprovado?  
(Funcionalidade, prioridade média)

Fazendo um sistema de check-in onde o motorista deverá tirar uma foto de si e um sistema com reconhecimento facial detecta se é ou não o motorista cadastrado

- Como garantir a qualidade do serviço ao passageiro?

(Funcionalidade, prioridade média)

Através de um sistema de feedback do motorista, na qual a exposição dele ao público é dita de acordo com os feedbacks dos passageiros

- Qual o protocolo em caso de perdas de objetos dentro do veículo? (Domínio, prioridade baixa)

O passageiro poderá entrar em contato com o motorista para que ele esteja ciente do acontecimento e poderá entregar o objeto de volta

- Vamos continuar? Temos conhecimento suficiente?

Iremos continuar, sim

# Esboço da solução

## **Esboço da solução:**

Um aplicativo móvel para o usuário e para o motorista que estará prestando o serviço, o cliente pode ver todas as opções possíveis de transporte que pode ser contratado baseado na sua rota, contratar esses serviços e modificar suas contratações, observar os dados. E o motorista pode fazer publicação de seu serviço, receber pagamentos pelo app, e receber informações sobre seus clientes.

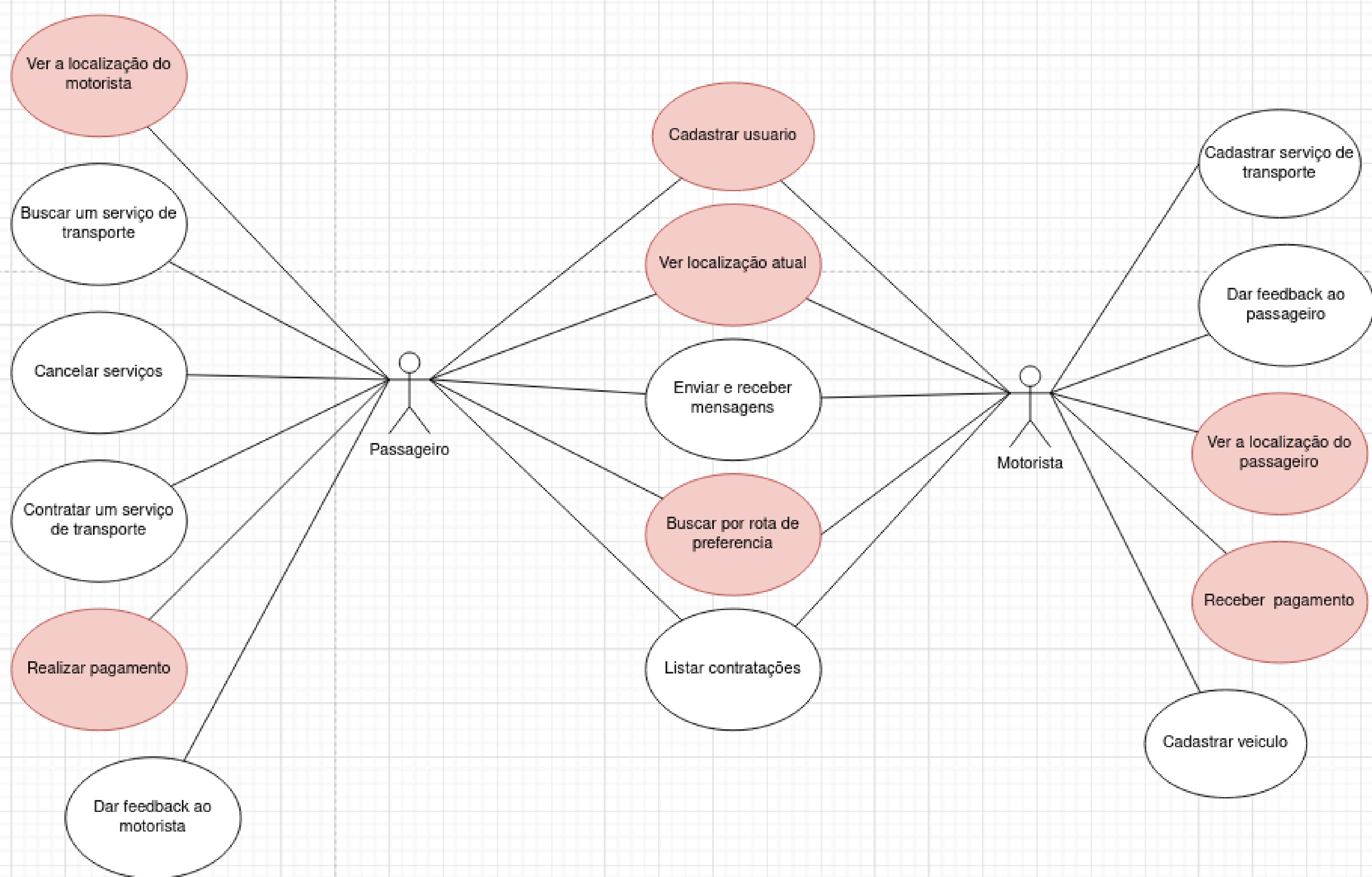
## Requisitos funcionais:

- Mapa mostrando localização do usuário e do prestador de serviço
- Maneira fácil de encontrar serviços de van em um estilo de marketplace, e realizar pagamentos
- Enviar/receber mensagens diretas
- Rota para o motorista chegar no ponto de encontro dos passageiros e até o destino final e vice-versa
- Fácil maneira de um motorista encontrar clientes e receber pagamentos



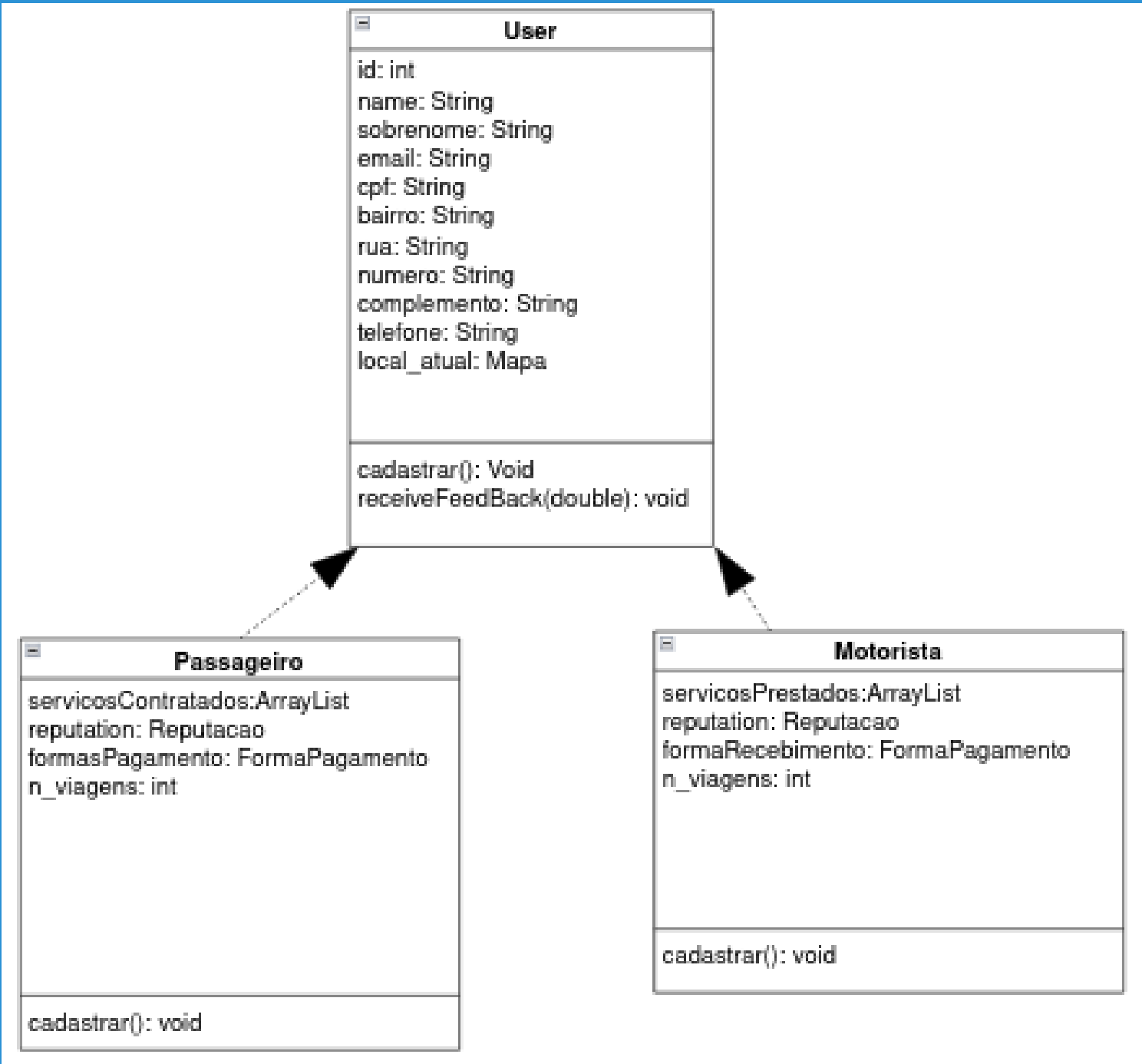
## Requisitos não funcionais:

- Sistema confiável e adaptável para manter as informações dos usuários com segurança
- Qual o limite para acesso simultâneos na API de mapa, ter confiabilidade
- Qual o método para processar pagamentos e como garantir sua segurança e confiabilidade
- Facilidade para usar o app

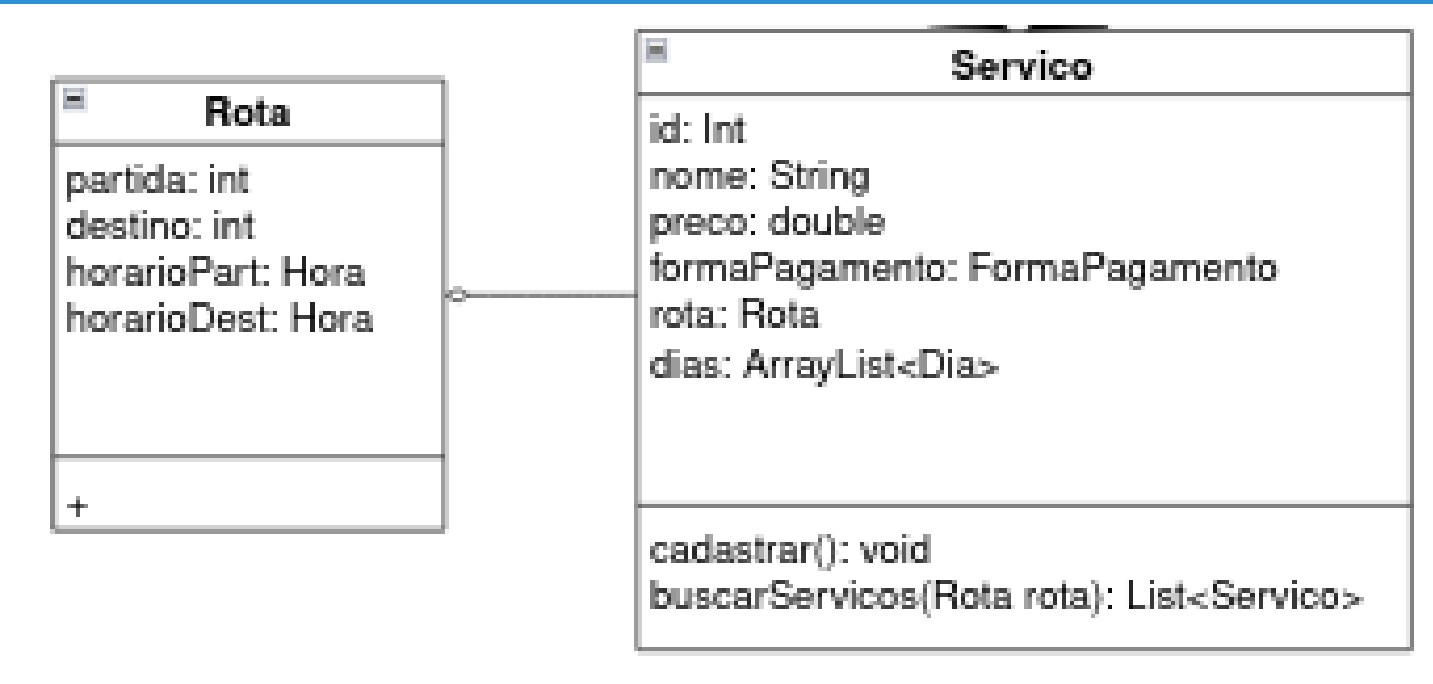


# **Diagrama de classe preliminar e protótipo de telas**

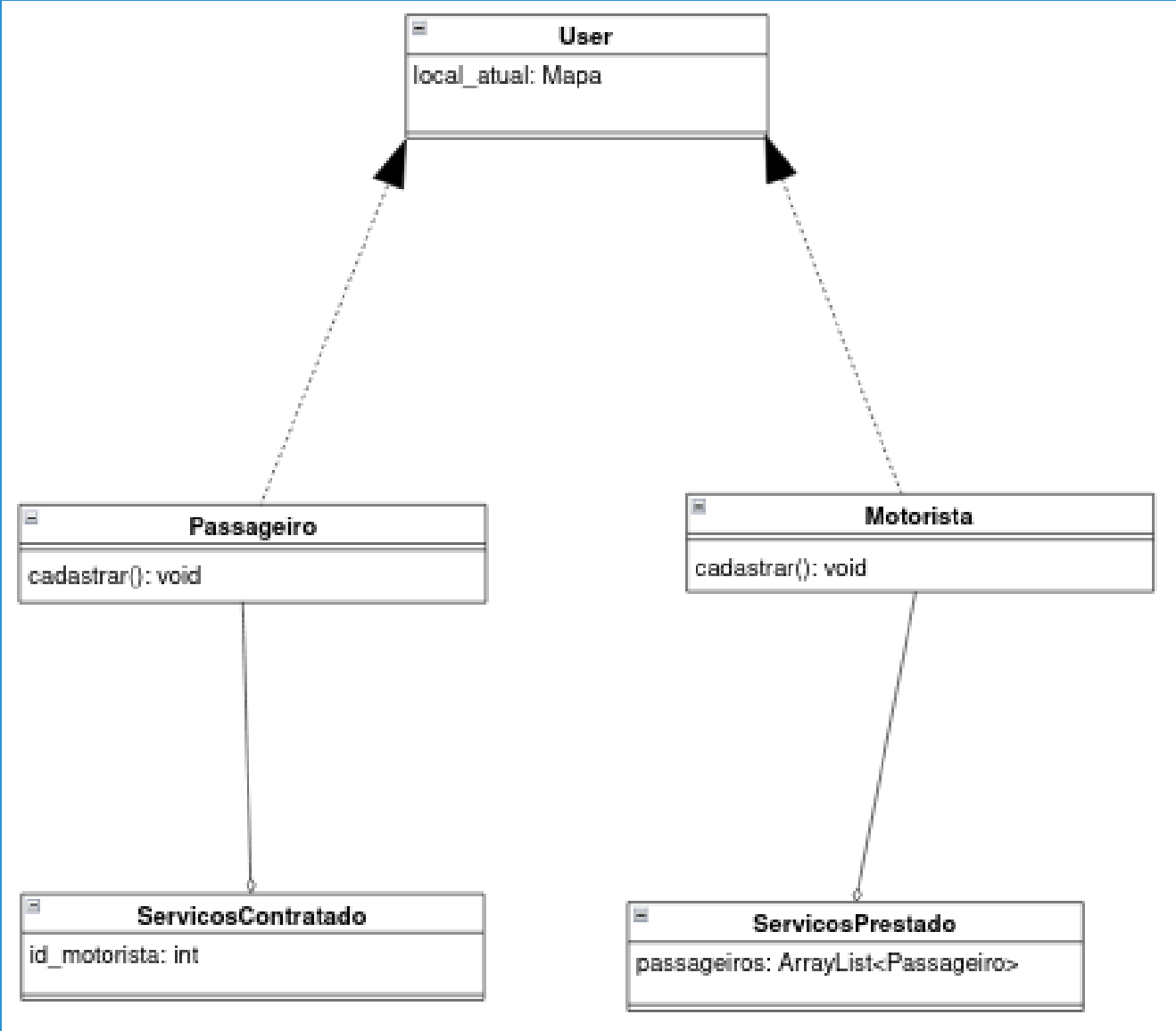
Cadastrar Usuário:



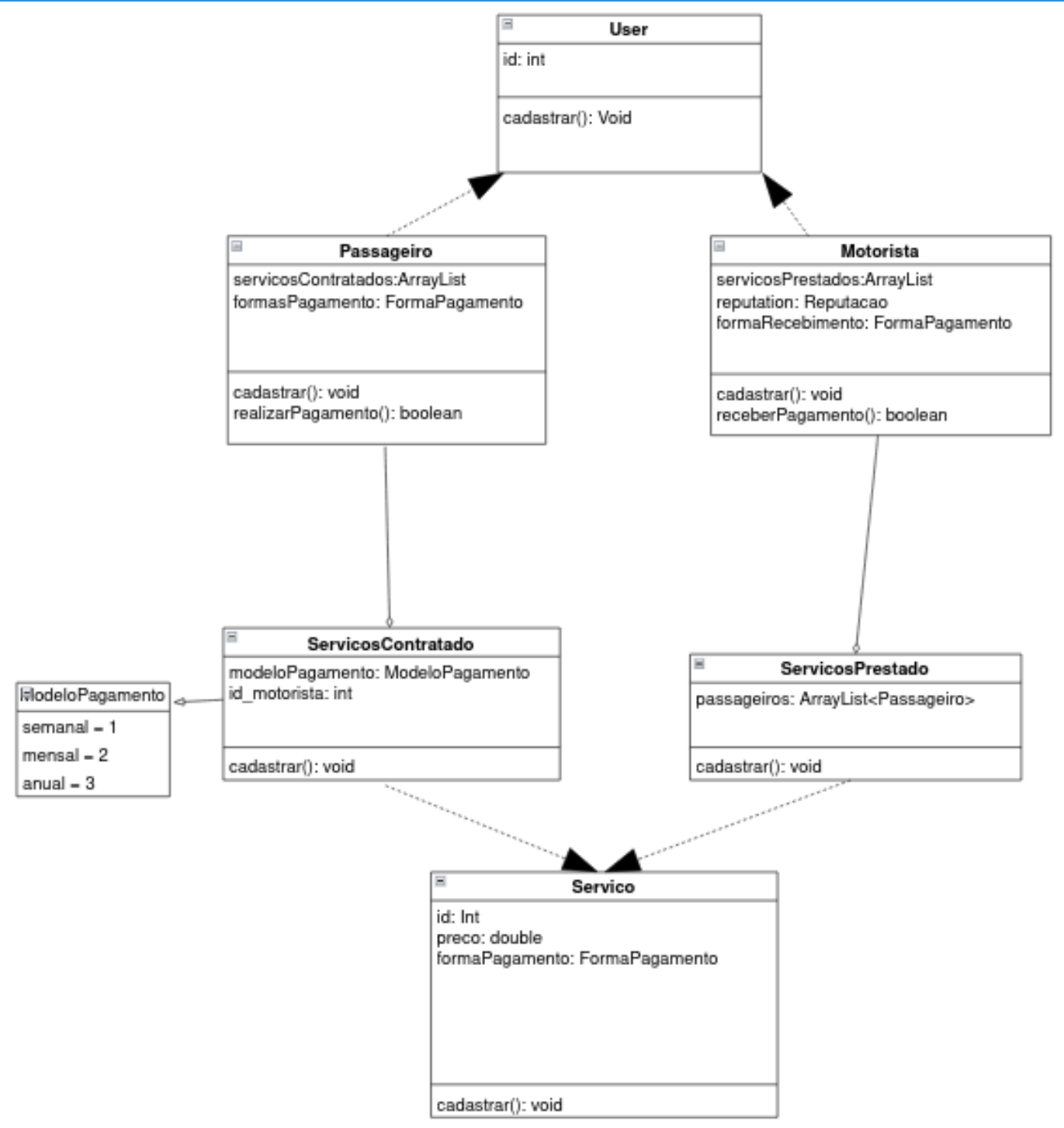
Buscar por rota de preferência:

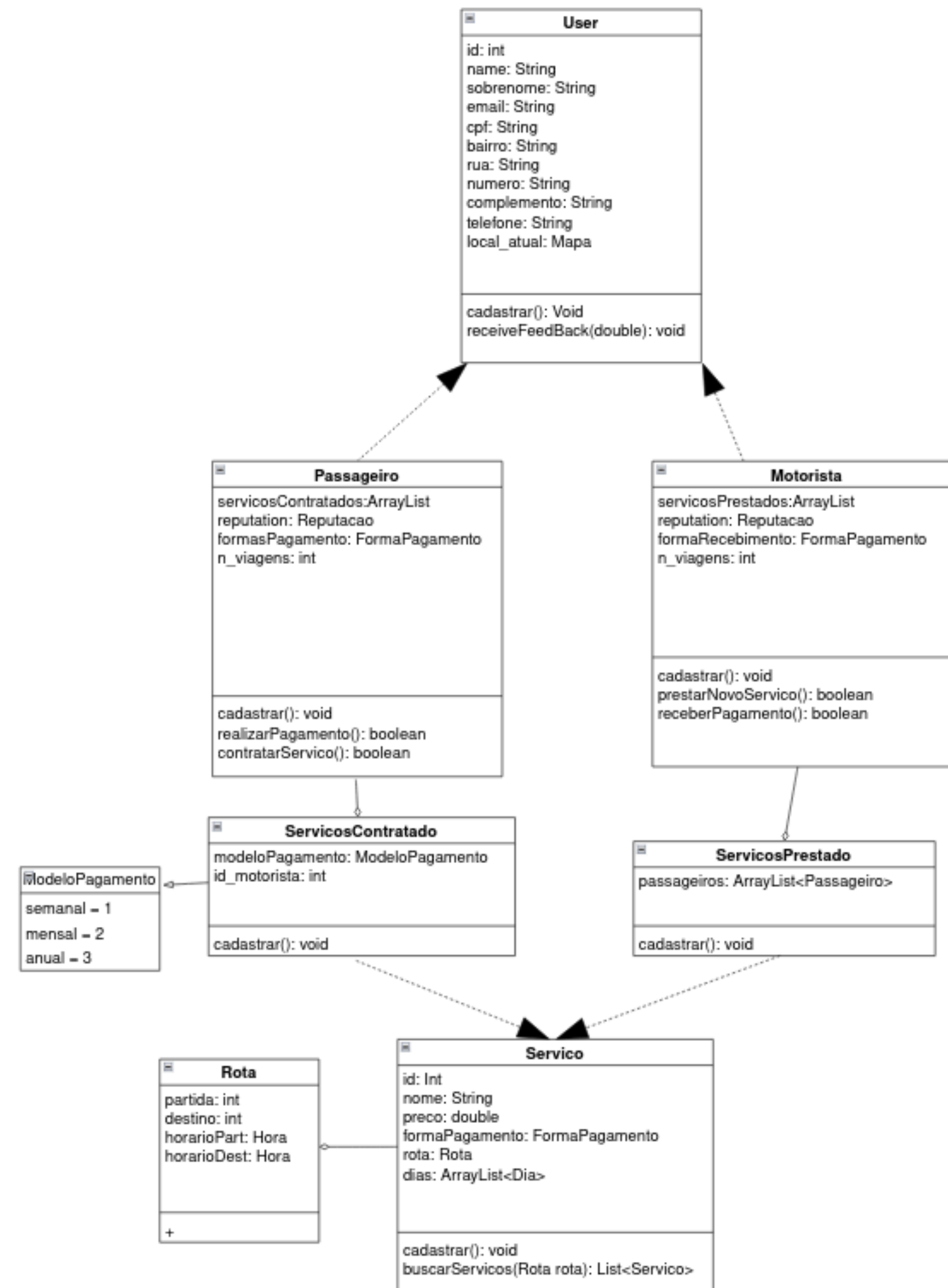


Ver localização do motorista/passageiro:



Realizar/receber pagamento:





# Protótipo das telas

Android Small - 1

Cadastrar Usuário

Nome

Sobrenome

Email

CPF

Bairro

Rua

Número

Complemento

Telefone

Salvar

Android Small - 2

Menu

Definir Rota

Ponto Partida

Ponto Destino

Hora Partida

00:00

Hora Busca

00:00

Buscar

Motorista	Veículo	Preço	
Motorista1	Veículo	R\$ 100,00	Detalhes
Motorista2	Veículo	R\$ 200,00	Detalhes
Motorista3	Veículo	R\$ 150,00	Detalhes

Android Small - 3

Menu

Definir Rota

Ponto Partida

Selecionar Região

Ponto Destino

Selecionar região

Hora Partida

00:00

Hora Busca

00:00

Buscar

Passageiro	Partida	Destino	
Passageiro1	Partida	Destino	Detalhes
Passageiro2	Partida	Destino	Detalhes
Passageiro3	Partida	Destino	Detalhes

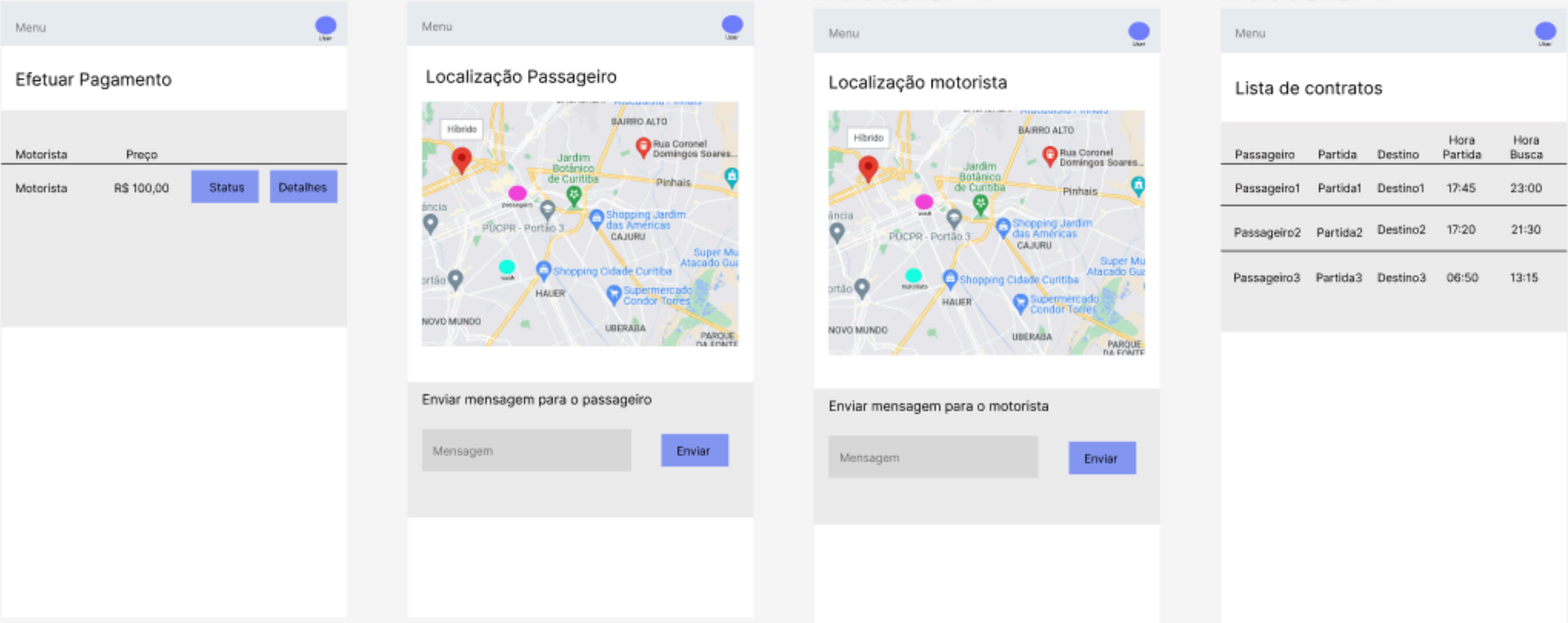
Android Small - 4

Menu

Receber Pagamento

Passageiro	Preço		
Passageiro1	R\$ 100,00	Status	Detalhes
Passageiro2	R\$ 200,00	Status	Detalhes
Passageiro3	R\$ 150,00	Status	Detalhes

https://www.figma.com/file/DEgjksUuASNHO4Sl0NRmyG/Untitled?node-id=0%3A1&t=Vml6gGKqXQdD2opO-1

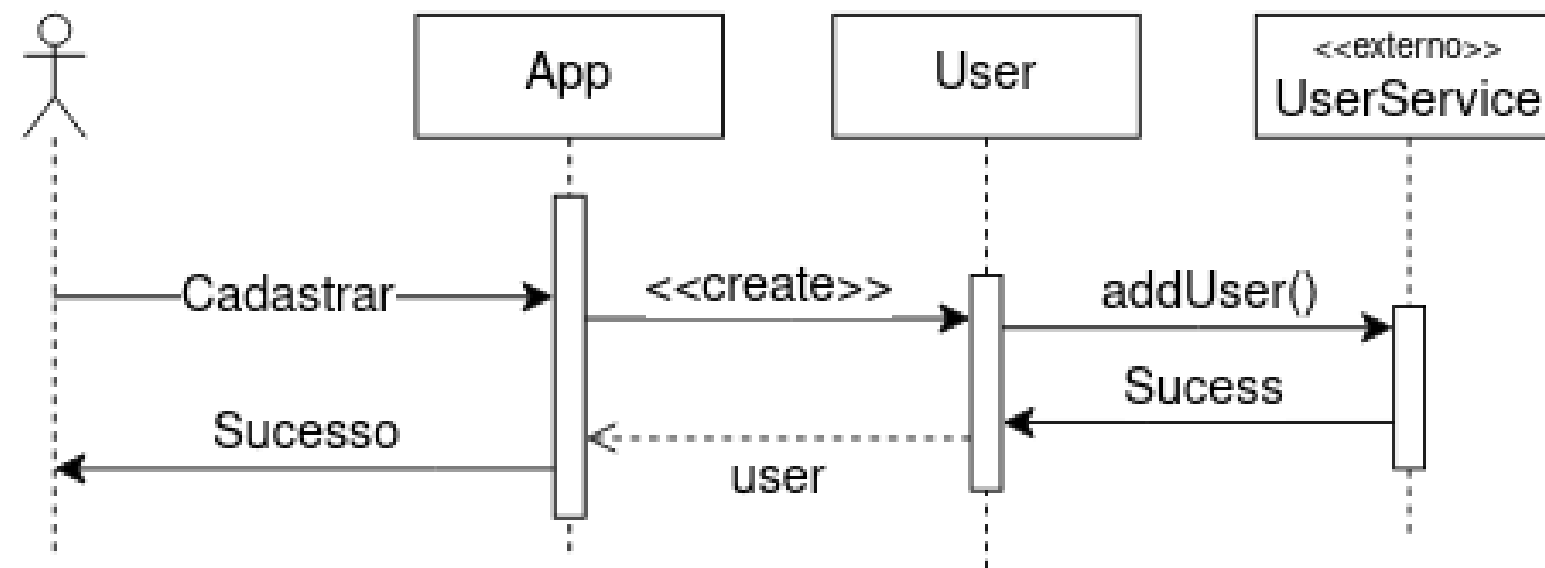




# Diagramas de Sequência

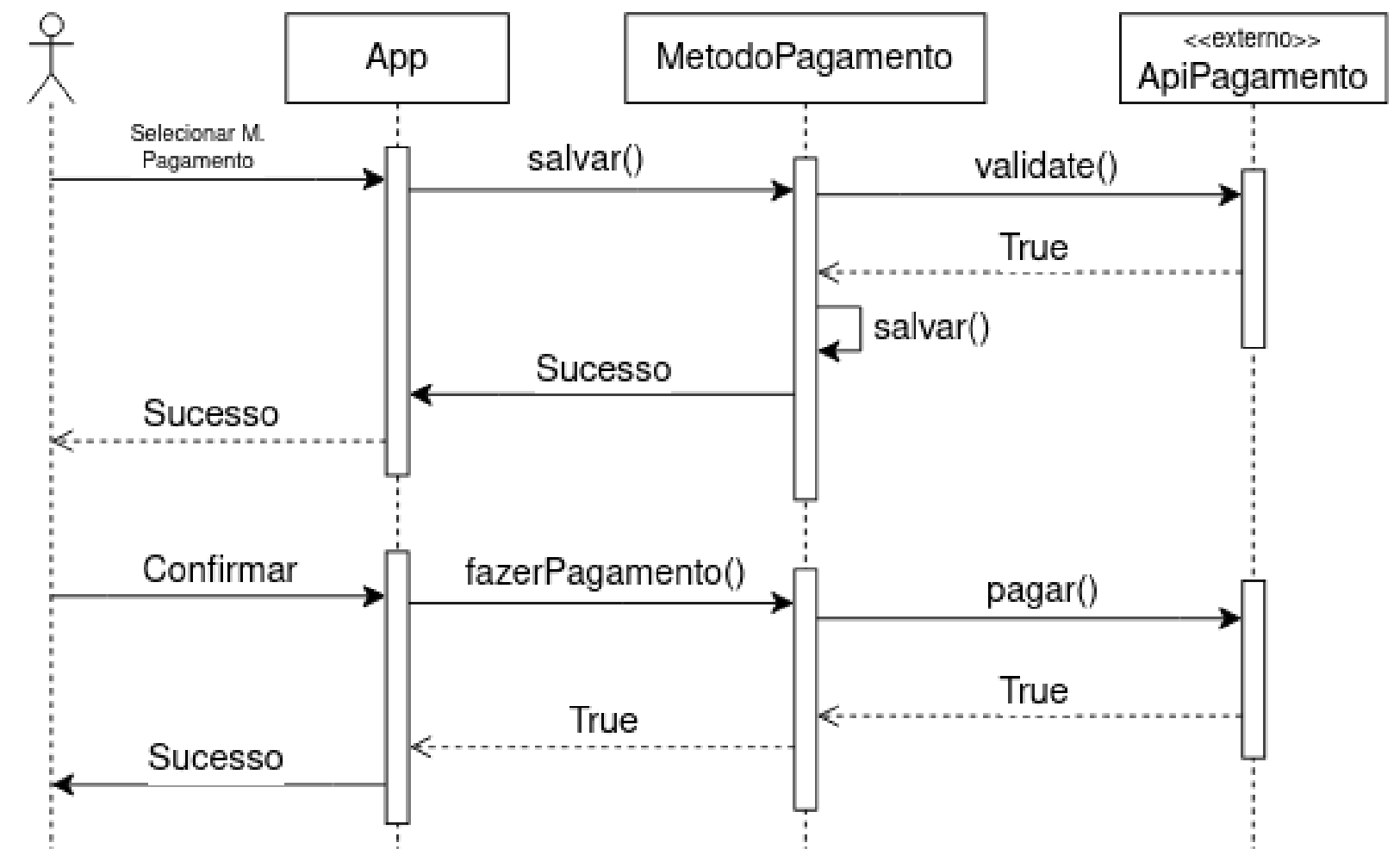
## Cadastrar usuário

Usuário

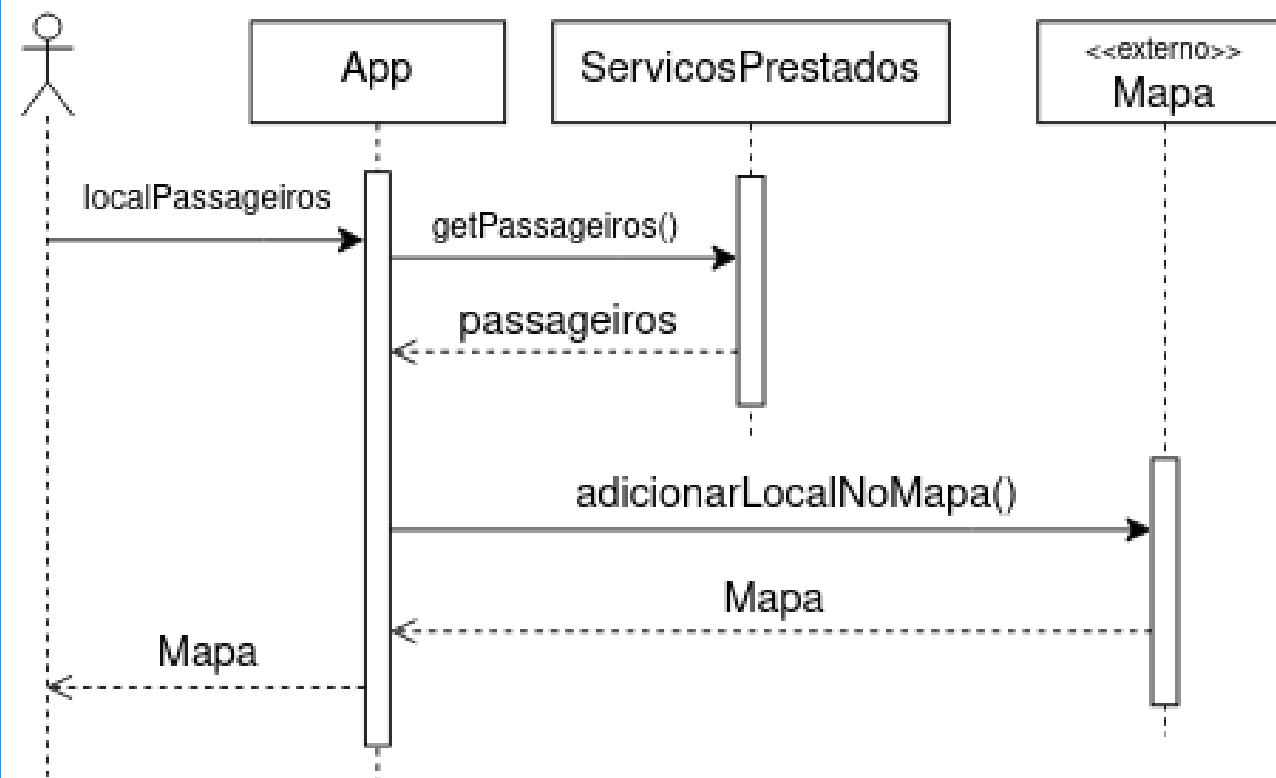


## Realizar pagamento

Passageiro



## Mostrar localizacao dos passageiros



# **Definição de Tecnologias e Padrão Arquitetural**

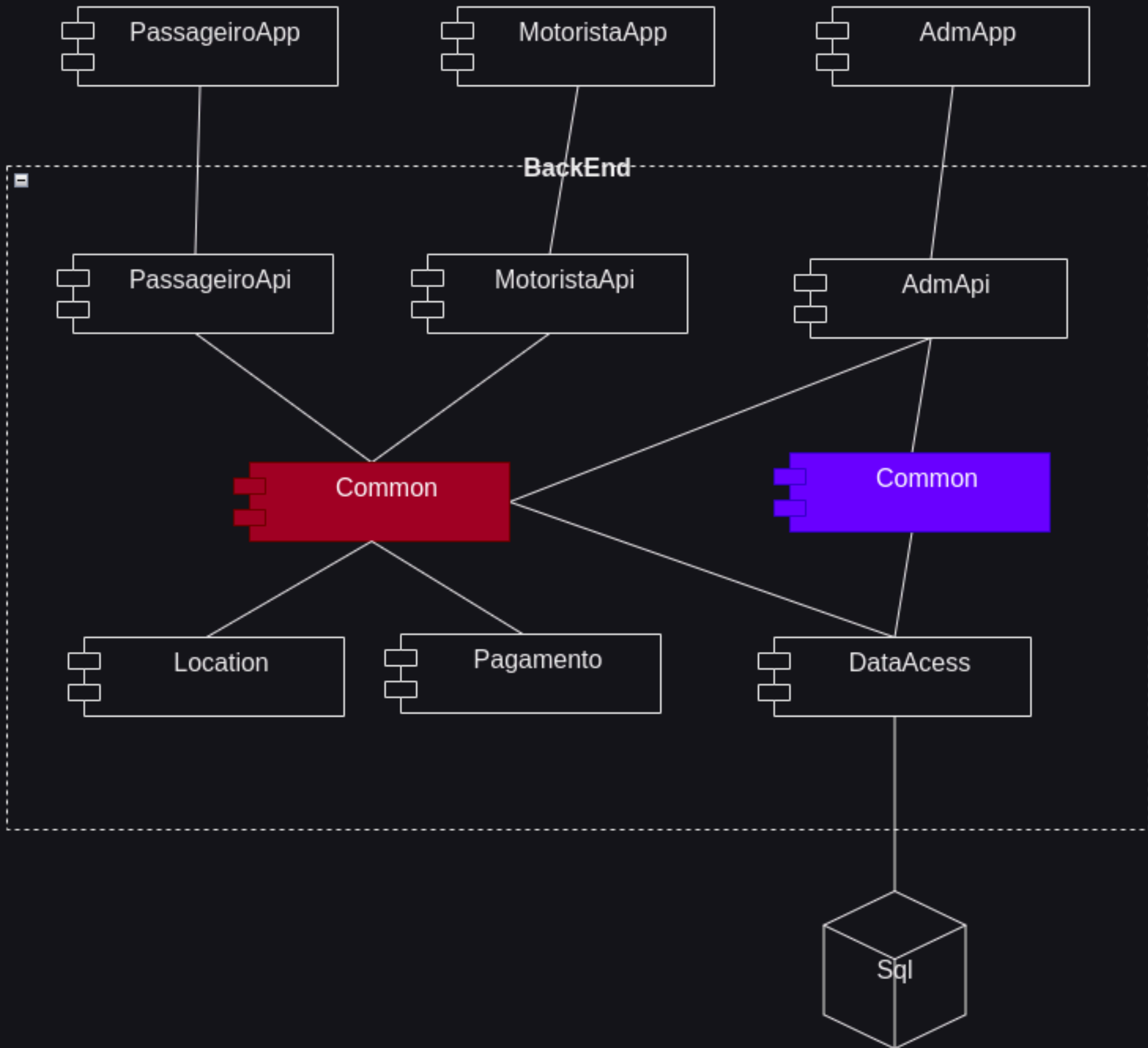
## Tecnologias Frontend:



## Tecnologias Backend:



# Padrão Arquitetural:



## Microserviços

# **Análises Arquiteturais**

# Cenário 1

Sumário

Máximo de requisições de login

Objetivos de negócio

Impedir mais de 5 requisições de login falhas em um curto periodo de tempo

Atributo de Qualidade

Segurança

Abordagem Arquitetural

Bloqueio de conta caso 5 tentativas de login não obtiveram sucesso

Riscos

Usuário terá que realizar uma requisição por outro meio para  
desbloqueio de conta

Trade-offs

Ataques de força bruta para acesso indevido  
de contas será inviabilizado

## Cenário 2

Sumário	Acesso ao serviço estará sempre disponível
Objetivos de negócio	Usuário não pode encontrar problemas de conexão com o servidor em nenhum momento
Atributo de Qualidade	Disponibilidade
Abordagem Arquitetural	Servidor estará em Cloud com constantes backups feitos pelo sistema, mantendo o servidor disponível por maior parte do tempo
Riscos	Nenhum
Trade-offs	Alta disponibilidade do serviço, com um maior custo e maior necessidade de manutenção para ser sempre mantido online



## Cenário 3

Sumário

Usuário vai realizar o pagamento mensal/semanal

Objetivos de negócio

Evitar transtornos durante a realização do pagamento

Atributo de Qualidade

Segurança

Abordagem Arquitetural

Terceirização dos serviços de pagamento, para entidades confiáveis e com reputação aceitável

Riscos

Entidade terceirizada terá queda no serviço de pagamento, causando transtornos ao usuário

Trade-offs

Delegação de serviço de pagamento, tornando o serviço geral mais seguro