

RELATÓRIO – ELABORATION & CONSTRUCTION

Construção

Conteúdos

1 Introdução	1
1.1 Sumário executivo	1
1.2 Controlo de versões	2
1.3 Referências e recursos suplementares	2
2 Arquitetura do sistema	2
2.1 Objetivos gerais	2
2.2 Requisitos com impacto na arquitetura	3
2.3 Decisões e justificação	3
2.4 Arquitetura do software	4
2.5 Arquitetura física de instalação	5
3 Incremento 1	5
3.1 Casos de utilização no Incremento 1	5
3.2 Histórias de utilização selecionadas	6
3.3 Estratégia e estado da implementação	6

1 Introdução

1.1 Sumário executivo

Este relatório apresenta os resultados da construção dos incrementos, adaptado aos resultados esperados na etapa de Elaboration e Construction, do método OpenUP. A caracterização dos cenários a ser suportada é detalhada nos casos de utilização apresentados em apêndice (secção 4).

O primeiro incremento, desenvolvido na Iteração 3, foca a validação da arquitetura proposta. Foram consideradas sobretudo as funcionalidades relacionadas com o agendamento de manutenções do automóvel do cliente, feitas pela oficina. Além disso as funcionalidades básicas do site como registo/login, página principal e perfil foram também abordadas.

1.2 Controlo de versões

Quando?	Responsável	Alterações significativas
24/05/2024	Diogo	1.1, 1.3
25/05/2024	Diogo	3.1
26/06/2025	Tiago	2.2

26/05/2024	Tiago	2.5
27/05/2024	Tomás	2.4
26/05/2024	Leonardo	2.1
27/05/2024	Leonardo	3.3
27/05/2024	Diogo	3.2

1.3 Referências e recursos suplementares

Para o desenvolvimento deste relatório, o grupo de projeto procurou artigos e websites relacionados à indústria automóvel, especificamente sobre a manutenção.

- Lorber, E. (2022) *How will prediction models impact the future of vehicle maintenance?*, *RTInsights*. <https://www.rtinsights.com/how-will-prediction-models-impact-the-future-of-vehicle-maintenance/> (Accessed: 15 April 2024).
- Kozlowski, A. (2023) *How predictive maintenance changes the automotive industry*, *Grape Up*. <https://grapeup.com/blog/how-predictive-maintenance-changes-the-automotive-industry/> (Accessed: 15 April 2024).

2 Arquitetura do sistema

2.1 Objetivos gerais

Os principais objetivos da arquitetura deste produto são integrar com sistemas externos, incluindo plataformas de pagamento eletrónico como Paypal e MBWay. Garantir um desempenho robusto e contínuo, com tolerância a falhas e suporte para até 2000 utilizadores simultâneos sem degradação do desempenho. Por outro lado, assegurar também compatibilidade multi-plataforma, permitindo acesso via navegadores web em desktops, tablets e smartphones, com uma interface responsiva e consistente.

Já a usabilidade deve ser intuitiva, com navegação simples, validação de campos em tempo real, busca inteligente e filtros detalhados para melhor procura de peças. O design do sistema deve ser escalável para acomodar um aumento no número de utilizadores e transações, com flexibilidade para adicionar novas funcionalidades, (como o sistema de leilão e um marketplace de peças) e realizar manutenção sem impacto negativo.

O sistema deve sugerir peças compatíveis com os veículos registados na garagem do utilizador e informar quais peças são ou não compatíveis com o seu veículo. Os clientes devem conseguir agendar manutenções de forma facilitada através de uma barra de agendamento com um calendário intuitivo. Além disso, os clientes devem poder acompanhar o estado das suas compras de peças em tempo real.

2.2 Requisitos com impacto na arquitetura

Requisitos	Descrição
RDes.5	O sistema deve suportar até 2000 utilizadores em simultâneos sem degradação significativa no desempenho..
RSegIn.1	Implementar autenticação de dois fatores para aumentar a segurança no registo e no login.

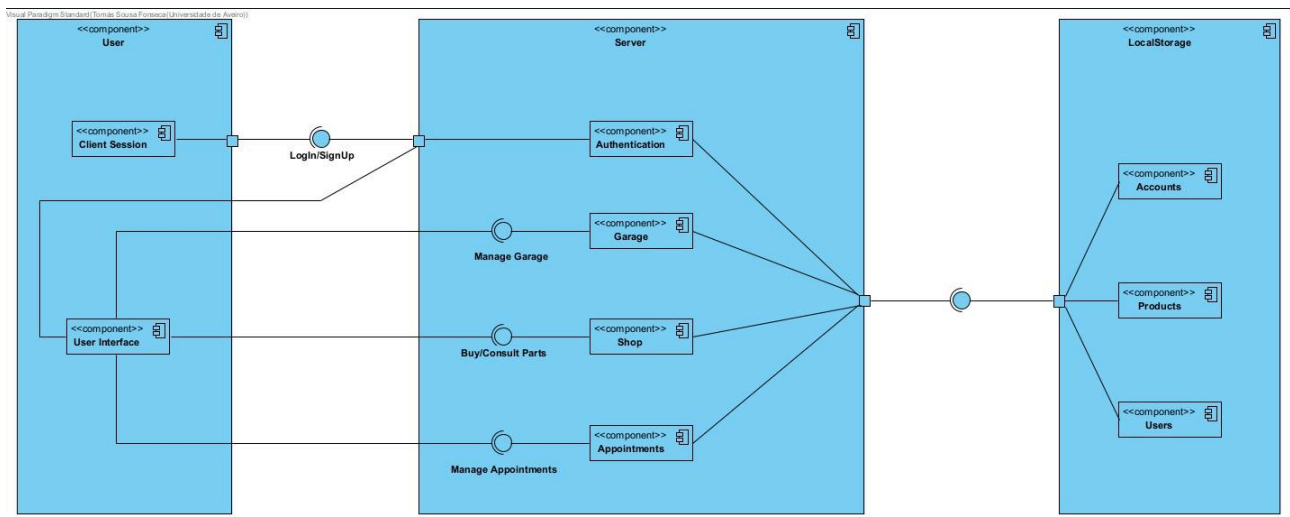
Requisitos	Descrição
RSegIn.2	Todos os dados pessoais dos usuários devem ser criptografados durante o armazenamento e a transmissão.
RSeg.1	Integração com sistemas de pagamento online externos para processar transações de compra de peças.

2.3 Decisões e justificação

Tendo em conta os objetivos para a arquitetura, e os requisitos levantados na Análise, foram tomadas as seguintes decisões:

- O front-end é implementado com HTML e CSS. O HTML foi escolhido por ser a linguagem mais simples e fácil de manipular para escrever as bases das páginas e CSS foi escolhido também por ser direto na sua linguagem para a customização das páginas.
- Também foi utilizado Javascript para algumas funcionalidades do front-end, por ser a melhor linguagem para escrever pequenas funcionalidades a um website, que não necessitam muito código.
- Os dados do website vão ser guardados em Local Storage, pois são poucas as informações necessárias a guardar pelo que não justifica a implementação de uma base de dados mais complexa.

2.4 Arquitetura do software



A articulação entre os módulos decorre da seguinte forma:

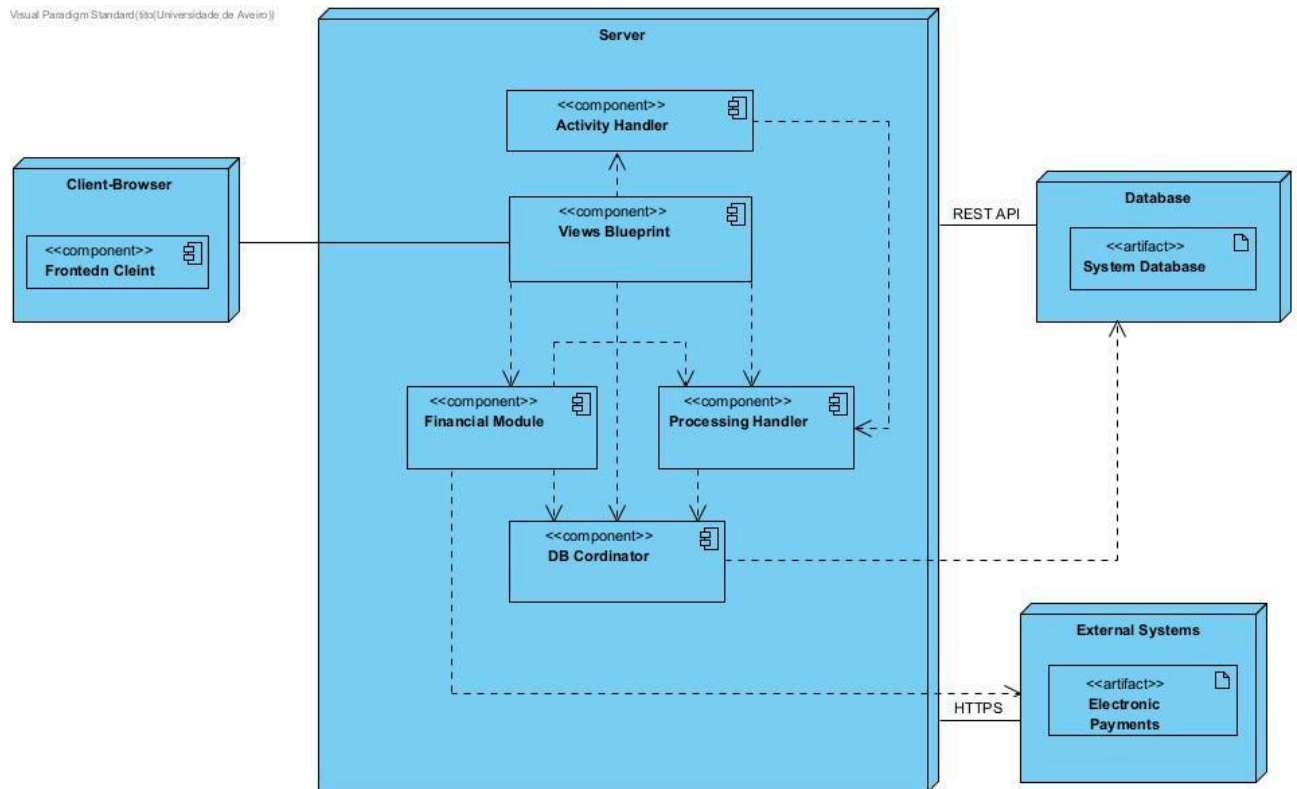
A sessão do utilizador inicializa-se quando o mesmo efetua o “Login”, sendo que o servidor realiza a autenticação, verificando no armazenamento local se este está presente e se os seus dados estão corretos. De seguida, a partir da interface de utilizador (Website), o utilizador pode selecionar as seguintes opções:

- Garagem, em que o servidor redireciona à página onde o utilizador pode gerir os seus veículos
- Loja (Página Inicial), onde o utilizador pode consultar/comprar peças automóveis

- Agenda, localizada no perfil onde o utilizador poderá agendar manutenções ou consultar as já existentes

Qualquer uma destas opções terá de aceder ao armazenamento local para proceder a veículos registados, produtos disponíveis e as suas respectivas informações, e por último para verificar as marcações efetuadas na agenda.

2.5 Arquitetura física de instalação



- Os componentes contidos no “Server” são instalados no servidor que irá hospedar o serviço;
- Os utilizadores acedem à plataforma através de uma página WEB, representada pelo componente “Frontend Client”, este interage com o componente “Views Blueprint”, para que o utilizador seja redirecionado para a página desejada;
- O componente “Activity Handler” regista a atividade do utilizador, de forma a garantir os objetivos de segurança;
- O componente “Processing Handler” ajuda no processamento da informação a ser apresentada e retirada da página WEB;
- O componente “Financial Module” acede através de HTTPS a uma infraestrutura externa que aloja o “Electronic Payments” de forma a ser possível a realização de pagamentos eletrónicos através dos mais diversos métodos.
- O componente “DB Coordinator” é responsável pela leitura e escrita na base de dados;
- A Base de dados é instalada no servidor que irá hospedar o serviço, acedida apenas pelo componente “DB Coordinator”. A conexão é feita com REST API, via HTTPS.

3 Incremento 1

3.1 Casos de utilização no Incremento 1

A primeira fase da construção foi centrada numa das funcionalidades principais, o agendamento de manutenções para o veículo. No entanto, antes de aprofundar essa funcionalidade, foi dedicado tempo para criar funções base do site tais como o registo/login e páginas principais.

No primeiro incremento implementado, o foco esteve na validação da arquitetura proposta, através da implementação de funcionalidade representativa do *core* do negócio. Para isso, seleccionámos casos de utilização da autenticação, Registo e Login, e ainda um caso para a reserva da manutenção, Adicionar Evento.

- **Registo:** O cliente cria uma nova conta no website, fornecendo um username, email e password.
- **Login:** O cliente entra numa conta já criada no website, fornecendo o email e password.
- **Adicionar Evento:** O cliente reserva uma data para a manutenção, fornecendo um nome para o evento e escolhendo dia, mês e ano. Também é possível remover e mudar o nome do evento.

A especificação detalhada dos casos de utilização encontra-se no anexo B do relatório de Análise (E3). A partir dessa análise, definiram-se as histórias de utilização a implementar.

3.2 Histórias de utilização seleccionadas

As histórias (*user stories*) incluídas nesta interação fazem parte do *backlog* do projeto, acessíveis em <https://docauto.atlassian.net/jira/software/projects/DA/boards/1/backlog>

Histórias incluídas nesta interação:

História/use case slice	Critérios de aceitação
O Tiago cria uma conta no website DocAuto O Tiago descobriu a existência do website por recomendação de um amigo. Tem um carro mais antigo que precisa de manutenções constantes. Pretende criar uma conta para investigar melhor o website.	Cenário 1: Registo com sucesso Dado que estou na página principal da DocAuto. Clico no ícone “Sign In” e sou redirecionado para a página de login. Como não tenho conta clico no link “Register”. Preencho os campos com Username: Tiago; Email: tiago@gmail.com ; Password: Tiago123; Repeat Password: Tiago123. Aceito a Política de Privacidade. Clico no botão “Registar” e sou redirecionado para a página principal com a conta ativada. Cenário 2: Registo sem sucesso Dado que estou na página principal da DocAuto. Clico no ícone “Sign In” e sou redirecionado para a página de login. Como não tenho conta clico no link “Register”. Preencho os campos com Username: Tiago; Email: tiagmail.com ; Password: 123; Repeat Password: 123. Aceito a Política de Privacidade.

	Clico no botão “Registrar” e sou avisado de problemas: Email mal formatado e Password demasiado fraca.
O Diogo marca uma manutenção para 06/06/2024 O Diogo já tem uma conta no website. Normalmente arranjava o carro sozinho mas agora pretende marcar uma manutenção feita por um profissional na oficina.	Cenário 1: Marcação com sucesso Dado que estou na página principal da DocAuto. Clico no ícone “Perfil” e sou redirecionado para o perfil. Clico no ícone de calendário e sou redirecionado para a página de eventos. Clico na data correta e clico em “Adicionar Evento”. Preencho o campo com Evento: Manutenção 6 . Clico no botão “Salvar” e aparece um evento novo na direita.
A Luísa elimina a manutenção do dia 05/06/2024 A Luísa já tem uma conta no website. Tinha uma manutenção marcada para o dia 5, mas teve um contratempo e tem que a desmarcar.	Cenário 1: Evento removido com sucesso Dado que estou na página principal da DocAuto. Clico no ícone “Perfil” e sou redirecionado para o perfil. Clico no ícone de calendário e sou redirecionado para a página de eventos. Clico na data correta e aparece o evento “Manutenção 5”. Clico no ícone do caixote do lixo e o evento desaparece.

3.3 Estratégia e estado da implementação

Implementações Realizadas

1. Registo e Login de Utilizadores:
 - Implementação de registo e login de utilizadores com autenticação de dois fatores.
2. Página Principal e Gestão de Perfil:
 - Criação da página principal e funcionalidades de gestão de perfil, permitindo atualização de informações pessoais.
3. Agendamento de Manutenções:
 - Implementação da funcionalidade de agendamento de manutenções com uma interface de calendário intuitiva.

Abordagem e Ferramentas Utilizadas

1. Front-end:
 - HTML, CSS, JavaScript e Bootstrap para uma interface responsiva.
2. Back-end:
 - Python e Flask para lógica de negócio e servidor.
3. Base de Dados:
 - MySQL.
4. Segurança:
 - HTTPS, autenticação de dois fatores e criptografia de dados.

Pontos em Falta

1. Conclusão da opção Garagem
 - A realizar na próxima iteração, onde o utilizador irá adicionar o seu veículo.
2. Sugestão de peças
 - Sugestão de peças baseado no tipo e modelo de veículo do utilizador, planeado para o futuro.
3. Acompanhamento de Estado de Encomendas/Manutenções:
 - Em desenvolvimento, a ser completado em iterações futuras.
4. Filtros mais Avançados no Marketplace:
 - Planeados para uma próxima iteração.