RELATÓRIO – Iteração 4

Implementação – Incremento 1

| Projeto: | AutoHub: Serviços automóveis |
| --- | --- |
| Grupo: | 110509: Rui Albuquerque  117450: Ellen Sales  119859: João Leite  120202: Paulo Lacerda |
| Data/versão: | 2024/06/04, v1 |
| Sumário: | Esta iteração foca o primeiro incremento funcional do AutoHub, validando a arquitetura proposta e implementando funcionalidades essenciais como registo, login, pesquisa e agendamento de serviços, bem como sistema de avaliações. Foram desenvolvidas versões web (para prestadores) e mobile (para clientes), com comunicação simulada via API e persistência temporária em LocalStorage, prevendo futura integração com base de dados real. |

**Conteúdo**

[**1**](#_heading=h.tq9m3jnn2zrf) **Introdução 1**

[1.1](#_heading=h.r4hu5c4t8oke) Sumário executivo 1

[1.2](#_heading=h.t26tgss2d4se) Controlo de versões 2

[**2**](#_heading=h.fm0ubgkjbh40) **Incremento 1 2**

[2.1](#_heading=h.o0nqukvytus3) Histórias de utilização selecionadas 2

[2.2](#_heading=h.rsz0f1cva5jt) Estratégia e estado da implementação 3

[2.3](#_heading=h.nz44wshxyin4) Aceitação e garantia de qualidade 3

# 

# Introdução

## Sumário executivo

Este relatório apresenta os resultados da construção, adaptado os resultados esperados na etapa de Elaboration e Construction, do método OpenUP.

O primeiro incremento foca a validação da arquitetura proposta e histórias nucleares. Foram considerados sobretudo as funcionalidades relacionadas com o registo e autenticação de utilizadores, pesquisa e filtragem de serviços, agendamento de serviços, integração com gateway de pagamentos e sistema de avaliações e feedback.

## Controlo de versões

| **Quando?** | **Responsável** | **Alterações significativas** |
| --- | --- | --- |
| 28/05 | Paulo Lacerda | Introdução |
| 29/05 | Paulo Lacerda | Incremento 1 |
| 01/06 | Ellen Sales | Integração de sistemas |
| 01/06 | Ellen Sales | Incremento 1 - Modificações em histórias do utilizador |

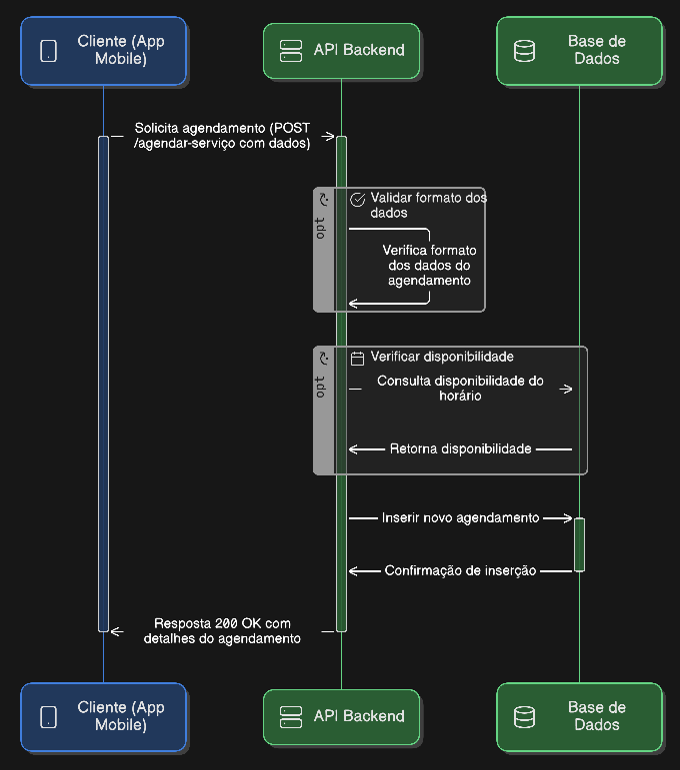
# 

# Integração de sistemas

O sistema prevê integração entre diferentes interfaces (web e mobile), sendo necessário disponibilizar uma **API própria** que permita a comunicação entre clientes e fornecedores de serviços.

A arquitetura contempla os seguintes cenários de integração:

* **Disponibilização de API RESTful**: o sistema disponibiliza endpoints que permitem à aplicação mobile dos clientes interagir com o backend, nomeadamente para operações como: Agendar serviços, consultar estado de agendamentos, cancelar ou reagendar pedidos.
* **Futura ligação com base de dados**: atualmente, o sistema utiliza LocalStorage para simulação de persistência. Está prevista a futura substituição por uma base de dados real, acessada através do backend.
* **Possível consumo de serviços externos no futuro**: embora ainda não implementado, está prevista a integração com serviços externos, como envio automático de emails de confirmação ou notificações via API (e.g. SendGrid ou Firebase Cloud Messaging).

A seguir apresenta-se um **diagrama de sequência** representando o cenário em que a aplicação mobile de um cliente invoca a API para realizar um agendamento de serviço, o qual é processado e armazenado no sistema para posterior consulta por parte do fornecedor.

# Incremento 1

## Histórias de utilização selecionadas

| **Backlog ID** | **Tipo** | **User Story** |
| --- | --- | --- |
| C01 | story/feature | Como cliente, quero criar uma conta na plataforma para poder aceder aos serviços. |
| C02 | story/feature | Como cliente, quero efetuar login para aceder de forma segura à minha conta. |
| C03 | story/feature | Como cliente, quero pesquisar serviços por tipo, localização e preço para escolher a melhor opção. |
| C04 | story/feature | Como cliente, quero agendar data, hora e local de um serviço para marcar atendimento com o prestador. |
| C05 | story/feature | Como cliente, quero avaliar o serviço com pontuação de 0 a 5 e fazer um comentário para partilhar a minha experiência. |
| P01 | story/feature | Como prestador, quero criar uma conta na plataforma para poder oferecer os meus serviços e aceder ao painel de controlo. |
| P02 | story/feature | Como prestador, quero efetuar login e recuperar a password para aceder de forma segura ao meu painel de controlo. |
| P03 | story/feature | Como prestador, quero adicionar, editar e remover serviços (definindo tipo, localização e preço) para gerir o meu portfólio de ofertas. |
| P04 | story/feature | Como prestador, quero consultar, confirmar ou recusar pedidos de agendamento para organizar a minha agenda de forma eficiente. |
| P05 | story/feature | Como prestador, quero visualizar avaliações dos clientes e responder a comentários para melhorar continuamente a qualidade do meu serviço. |

As histórias (*user stories*) incluídas nesta interação fazem parte do *backlog* do projeto, acessíveis em <https://trello.com/b/PaiDUG2V/autohub-implementation> - as user stories encontram-se nas descrições dos cards.

As histórias incluídas nesta interação em relação aos clientes são:

| **História/*use case slice*** | **Critérios de aceitação** |
| --- | --- |
| **Registo de utilizador**  O cliente cria uma conta na plataforma para aceder aos serviços | **Cenário 1: Registo bem-sucedido**  Dado que o cliente não possui conta, quando preenche nome, e-mail e password válidos e submete o formulário o sistema envia confirmação por e-mail e redireciona o cliente para o dashboard. |
| **Autenticação:**  O cliente efetua login e recupera a password para aceder à sua conta de forma segura. | **Cenário 1: Login bem-sucedido**  Dado que o cliente tem conta registada e insere credenciais corretas, quando clica em “Entrar” o sistema autentica o cliente e exibe o dashboard.  **Cenário 2: Password incorreta**  Dado que o cliente insere uma password inválida quando submete as credenciais, o sistema apresenta “Credenciais inválidas”. |
| **Pesquisa e filtragem de serviços:**  O cliente realiza pesquisas, podendo filtrar por oficinas e serviços. | **Cenário 1: Pesquisa com resultados**  Dado que existem oficinas ou serviços que correspondem ao texto pesquisado, quando os filtros são aplicados, os resultados aparecem de acordo com a busca.  **Cenário 2: Sem resultados**  Dado que não existem serviços que satisfaçam os critérios selecionados, informa-se que não há resultados encontrados. |
| **Agendamento de serviços**  O cliente agenda data, hora e local de um serviço para marcar atendimento com o prestador. | **Cenário 1: Agendamento bem-sucedido**  Dado que o horário escolhido está disponível quando se seleciona data, hora e local e confirma, o sistema envia confirmação imediata ao cliente e notifica o prestador. |
| **Avaliações e feedback**  O cliente avalia o serviço com pontuação de 0 a 10 e comentário para partilhar a sua experiência. | **Cenário 1: Avaliação submetida**  Dado que o cliente concluiu um serviço, quando insere pontuação e comentário e submete, o sistema regista a avaliação e exibe “Avaliação registada”.  **Cenário 2: Avaliação sem comentário**  Dado que o cliente submete apenas a pontuação, quando confirma a avaliação, o sistema regista a pontuação sem texto |

Para os provedores de serviço são:

| **História/*use case slice*** | **Critérios de aceitação** |
| --- | --- |
| **Registo de utilizador**  O cliente cria uma conta na plataforma para aceder aos serviços | **Cenário 1: Registo bem-sucedido**  Dado que o prestador ainda não possui conta, quando preenche nome, e-mail, password e dados da empresa válidos e submete o formulário, o sistema redireciona o prestador para o painel de controlo. |
| **Gestão de serviços**  O prestador adiciona, edita e remove serviços, definindo tipo, localização e preço. | **Cenário 1: Criação de serviço**  Dado que o prestador preenche título, descrição, tipo de serviço, localização e preço válidos, ao submeter o novo serviço, o sistema regista o serviço e exibe-o na lista de ofertas do prestador.  **Cenário 2: Edição de serviço**  Dado que o prestador altera os dados de um serviço existente, quando submete as alterações o sistema atualiza o serviço e reflete imediatamente as mudanças no anúncio.  **Cenário 3: Remoção de serviço**  Dado que o prestador decide remover um serviço ao confirmar a eliminação, o sistema apaga o serviço e deixa de o mostrar aos clientes. |
| **Gestão de agendamentos**  O prestador consulta pedidos de agendamento e confirma ou recusa horários. | **Cenário 1: Confirmação de agendamento**  Dado que um cliente marcou, o prestador de serviço consegue aprovar o pedido.  **Cenário 2: Recusa de agendamento**  Dado que o horário solicitado não é conveniente, o prestador de serviço consegue rejeitar o pedido.  **Cenário 3: Visualização de agenda**  Dado que existem múltiplos agendamentos futuros, quando o prestador acede à secção de agenda, o sistema mostra lista cronológica de todos os agendamentos confirmados e pendentes. |
| **Gestão de avaliações e feedback**  O prestador consulta avaliações e responde a comentários dos clientes. | **Cenário 1: Visualização de avaliações**  Dado que existem avaliações submetidas por clientes, quando o prestador acede à secção de feedback, o sistema fornece a lista de todas as avaliações com pontuação e comentário e calcula a média das avaliações. |

## 

## Estratégia e estado da implementação

Neste incremento, o nosso grupo estruturou o projeto num repositório GitHub com o controlo de versões através de branches e uso de pull requests. As tarefas foram organizadas e distribuídas através do Trello, com as colunas Todo, Doing e Done (Método Kanban) permitindo o acompanhamento visual do progresso ao longo das sprints. Implementámos funcionalidades essenciais como registo, login, pesquisa e agendamento de serviços, utilizando dados simulados. O frontend foi desenvolvido em React, com versões distintas para web e mobile, garantindo uma experiência adaptada a cada plataforma. Está ainda em falta a integração entre as aplicações web e mobile, o que exige a implementação de uma base de dados dinâmica e partilhada, essencial para a persistência de dados e sincronização entre interfaces.

O incremento já implementado pode ser acedido na versão web para fornecedores em: <https://autohub-provider.vercel.app>.

A aplicação para utilizadores foi desenvolvida para dispositivos móveis, mas está acessível via navegador para efeitos de avaliação. Para isso, ela está acessível em <https://autohub-client.vercel.app> , entretanto recomenda-se que ativem o modo de visualização mobile na DevTools.

## Aceitação e garantia de qualidade

O sistema AutoHub foi concebido desde a fase inicial com uma arquitetura modular e baseada na integração de interfaces web e mobile, conforme descrito nos relatórios anteriores. Durante a implementação do primeiro incremento, foi validada a comunicação entre estas interfaces através de endpoints definidos numa API RESTful, simulando a persistência de dados via LocalStorage.

A aplicação foi dividida em duas interfaces principais:

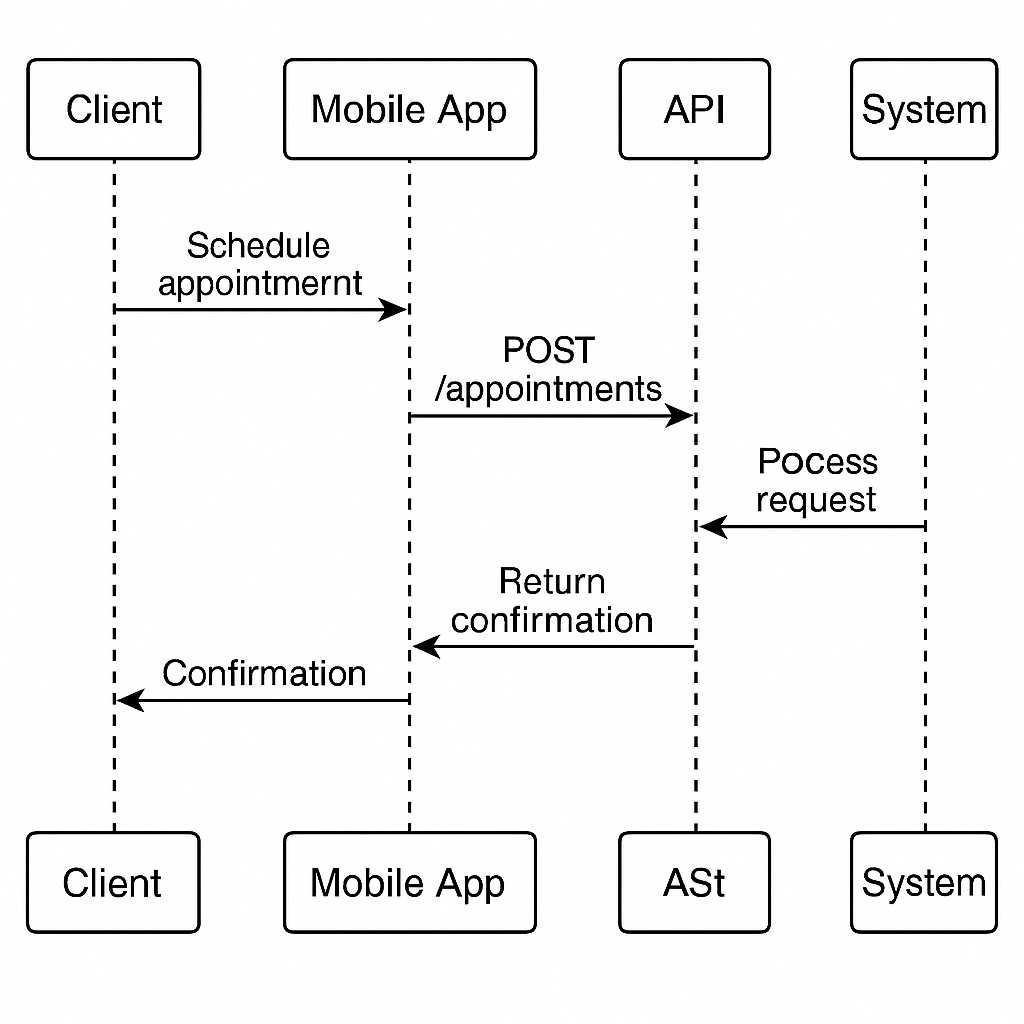
* Portal Web para Prestadores de Serviço: permite a gestão de serviços, agendamentos e resposta a avaliações.
* Aplicação Mobile para Clientes: permite pesquisa de serviços, agendamento e avaliação. Recomenda-se ativar o modo de visualização mobile nas DevTools para uma melhor experiência.

Para efeitos de demonstração, ambas as interfaces operam com dados simulados. A comunicação entre os componentes foi organizada da seguinte forma:

* O frontend (React) comunica com serviços backend simulados.
* Funcionalidades como login, registo, agendamento e avaliações funcionam com dados temporários em memória/local storage.
* As respostas são tratadas de forma síncrona, simulando o comportamento final após futura integração com backend real.

A futura substituição do LocalStorage por uma base de dados real (ex.: Firebase ou PostgreSQL) permitirá persistência de dados entre sessões e sincronização entre plataformas web e mobile.

A interação entre cliente, aplicação e sistema encontra-se representada no diagrama de sequência seguinte:



Este diagrama ilustra o fluxo de comunicação entre o Cliente, a App Mobile, a API e o Sistema, desde a solicitação de um agendamento até a confirmação por parte do sistema.