

# App Mulher Viajante

## Documento de Arquitetura de Software

Versão <1.0>

### Histórico da Revisão

Data	Versão	Descrição	Autor
<dd/mm/aa>	<x.x>	<detalhes>	<nome>

## Índice Analítico

<b>Introdução</b>	<b>4</b>
Finalidade	4
Escopo	4
Definições, Acrônimos e Abreviações	4
Referências	4
<b>Requisitos e Restrições da Arquitetura</b>	<b>5</b>
<b>Visão de Casos de Uso</b>	<b>5</b>
<b>Visão Lógica</b>	<b>5</b>
Visão Geral	5
Pacotes de Design Significativos do Ponto de Vista da Arquitetura	6
<b>Visão de Processos (opcional)</b>	<b>7</b>
<b>Visão de Implantação</b>	<b>7</b>
<b>Visão da Implementação</b>	<b>7</b>
<b>Visão de Dados (opcional)</b>	<b>7</b>
<b>Volume e Desempenho</b>	<b>7</b>
<b>Qualidade</b>	<b>7</b>

## Documento de Arquitetura de Software

### 1. Introdução

#### 1.1 Finalidade

Este documento oferece uma visão geral arquitetural abrangente do sistema, usando diversas visões arquiteturais para representar diferentes aspectos do sistema. O objetivo deste documento é capturar e comunicar as decisões arquiteturais significativas que foram tomadas em relação ao sistema. O documento irá adotar uma estrutura baseada na visão “4+1” de modelo de arquitetura, conforme (Kruchten, 1994).

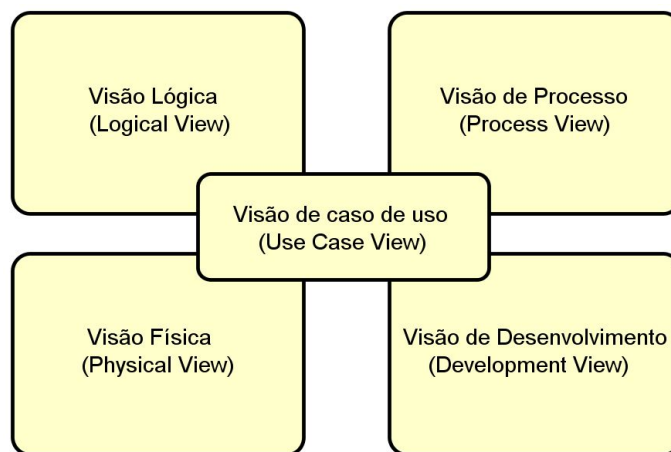


Figura 1 - Visões 4+1 da arquitetura de software (Kruchten, 1994)

## 1.2 Escopo

Este documento de arquitetura fornece uma visão geral de arquitetura do aplicativo Mulher Viajante que está sendo desenvolvido com intuito de facilitar e fomentar a viagem de mulheres.

## 1.3 Definições, Acrônimos e Abreviações

## 1.4 Referências

## 2. Requisitos e Restrições da Arquitetura

<i>Requisito</i>	<i>Solução</i>
<b>Linguagem</b>	O aplicativo será desenvolvido em Javascript/Typescript.
<b>Plataforma</b>	Ionic Framework
<b>Segurança</b>	O sistema utilizará o login do facebook como única forma de autenticação de segurança, mantendo apenas usuárias do sexo feminino, utilizando cookies e Session ID para monitoramento de sessões, além de criptografia na comunicação entre o cliente e o Servidor Web.
<b>Persistência</b>	Os dados serão persistidos no MongoDB.

<b>Testabilidade</b>	Utilização do TDD(Desenvolvimento orientado a Testes), utilizando a ferramenta Jest nos testes unitários automatizados.
<b>Desenvolvimento</b>	Utilização dos seguintes <i>Design Patterns</i> : Injeção de dependências, <i>Factorys</i> e <i>Strategy</i> , além de constante Refatoração do código a fim de mantê-lo mais limpo possível.
<b>Internacionalização</b>	O software será construído, primeiramente, para uso em território brasileiro, sendo assim utilizando o idioma português brasileiro.

*Tabela 1 – Exemplo de requisitos e restrições*

### 3. Visão de Casos de Uso/ Histórias de Usuário

Nessa seção será apresentado os casos de uso usados nessa arquitetura de software. A Visualização de Caso de Uso é uma importante ferramenta para a criação de histórias e cenários que por sua vez são responsáveis pelo levantamento dos das funcionalidades do aplicativo, segue abaixo a listagem dos casos de uso levantados durante o processo.

- Cadastrar usuário(apenas com opção de utilizar o facebook)
- Editar perfil
- Inserir preferências de viagem
- Pesquisar locais baseado no índice de segurança
- Buscar companhia
- Chat para contato entre as mulheres
- Avaliar destino
- Inserir palavras-chaves do local\*
- Enviar localização em tempo real
- Buscar companhia para futuras viagens
- Acessar sugestões de preferências\*
- Enviar localização fixa
- Inserir possíveis destino de viagem

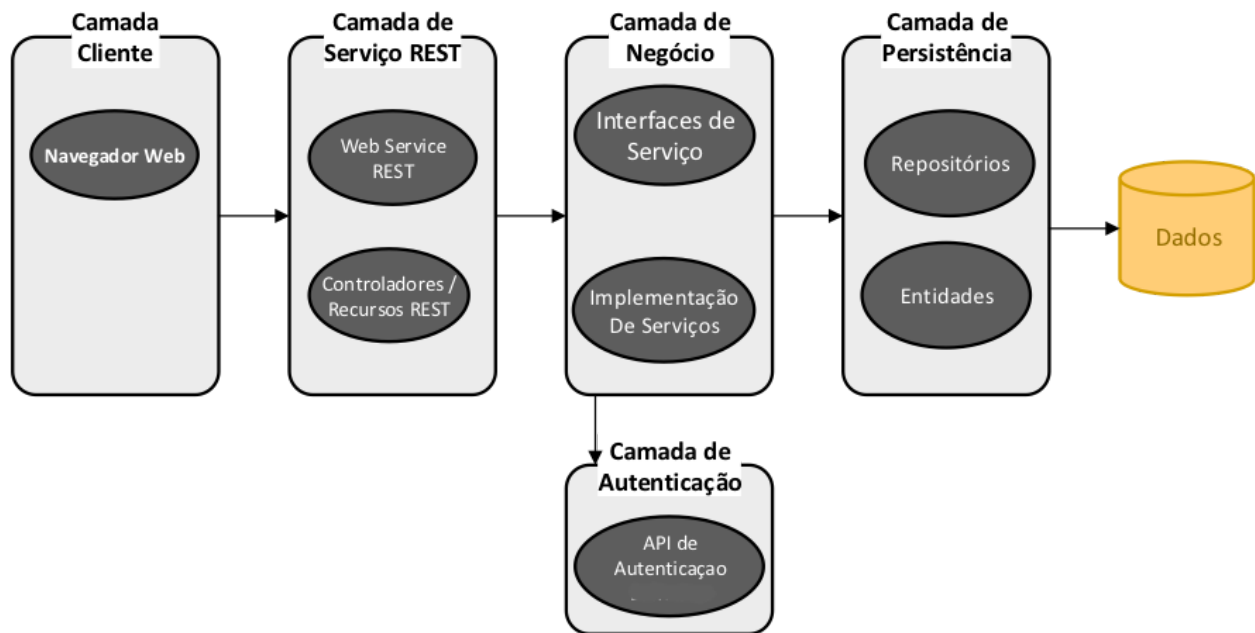
### 4. Visão Lógica

*[Esta seção descreve as partes significativas do ponto de vista da arquitetura do modelo de design, como sua divisão em subsistemas e pacotes. Além disso, para cada pacote significativo, ela mostra sua divisão em classes*

*e utilitários de classe. Apresente as classes significativas do ponto de vista da arquitetura e descreva suas responsabilidades, bem como alguns relacionamentos, operações e atributos de grande importância.]*

### 4.1 Visão Geral

*[Esta subseção descreve toda a decomposição do modelo de design em termos de camadas e de hierarquia de pacotes.]*



*Figura 2 - Exemplo de diagrama de camadas da aplicação*

### 4.2 Pacotes de Design Significativos do Ponto de Vista da Arquitetura

*[Para cada pacote significativo, inclua uma subseção com o respectivo nome, uma breve descrição e um diagrama com todos os pacotes e classes significativos nele contidos.]*

*Para cada classe significativa no pacote, inclua o respectivo nome, uma breve descrição e, opcionalmente, uma descrição de algumas das suas principais responsabilidades, operações e atributos.]*

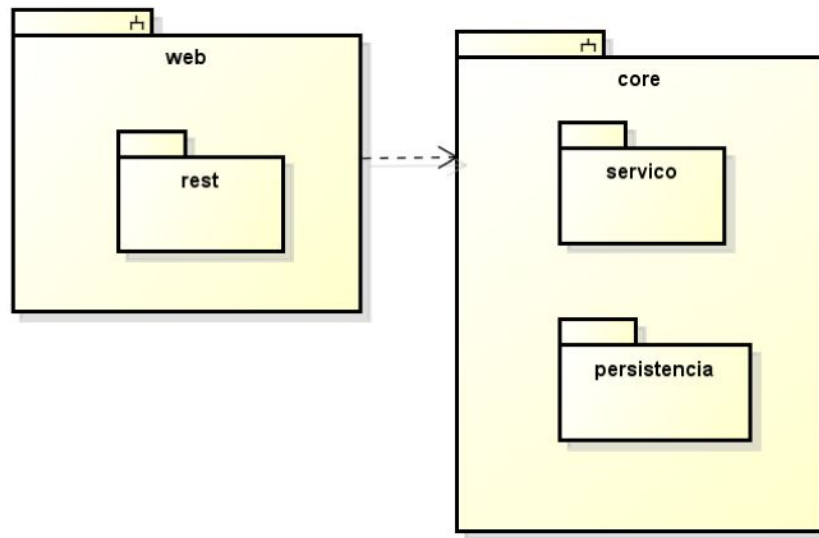


Figura 3 - Exemplo de diagrama de pacotes da UML

### 5. Visão de Processos (opcional)

*[Descreve a decomposição do sistema em processos leves (threads simples de controle) e processos pesados (agrupamentos de processos leves). Organize a seção em grupos de processos que se comunicam ou interagem. Descreva os modos principais de comunicação entre processos, como transmissão de mensagens e interrupções.]*

### 6. Visão de Implantação

*[Esta seção descreve uma ou mais configurações da rede física (hardware) na qual o software é implantado e executado. No mínimo, para cada configuração, ela deve indicar os nós físicos (computadores, CPUs) que executam o software e suas interconexões (barramento, LAN, ponto a ponto, etc.).]*

### 7. Visão da Implementação

*[Esta seção descreve a estrutura geral do modelo de implementação, a divisão do software em camadas e os subsistemas no modelo de implementação e todos os componentes significativos do ponto de vista da arquitetura.]*

### 8. Visão de Dados (opcional)

*[Descrição da perspectiva de armazenamento de dados persistentes do sistema. Será opcional se os dados forem poucos ou inexistentes ou se a conversão entre o Modelo de Projeto e o Modelo de Dados for trivial.]*

### 9. Volume e Desempenho

*[Uma descrição das principais características de dimensionamento do software que têm um impacto na arquitetura, bem como as restrições do desempenho desejado. Exemplos:*

- *Volume: número estimado de usuários, número estimado de acessos diários, tempo de seção de um*

*usuário.*

- *Desempenho: tempo de resposta para uma determinada transação, tempo de iniciação de um seção, etc.]*

## **10. Qualidade**

*[Uma descrição de como a arquitetura do software contribui para todos os recursos (exceto a funcionalidade) do sistema: extensibilidade, confiabilidade, portabilidade e assim por diante. Se essas características possuírem significado especial, como implicações de segurança, garantia ou privacidade, elas deverão ser delineadas claramente. A tabela a seguir exemplifica este tópico. O conteúdo deve ser adaptado conforme o software.]*

<i>Item</i>	<i>Descrição</i>	<i>Solução</i>
<b>Escalabilidade</b>	<i>[Breve Descrição]</i>	<i>[Breve descrição da Solução]</i>
<b>Confiabilidade</b>	<i>[Breve Descrição]</i>	<i>[Breve descrição da Solução]</i>
<b>Disponibilidade</b>	<i>[Breve Descrição]</i>	<i>[Breve descrição da Solução]</i>
<b>Portabilidade</b>	<i>[Breve Descrição]</i>	<i>[Breve descrição da Solução]</i>
<b>Segurança</b>	<i>[Breve Descrição]</i>	<i>[Breve descrição da Solução]</i>