Miguel Vila (107276), Diogo Silva (108212), Martim Carvalho (108749), Ivo Delgado (107757)
Versão deste relatório: **09-05-2023**, v1.0.0

RELATÓRIO – *ELABORATION & CONSTRUCTION*

Construção

Conteúdos

1. Introdução

Neste relatório é apresentada a "*Gestire",* uma solução de software e hardware que permite que organizações giram facilmente a disponibilidade dos seus espaços e equipamentos por parte dos seus funcionários e utentes.

A *Gestire* será composta por uma plataforma multiplataforma acessível em qualquer lugar e em qualquer momento para gerir a requisição de salas e equipamentos, que comunicaria com cacifos inteligentes capazes de oferecer um controlo completo e eficaz sobre os equipamentos.

1.1 Sumário executivo

Este relatório apresenta os resultados da 3º iteração (fase de Elaboration e Construction adaptada do método OpenUp), em que se desenvolveu a arquitetura de execução do projeto dos incrementos. A caracterização dos cenários a ser suportada é detalhada nos casos de utilização apresentados em apêndice (secção 4).

1.2 Controlo de versões

Quando?	Responsável	Alterações significativas

1.3 Referências e recursos suplementares

A própria equipa que, como antigos estudantes do DETI (Departamento de Engenharia de Telecomunicações e Informática) da Universidade de Aveiro, sabem as dificuldades que existem quer em requisitar salas, quer em gerir a utilização de equipamentos partilhados, já que, por vezes, passaram por alguns constrangimentos. Também foram feitos questionários aos atuais alunos e professores da instituição de modo a perceber quais seriam as funcionalidades de maior interesse e utilidade.

2 Arquitetura do sistema

2.1 Objetivos gerais

- Os utilizadores poderão aceder à plataforma a partir de qualquer dispositivo com acesso à internet (computadores, smartphones, tablets) através das apps, ou utilizando um navegador. A navegação e o modo de funcionamento da aplicação deverá ser idêntico, independente do dispositivo utilizado para o acesso;
- Os utilizadores poderão consultar informações detalhadas sobre as salas, como a sua lotação máxima, como está equipada e o número de tomadas nela existentes;
- Os utilizadores poderão filtrar as salas pelas características que lhe sejam convenientes;
- Os utilizadores poderão requisitar salas para determinado dia e período de tempo;
- Os utilizadores poderão consultar a disponibilidade dos equipamentos no sistema de cacifos;

- Os utilizadores poderão requisitar equipamentos durante um período de tempo e posteriormente devolvê-los.
- A plataforma deverá ser capaz de usar o IdP da instituição para efetuar a autenticação.

2.2 Requisitos com impacto na arquitetura

Requisitos	Descrição
RInt.1	Ser facilmente utilizada na utilização com screen readers.
Rint.2	Identificar alunos através da autenticação no idP da UA.
Rdes.1	Capacidade para vários utilizadores em simultâneo.
Rdes.2	UI adaptada a qualquer dispositivo.

2.3 Decisões e justificação

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

- Frontend implementado com Dart/Flutter: existem muitos recursos sobre estas tecnologias, tem uma ótima documentação oficial e é fácil o desenvolvimento para todas as plataformas-alvo deste serviço.
- Backend implementado com Python/Flask: existem muitos recursos sobre estas tecnologias, a *user-base*. Dada a sua facilidade, é fácil de fazer a manutenção do código. Dada a utilização que se prevê para a instituição em causa (DETI - UAveiro), são as tecnologias que oferecem uma melhor relação entre facilidade de utilização e performance.
- Hardware implementado com ESP32 e Micro Python: o ESP32 é uma placa de baixo custo, com uma boa eficiência energética que se enquadra perfeitamente em termos de preço para esta solução.

2.4 Arquitetura do software

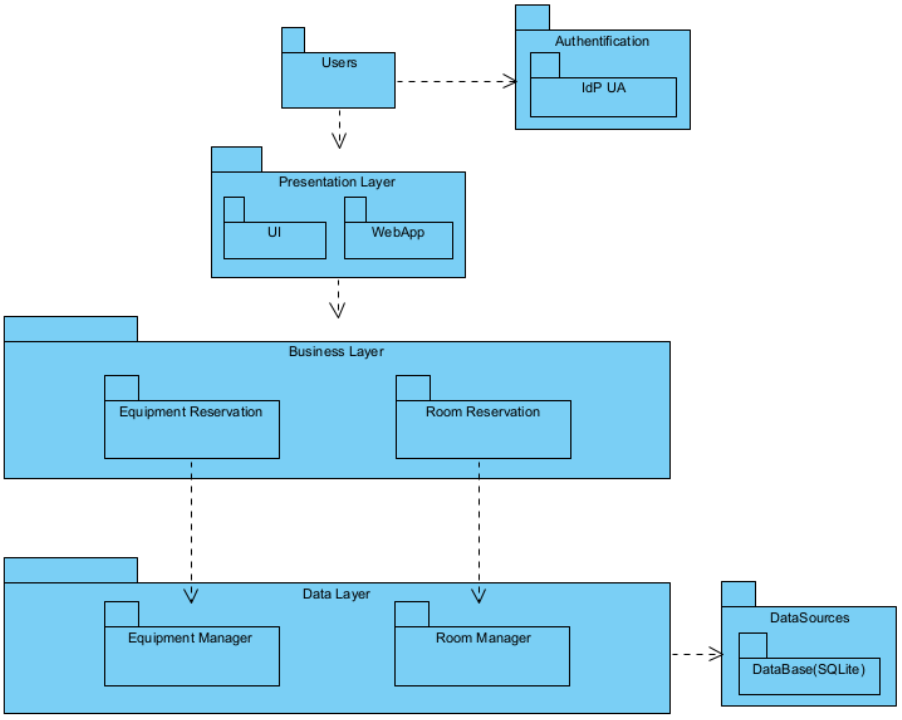

Fig. 1 - Diagrama de pacotes representativo da organização do software.

De modo a representar adequadamente a arquitetura do software utilizou-se uma vista lógica por módulos que consiste num diagrama de pacotes baseado em “blocos” e “setas” organizado por camadas (layered architecture).

A articulação entre os módulos decorre da seguinte forma:

No topo do diagrama está um pacote que representa o utilizador do sistema, que necessita de se autenticar utilizando o IdP da universidade de Aveiro. Na camada abaixo, encontra-se a lógica de apresentação do sistema, ou aquilo que é apresentado ao utilizador, como por exemplo, a UI do sistema. Esta camada estabelece um relação de dependência com a camada diretamente abaixo, a lógica do domínio, onde são apresentadas as principais funções do sistema que, por sua vez, dependem de um gerenciamento de dados com recurso a uma base de dados, neste caso implementada utilizando SQLite.

2.5 Arquitetura física de instalação

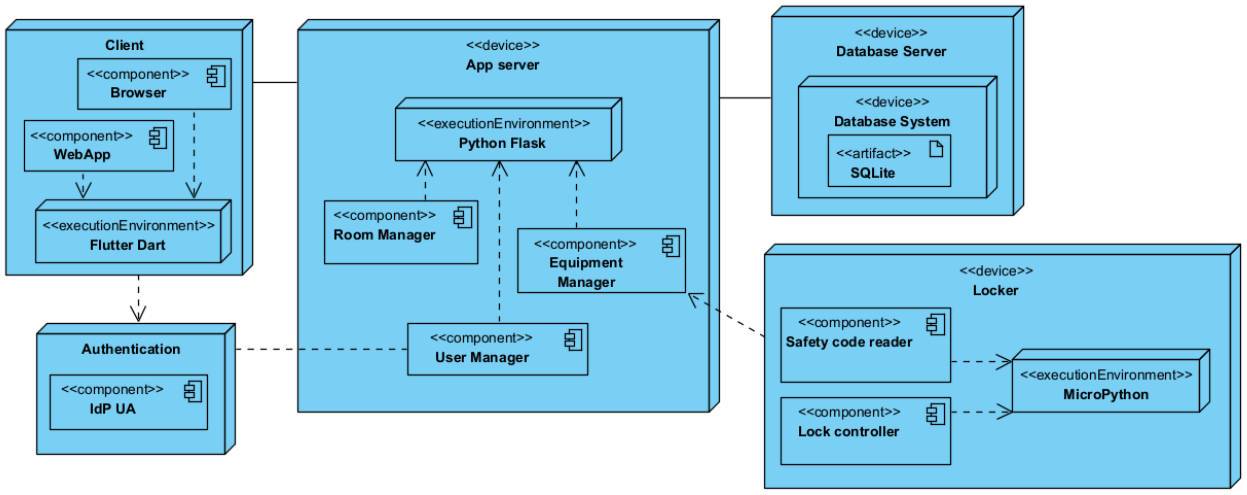

Fig. 2 - Diagrama de instalação relativo à organização da solução.

Para explicar graficamente qual se prevê ser a solução para a implementação dos diversos componentes do sistema num ambiente de produção criou-se um diagrama de instalação(Fig. 2).

Alguns principais componentes do sistema, como pode ser visto no diagrama são o cliente e o serviço de autenticação, o *identity provider* da UA(IdP UA), a que este recorre para ter acesso ao sistema. No centro do diagrama está representado o servidor da aplicação e os serviços que o mesmo engloba, que são a gestão de salas, equipamentos e utilizadores. Este servidor, como pode ser visto no módulo "*executionEnvironment"* utiliza o framework web *Flask*, escrito em *Python*.

O diagrama também apresenta uma base de dados implementada com SQL, da qual o servidor está dependente. Por fim, o módulo "*Locker"* representa um dos cacifos inteligentes do qual o sistema irá fazer uso para que o utilizador possa levantar e depositar os equipamentos que requisitar. Por esse mesmo motivo, está estabelecida uma relação de dependência entre o "*Locker"* e o "*Equipment Manager"*.

3 Incremento 1

3.1 Casos de utilização no Incremento 1

No primeiro incremento implementado, o foco esteve na validação da arquitetura proposta, através da implementação de funcionalidades úteis para o *core* do negócio. Para isso, selecionamos os casos de uso "Autenticar", "Pesquisar Sala" e "Pesquisar Equipamento".

Caso de Utilização	Sinopse
1 - Autenticar	O utilizador acede à plataforma e, como não tem sessão iniciada, autentica-se com a sua conta UA.

2 - Pesquisar Sala	O utilizador já autenticado consegue pesquisar pelo nome da sala.
3- Pesquisar Equipamento	O utilizador já autenticado consegue pesquisar pelo nome do equipamento.

A especificação detalhada dos casos de utilização encontra-se em anexo (secção 4). A partir dessa análise, definiram-se as histórias de utilização a implementar.

3.2 Histórias de utilização selecionadas

As histórias (*user stories*) incluídas nesta interação fazem parte do *backlog* do projeto, estando presentes no mesmo.Histórias incluídas nesta interação:

História/*use case slice*	Critérios de aceitação
O Jorge pesquisa por um equipamento na aplicação. Sendo o Jorge, um indivíduo com login válido da UA. Quer requisitar um equipamento pelo nome. De modo, a poder utilizar para as suas necessidades dentro do DETI.	**Cenário 1: Pesquisa com sucesso** Abro a aplicação e procedo à autenticação na UA, Clico na aba de equipamentos, Seleciono o equipamento pretendido perante os apresentados, Caso esteja disponível insiro a duração e onde me vou encontrar, Introduzo o código gerado no cacifo, Retiro o equipamento e faço usufruto do mesmo, Quando for guardar o equipamento abro a aplicação, Introduzo um novo código no cacifo e guardo o equipamento. **Cenário 2: Login inválido** Dado que estou na página de entrada e não consigo realizar a autenticação, Existe a menção “Nome de utilizador desconhecido” ou “A palavra-passe fornecida está incorreta”. **Cenário 3: Pesquisa sem resultados** Abro a aplicação e procedo à autenticação na UA, Clico na aba de equipamentos, Na página o equipamento pretendido não está disponível, apresenta a menção “Unavailable”.
A Professora Maria procura por uma sala no DETI. Sendo a Maria, uma professora com login válido da UA. Quer requisitar uma sala com pelo menos 25 cadeiras. Com o objetivo de fazer uma sessão de esclarecimento de dúvidas.	**Cenário 1: Pesquisa com sucesso** Abro a aplicação e procedo à autenticação na UA, Clico na aba de Salas, Filtro a sala de acordo com as minhas necessidades (nº de lugares, tomadas, computadores ou outros equipamentos), Insiro a data, hora e duração e opcionalmente coloco a finalidade, Caso a reserva seja aceite irá aparecer a mensagem “Success!”. **Cenário 2: Login inválido** Dado que estou na página de entrada e não consigo realizar a autenticação, Existe a menção “Nome de utilizador desconhecido” ou “A palavra-passe fornecida está incorreta”.

3.3 Estratégia e estado da implementação

No final deste incremento, demos como implementadas com sucesso:

- uma página inicial de login com autenticação de email da ua, onde é possível verificar a validação das credenciais de autenticação (caso não contenham @ua.pt não serão validos);
- uma página com a lista das salas disponíveis e as suas características com uma barra de pesquisa para facilitar a procura;
- uma página com a lista de equipamentos disponíveis e as suas características com uma barra de pesquisa para facilitar a procura;

Nas implementações em cima referidas foram utilizadas as seguintes ferramentas:

- Python: linguagem de programação utilizada no backend responsável pela API, persistência dos dados e autenticação (para simular um IdP);
- Flask: framework do Python para a criação da API para que os clientes e os cacifos consigam comunicar com o servidor por http;
- Dart: linguagem de programação utilizada no lado do cliente;
- Flutter: framework que permite a criação de interfaces de utilizador multi-plataforma em Dart.

No fim desta iteração não foi possível concluir a pesquisa por filtragem (tanto nas salas como nos equipamentos) já que a seriação de dados será feita no *backend*, o qual se encontra ainda num estado muito inicial. Por agora, apenas é possível fazer a pesquisa por palavras. O menu de entrega/devolução de equipamentos e requisição de salas ainda não foi implementado. A nível de hardware ainda não se conseguiu fazer a integração do nosso sistema com o controlador dos cacifos. Este controlador consiste num ESP32.

Apêndice

4 Especificação dos casos de utilização

4.1 Pacote: Autenticação

Caso de utilização:	#1: Autenticar
Versão:	Iteração 1, v2023-06-23
Breve descrição:	O utilizador acede à plataforma e autentica-se com a sua conta UA.
Pré-condições:	conexão à internet, plataforma disponível, não ter sessão iniciada
Pós-condições:	NA
Fluxo base:	**1. Abrir a plataforma gestire em qualquer dispositivo** Inicia quando o utilizador abre a aplicação pela primeira vez, terminou sessão ou a sessão expirou.

	2. Preencher Dados O utilizador escreve o seu utilizador universal acompanhado da sua palavra passe e carrega no botão de login. **3. Acesso à Secção Salas** Após o login com sucesso, é apresentada ao utilizador a secção “Rooms”
Fluxos alternativos:	**Passo 3: Mensagem de Erro** O sistema informa o utilizador de que a combinação de utilizador universal e password não são reconhecidos.
Exceções:	**Ex1: Passo 2: A API não responde dentro de 10 segundos.**
Requisitos especiais:	[Desempenho] Não deverá demorar mais do que 10 segundos.
Aspectos em aberto:	NA

4.2 Pacote: Pesquisa

4.2.1 Pesquisa por Sala

Caso de utilização:	#1: Pesquisar Sala
Versão:	Iteração 1, v2023-06-23
Breve descrição:	O utilizador pesquisa pelo nome da sala.
Pré-condições:	conexão à internet, plataforma disponível, ter sessão iniciada
Pós-condições:	NA
Fluxo base:	**1. Abrir a plataforma gestire em qualquer dispositivo** Inicia quando o utilizador abre a secção “Rooms”. **2. Preencher Dados** O utilizador escreve o termo de pesquisa. **3. Visualização dos Resultados** É apresentada ao utilizador uma lista com resultados que respeitem os seus termos de pesquisa.
Fluxos alternativos:	**Passo 3: Mensagem de Erro** O sistema informa o utilizador de que os termos de pesquisa não levaram a nenhum resultado.
Exceções:	NA
Requisitos especiais:	[Desempenho] Não deverá demorar mais do que 10 segundos.
Aspectos em aberto:	A filtragem dos resultados pelos termos de pesquisa deverá ser feita no cliente ou por uma chamada ao backend.

4.2.2 Pesquisa por Equipamento

Caso de utilização:	#1: Pesquisar Equipamento
Versão:	Iteração 1, v2023-06-23
Breve descrição:	O utilizador pesquisa pelo nome do equipamento.
Pré-condições:	conexão à internet, plataforma disponível, ter sessão iniciada
Pós-condições:	NA
Fluxo base:	**1. Abrir a plataforma gestire em qualquer dispositivo** Inicia quando o utilizador abre a secção “Equipments”. **2. Preencher Dados** O utilizador escreve o termo de pesquisa. **3. Visualização dos Resultados** É apresentada ao utilizador uma lista com resultados que respeitem os seus termos de pesquisa.
Fluxos alternativos:	**Passo 3: Mensagem de Erro** O sistema informa o utilizador de que os termos de pesquisa não levaram a nenhum resultado.
Exceções:	NA
Requisitos especiais:	[Desempenho] Não deverá demorar mais do que 10 segundos.
Aspectos em aberto:	A filtragem dos resultados pelos termos de pesquisa deverá ser feita no cliente ou por uma chamada ao backend.