

Instituto Politécnico de Viseu
Escola Superior de Tecnologia e Gestão de Viseu
Departamento de Informática



Relatório de Projeto Integrado

Aplicação de Gestão de Reservas para Reuniões

Realizado por

Daniel Bernardo, estgv17918

Daniel Tejo, estgv17743

João Pinto, estgv17744

Marcelo Silva, estgv17724

Ricardo Sá, estgv17722

Viseu, 2022

Instituto Politécnico de Viseu
Escola Superior de Tecnologia e Gestão de Viseu
Departamento de Informática

Relatório de Projeto Integrado
Cursos de Licenciatura em
Engenharia Informática

Aplicação de Gestão de Reservas para Reuniões

Realizado por
Daniel Bernardo, estgv17918
Daniel Tejo, estgv17743
João Pinto, estgv17744
Marcelo Silva, estgv17724
Ricardo Sá, estgv17722

Ano Letivo 2021/2022

Viseu, 2022

Agradecimentos

Queremos agradecer à ESTGV e à Softinsa por se terem unido de modo a dar-nos a oportunidade de desenvolvermos este projeto, que engloba as unidades curriculares Projeto Integrado, Aplicações para a Internet II, Programação de Dispositivos Móveis e Base de Dados I da Licenciatura de Engenharia Informática.

Queremos agradecer também a todos os docentes do departamento de informática que nos acompanharam e guiaram ao longo do desenvolvimento deste Projeto Integrado.

Índice

1. Introdução.....	1
1.1. Contextualização do projeto	1
1.2. Descrição das entidades.....	2
1.2.1. ESTGV.....	2
1.2.2. Softinsa	3
1.3. Objetivos	4
1.4. Estrutura do relatório.....	5
2. Ferramentas e Técnicas	6
2.1. HTML	6
2.2. CSS	7
2.3. JavaScript	8
2.4. Bootstrap.....	9
2.5. Node.js.....	10
2.6. Express.js	11
2.7. React.....	12
2.8. Heroku	13
2.9. JWT.....	14
2.10. PostgreSQL.....	15
2.11. SQL	16

2.12. Android Studio	17
2.13. Kotlin.....	18
3. Corpo	19
3.1. Conceção	19
3.1.1. Requisitos.....	19
3.1.2. Checklist de requisitos	22
3.1.3. Modelo de dados	25
3.2. Aplicação Mobile.....	26
3.3. Aplicação Web.....	33
4. Conclusão.....	40
5. Bibliografia	41

Índice de Figuras

Figura 1 - Modelo de Entidade-Relacionamento.....	25
Figura 2 - Modelo Físico	25
Figura 3 - Login (Mobile).....	26
Figura 4 - Menu	27
Figura 5 - Lista Centros	27
Figura 6 - Reservar Sala	28
Figura 7 - Reservas Efetuadas	29
Figura 8 - Detalhes Reserva.....	29
Figura 9 - Editar Reserva.....	30
Figura 10 - Mudar Password (Mobile)	31
Figura 11 - Exemplo de tela de dispositivo Fixo.....	31
Figura 12 - Exemplo Sala 1 dispositivo Fixo	32
Figura 13 - Login (Web).....	33
Figura 14 - Dashboard	33
Figura 15 - Lista de Reservas	34
Figura 16 - Criar Centro	34
Figura 17 - Lista dos Centros	34
Figura 18 - Criar Sala	35
Figura 19 - Ativar/Inativar Sala.....	35

Figura 20 - Definir Tempo de Limpeza.....	36
Figura 21 - Definir Limite de Sala.....	36
Figura 22 - Registrar Utilizador.....	37
Figura 23 - Enviar Aviso de Limpeza	37
Figura 24 - Bulk Insert	38
Figura 25 - Lista de Utilizadores	38
Figura 26 - Associar Centro ao Utilizador.....	39
Figura 27 - Mudar permissão utilizador	39

Lista de Abreviaturas e Acrónimos

ESTGV	Escola Superior de Tecnologia e Gestão de Viseu
IPV	Instituto Politécnico de Viseu
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
JS	JavaScript
JSON	JavaScript Object Notation
JWT	JSON Web Token
SQL	Structured Query Language

1. Introdução

1.1. Contextualização do projeto

Este projeto foi realizado no âmbito da unidade curricular Projeto, lecionada no segundo semestre do segundo ano da licenciatura em Engenharia Informática da Escola Superior de Tecnologia e Gestão de Viseu, em colaboração com a empresa Softinsa.

Este projeto visa integrar as unidades curriculares, Aplicações para a Internet II, Programação de Dispositivos Móveis e Base de Dados I.

Pretende-se a criação de uma solução integrada (Aplicação Mobile e Aplicação Web) que permita a gestão e reserva de salas em espaços privados face à sua capacidade e em virtude da alocação máxima permitida por efeitos da pandemia Covid-19.

1.2. Descrição das entidades

1.2.1. ESTGV

Fundado no dia 26 de dezembro de 1979, o Instituto Politécnico de Viseu (IPV), é o primeiro e único estabelecimento de ensino superior público do distrito, assinalando assim um marco importante de desenvolvimento para a região de Viseu.

No total, o Instituto Politécnico de Viseu tem uma oferta formativa de 29 cursos de licenciatura, 29 mestrados, 13 pós-graduações, 6 pós-licenciaturas e 29 CTeSP, fazendo um total de 106 cursos e é composto por 5 Escolas Superiores: Escola Superior de Educação de Viseu, Escola Superior de Tecnologia e Gestão de Viseu, Escola Superior Agrária de Viseu, Escola Superior de Saúde de Viseu e Escola Superior de Tecnologia e Gestão de Lamego.

A ESTGV foi criada em 1985 e o curso de Gestão foi o primeiro a funcionar no ano letivo de 1987/1988.

A ESTGV, enquanto estabelecimento de ensino superior, é um centro de criação, difusão e transmissão de cultura, ciência e tecnologia, articulando as suas atividades nos domínios do ensino, da formação profissional, da investigação e da prestação de serviços à comunidade. A ESTGV rege-se por padrões de qualidade que asseguram formação adequada às necessidades da comunidade em que se insere. Cumpre formar, a nível superior, técnicos devidamente qualificados em setores carecidos na sua área de influência e promover atividades de investigação e desenvolvimento, visando a ligação entre o ensino superior e as empresas, como forma de contributo para solucionar os problemas regionais.

1.2.2. Softinsa

A Softinsa, uma subsidiária da IBM fundada em 1998, é especialista em serviços de consultoria, gestão e desenvolvimento de aplicações. Com 24 anos de história e experiência no mercado português, conta atualmente com uma equipa de mais de 700 profissionais, distribuídos por Lisboa, pelos Centros de Inovação de Tomar e Viseu e pelos Labs especializados do Fundão e Portalegre.

A Softinsa tem como base a inovação e a capacidade de se reinventar continuamente, enquanto procuram satisfazer as necessidades de negócio dos clientes, reforçando a relação de parceria. Os centros permitem impulsionar a capacidade de trabalho de modo a conseguir atingir os objetivos pretendidos.

Através de um programa de desenvolvimento profissional são definidos planos de carreira associados a competências e valores, onde cada colaborador adquire conhecimentos ao ritmo a que se propõe. Na Softinsa é a própria pessoa que define e traça o percurso profissional.

O plano de carreiras, que engloba as áreas de *Consulting, Digital & Applications Development, Cloud & Managed Services, Sales, Project Management e Corporate* reúne as condições necessárias para o desenvolvimento profissional e é orientado de acordo com as expectativas, motivações e experiência de cada colaborador.

A Softinsa tem também uma relação estreita de parceria com o IPV.

1.3. Objetivos

- Desenvolver as capacidades para resolver problemas de uma forma autónoma.
- Integrar e consolidar conhecimentos, envolvendo as várias unidades curriculares.
- Pesquisar informação e ser capaz de a utilizar de forma eficaz.
- Trabalhar em equipa.
- Efetuar gestão do projeto.
- Entregar o projeto multidisciplinar ao cliente.
- Interpretar a legislação na observância dos princípios de direito e deontologia, aplicáveis ao desenvolvimento de produtos na área da engenharia informática.

1.4. Estrutura do relatório

O capítulo 1 corresponde à Introdução.

O capítulo 2 corresponde às Ferramentas e Técnicas.

O capítulo 3 corresponde ao Corpo.

O capítulo 4 corresponde à Conclusão.

O capítulo 5 corresponde à Bibliografia.

2. Ferramentas e Técnicas

2.1. HTML

A HyperText Markup Language ou HTML é a linguagem de marcação padrão para documentos projetados para serem exibidos em um navegador da web. Pode ser auxiliado por tecnologias como Cascading Style Sheets (CSS) e linguagens de script como JavaScript.

Os navegadores da Web recebem documentos HTML de um servidor da Web ou do armazenamento local e processam os documentos em páginas da Web multimídia. HTML descreve a estrutura de uma página web semanticamente e originalmente incluiu dicas para a aparência do documento.

Os elementos HTML são os blocos de construção das páginas HTML. Com construções HTML, imagens e outros objetos, como formulários interativos, podem ser incorporados à página renderizada. HTML fornece um meio para criar documentos estruturados denotando semântica estrutural para texto, como títulos, parágrafos, listas, links, citações e outros itens. Os elementos HTML são delineados por tags, escritas com colchetes angulares. Tags como `` e `<input />` introduzem conteúdo diretamente na página. Outras tags como `<p>` cercam e fornecem informações sobre o texto do documento e podem incluir outras tags como subelementos. Os navegadores não exibem as tags HTML, mas as usam para interpretar o conteúdo da página.

O HTML pode incorporar programas escritos em uma linguagem de script, como JavaScript, que afeta o comportamento e o conteúdo das páginas da web. A inclusão de CSS define a aparência e o layout do conteúdo. O World Wide Web Consortium (W3C), tem incentivado o uso de CSS sobre HTML de apresentação explícita desde 1997. Uma forma de HTML, conhecida como HTML5, é usada para exibir vídeo e áudio, principalmente usando o elemento `<canvas>`, em colaboração com javascript. [1]

2.2. CSS

Cascading Style Sheets (CSS) é uma linguagem de folha de estilo usada para descrever a apresentação de um documento escrito em uma linguagem de marcação como HTML ou XML, sendo uma tecnologia fundamental da World Wide Web.

CSS é projetado para permitir a separação de apresentação e conteúdo, incluindo layout, cores e tipos de letra. Essa separação pode melhorar a acessibilidade do conteúdo; proporcionar maior flexibilidade e controlo na especificação das características de apresentação; permitir que várias páginas da Web compartilhem a formatação especificando o CSS relevante em um arquivo .css separado, o que reduz a complexidade e a repetição no conteúdo estrutural; e ative o cache do arquivo .css para melhorar a velocidade de carregamento da página entre as páginas que compartilham o arquivo e sua formatação.

A separação de formatação e conteúdo também torna viável a apresentação da mesma página de marcação em diferentes estilos para diferentes métodos de renderização, como na tela, impresso, por voz (via navegador baseado em fala ou leitor de tela) e em formato Braille dispositivos táteis. CSS também tem regras para formatação alternativa se o conteúdo for acedido em um dispositivo móvel.

O nome em cascata vem do esquema de prioridade especificado para determinar qual regra de estilo se aplica se mais de uma regra corresponder a um elemento específico. Este esquema de prioridade em cascata é previsível.

As especificações CSS são mantidas pelo World Wide Web Consortium (W3C). O tipo de meios de comunicação da Internet (tipo MIME) text/css é registado para uso com CSS pela RFC 2318 (março de 1998). O W3C opera um serviço de validação CSS gratuito para documentos CSS.

Além do HTML, outras linguagens de marcação suportam o uso de CSS, incluindo XHTML, XML simples, SVG e XUL. [2]

2.3. JavaScript

JavaScript, também conhecido por JS, é uma linguagem de programação sendo uma das principais tecnologias da World Wide Web, juntamente com HTML e CSS. Em 2022, 98% dos sites usam JavaScript no lado do cliente para o comportamento da página da Web, geralmente incorporando bibliotecas de terceiros. Todos os principais navegadores da Web têm um mecanismo JavaScript dedicado para executar o código nos dispositivos dos usuários.

JavaScript é uma linguagem compilada de alto nível, que possui tipagem dinâmica, orientação a objetos baseada em protótipos e funções de primeira classe. É multi-paradigma, suportando estilos de programação orientados a eventos, funcionais e imperativos. Possui interfaces de programação de aplicativos (APIs) para trabalhar com texto, datas, expressões regulares, estruturas de dados padrão e o Document Object Model (DOM).

O padrão ECMAScript não inclui nenhuma entrada/saída (E/S), como rede, armazenamento ou recursos gráficos. Na prática, o navegador da Web ou outro sistema de tempo de execução fornece APIs JavaScript para E/S.

Os mecanismos JavaScript foram originalmente usados apenas em navegadores da Web, mas agora são componentes principais de alguns servidores e de uma variedade de aplicativos. O sistema de tempo de execução mais popular para esse uso é o Node.js.

Embora Java e JavaScript sejam semelhantes em nome, sintaxe e respectivas bibliotecas-padrão, as duas linguagens são bastante distintas em design. [3]

2.4. Bootstrap

Bootstrap é um framework CSS gratuito e de código aberto direcionado ao desenvolvimento web front-end responsivo e mobile-first. Contém HTML, CSS e (opcionalmente) modelos de design baseados em JavaScript para tipografia, formulários, botões, navegação e outros componentes de interface.

Em julho de 2022, Bootstrap é o oitavo projeto mais estrelado no GitHub, com mais de 158.000 estrelas.

Bootstrap é uma biblioteca HTML, CSS e JS que se concentra em simplificar o desenvolvimento de páginas da web informativas (em oposição a aplicativos da web). O objetivo principal de adicioná-lo a um projeto da Web é aplicar as opções de cor, tamanho, fonte e layout do Bootstrap a esse projeto. Como tal, o principal fator é se os desenvolvedores responsáveis acham essas escolhas do seu agrado. Uma vez adicionado a um projeto, o Bootstrap fornece definições básicas de estilo para todos os elementos HTML. O resultado é uma aparência uniforme para prosa, tabelas e elementos de formulário nos navegadores da web. Além disso, os desenvolvedores podem aproveitar as classes CSS definidas no Bootstrap para personalizar ainda mais a aparência de seus conteúdos. Por exemplo, o Bootstrap provisionou tabelas de cores claras e escuras, títulos de página, citações de pull mais proeminentes e texto com destaque.

Bootstrap também vem com vários componentes JavaScript que não requerem outras bibliotecas como jQuery. Eles fornecem elementos adicionais da interface do usuário, como caixas de diálogo, dicas de ferramentas, barras de progresso, menus suspensos de navegação e carrosséis. Cada componente do Bootstrap consiste em uma estrutura HTML, declarações CSS e, em alguns casos, o código JavaScript que acompanha. Eles também estendem a funcionalidade de alguns elementos de interface existentes, incluindo, por exemplo, uma função de preenchimento automático para campos de entrada. [4]

2.5. Node.js

O Node.js é um ambiente de tempo de execução (runtime) JavaScript back-end de código aberto, multiplataforma e executado no mecanismo V8 e executa código JavaScript fora de um navegador da Web, projetado para criar aplicações de rede escaláveis. O Node.js permite que os desenvolvedores usem JavaScript para escrever ferramentas de linha de comando e para scripts do lado do servidor — executando scripts do lado do servidor para produzir conteúdo dinâmico da página da Web antes que a página seja enviada ao navegador da Web do utilizador. Consequentemente, o Node.js representa um paradigma "JavaScript em todos os lugares", unificando o desenvolvimento de aplicativos da Web em torno de uma única linguagem de programação, em vez de linguagens diferentes para scripts do lado do servidor e do lado do cliente.

O Node.js tem uma arquitetura orientada a eventos capaz de E/S assíncrona. Essas escolhas de design visam otimizar a taxa de transferência e a escalabilidade em aplicativos da Web com muitas operações de entrada/saída, bem como para aplicativos da Web em tempo real (por exemplo, programas de comunicação em tempo real e jogos de navegador).

O projeto de desenvolvimento distribuído Node.js era anteriormente governado pela Node.js Foundation, e agora fundiu-se com a JS Foundation formando a OpenJS Foundation, que é facilitada pelo programa Collaborative Projects da Linux Foundation.

Os utilizadores corporativos do software Node.js incluem GoDaddy, Groupon, IBM, LinkedIn, Microsoft, Netflix, PayPal, Rakuten, SAP, Walmart, Yahoo!, e Amazon Web Services. [5]

2.6. Express.js

Express.js, ou simplesmente Express, é uma estrutura de aplicação Web de back-end para Node.js, lançada como software livre e de código aberto sob a licença MIT. Ele é projetado para criar Web apps e APIs. Ele foi chamado de estrutura de servidor “padrão de facto” para Node.js.

O autor original, TJ Holowaychuk, descreveu-o como um servidor inspirado no Sinatra, o que significa que é relativamente mínimo com muitos recursos disponíveis como plugins. Express é o componente de back-end de pilhas de desenvolvimento populares como a pilha MEAN, MERN ou MEVN, juntamente com o software de banco de dados MongoDB e uma estrutura ou biblioteca front-end JavaScript. [6]

2.7. React

React (também conhecido como React.js ou ReactJS) é uma biblioteca JavaScript de front-end gratuita e de código aberto, utilizada para construir interfaces de utilizadores baseadas em componentes de UI. É mantido pelo Meta (anteriormente Facebook) e uma comunidade de desenvolvedores e empresas individuais. O React pode ser usado como base no desenvolvimento de aplicações de página única, móveis ou renderizados por um servidor com frameworks como Next.js. No entanto, o React preocupa-se apenas com a gestão de estado e renderização desse estado para o DOM, portanto, a criação de aplicações React geralmente requer o uso de bibliotecas adicionais para *routing*, bem como certas funcionalidades do lado do cliente. [7]

2.8. Heroku

Heroku é uma plataforma de nuvem como serviço (PaaS) que suporta várias linguagens de programação. Uma das primeiras plataformas em nuvem, Heroku está em desenvolvimento desde junho de 2007, quando suportava apenas a linguagem de programação Ruby. Agora suporta Java, Node.js, Scala, Clojure, Python, PHP e Go. Por esse motivo, o Heroku é considerado uma plataforma poliglota, possuindo recursos para um desenvolvedor criar, executar e dimensionar aplicações de maneira semelhante na maioria dos idiomas. A Heroku foi adquirida pela Salesforce em 2010 por US\$ 212 milhões.

As aplicações executadas no Heroku geralmente têm um domínio exclusivo usado para rotear solicitações HTTP para o *container* de aplicação correto ou dyno. Cada um dos dynos está espalhado por uma "grade de dinamômetro" que consiste em vários servidores. O servidor Git da Heroku lida com pushes de repositórios de aplicativos de usuários permitidos.

Todos os serviços Heroku são hospedados na plataforma de computação em nuvem EC2 da Amazon. [8]

2.9. JWT

JSON Web Token (JWT) é um padrão da Internet proposto para a criação de dados com assinatura opcional e/ou criptografia opcional cuja carga contém JSON que afirma algum número de declarações. Os tokens são assinados usando um segredo privado ou uma chave pública/privada.

Por exemplo, um servidor pode gerar um token com a declaração "conectado como administrador" e fornecê-lo a um cliente. O cliente pode então usar esse token para provar que está conectado como administrador. Os tokens podem ser assinados pela chave privada de uma parte (geralmente a do servidor) para que qualquer parte possa verificar posteriormente que o token é legítimo. Se a outra parte, por algum meio adequado e confiável, estiver de posse da chave pública correspondente, ela também poderá verificar a legitimidade do token. Os tokens são projetados para serem compactos, seguros para URL e utilizáveis especialmente num contexto de logon único (SSO) de navegador da web. As declarações JWT normalmente podem ser usadas para passar a identidade de usuários autenticados entre um provedor de identidade e um provedor de serviços ou qualquer outro tipo de declaração conforme exigido pelos processos de negócios.

O JWT depende de outros padrões baseados em JSON: JSON Web Signature e JSON Web Encryption. [9]

2.10. PostgreSQL

PostgreSQL, também conhecido como Postgres, é um sistema de gestão de banco de dados relacional gratuito e de código aberto que enfatiza a extensibilidade e a conformidade com o SQL. Ele foi originalmente chamado POSTGRES, referindo-se às suas origens como um sucessor do banco de dados Ingres desenvolvido na Universidade da Califórnia, Berkeley. Em 1996, o projeto foi renomeado para PostgreSQL para refletir seu suporte ao SQL. Após uma revisão em 2007, a equipe de desenvolvimento decidiu manter o nome PostgreSQL e o alias Postgres.

O PostgreSQL apresenta transações com propriedades de Atomicidade, Consistência, Isolamento, Durabilidade (ACID), exibições automaticamente atualizáveis, exibições materializadas, gatilhos, chaves estrangeiras e procedimentos armazenados. Ele foi projetado para lidar com uma variedade de cargas de trabalho, de máquinas únicas a data warehouses ou serviços da Web com muitos usuários simultâneos. É o banco de dados padrão para macOS Server e também está disponível para Windows, Linux, FreeBSD e OpenBSD. [10]

Todos os dados da Aplicação Web e da Aplicação Mobile serão guardados no banco de dados PostgreSQL.

2.11. SQL

SQL (Structured Query Language) é uma linguagem específica de domínio usada em programação e projetada para gerir dados num sistema de gestão de banco de dados relacional, ou para processamento de fluxo num sistema de gestão de fluxo de dados relacional. É particularmente útil no tratamento de dados estruturados, ou seja, dados que incorporam relações entre entidades e variáveis.

O SQL oferece duas vantagens principais sobre as APIs de leitura e gravação mais antigas, como ISAM ou VSAM. Em primeiro lugar, introduziu o conceito de aceder muitos registos com um único comando. Em segundo lugar, elimina a necessidade de especificar como alcançar um recorde, por exemplo. com ou sem índice.

Originalmente baseado em álgebra relacional e cálculo relacional de tuplas, o SQL consiste em muitos tipos de instruções, que podem ser informalmente classificadas como sublinguagens, comumente: uma linguagem de consulta de dados (DQL), uma linguagem de definição de dados (DDL), uma linguagem de controle de dados (DCL) e uma linguagem de manipulação de dados (DML). O escopo do SQL inclui consulta de dados, manipulação de dados (inserir, atualizar e excluir), definição de dados (criação de esquema e modificação) e controlo de acesso a dados. Embora SQL seja essencialmente uma linguagem declarativa (4GL), ela também inclui elementos de procedimento.

SQL foi uma das primeiras linguagens comerciais a usar o modelo relacional de Edgar F. Codd. O modelo foi descrito em seu influente artigo de 1970, "A Relational Model of Data for Large Shared Data Banks". Apesar de não aderir inteiramente ao modelo relacional descrito por Codd, tornou-se a linguagem de banco de dados mais utilizada.

SQL tornou-se um padrão do American National Standards Institute (ANSI) em 1986 e da International Organization for Standardization (ISO) em 1987. Desde então, o padrão foi revisado para incluir um conjunto maior de recursos. Apesar da existência de padrões, a maior parte do código SQL requer pelo menos algumas alterações antes de ser portado para diferentes sistemas de banco de dados. [11]

2.12. Android Studio

O Android Studio é o ambiente de desenvolvimento integrado (IDE) oficial para o sistema operativo Android da Google, construído no software IntelliJ IDEA da JetBrains e projetado especificamente para desenvolvimento Android. Está disponível para download em sistemas operativos baseados em Windows, macOS e Linux. É um substituto para o Eclipse Android Development Tools (E-ADT) como o IDE principal para o desenvolvimento de aplicações Android nativos.

O Android Studio foi anunciado em 16 de maio de 2013, na conferência Google I/O. Ele estava na fase de visualização de acesso antecipado a partir da versão 0.1 em maio de 2013, depois entrou na fase beta a partir da versão 0.8, lançada em junho de 2014. A primeira versão estável foi lançada em dezembro de 2014, a partir da versão 1.0.

Em 7 de maio de 2019, Kotlin substituiu o Java como a linguagem preferida do Google para o desenvolvimento de aplicativos Android. Java ainda é suportado, assim também como C++.

Os seguintes recursos são fornecidos na versão estável atual: suporte de compilação baseado em Gradle; *refactoring* e correções rápidas específicas do Android; ferramentas para capturar desempenho, usabilidade, compatibilidade de versão e outros problemas; integração ProGuard e recursos de assinatura de aplicações; assistentes baseados em modelos para criar designs e componentes Android comuns; um editor de layout rico que permite aos utilizadores arrastar e soltar componentes da interface do utilizador, opção para visualizar layouts em várias configurações de tela; suporte para criar aplicações Android Wear; suporte integrado para Google Cloud Platform, permitindo integração com Firebase Cloud Messaging (anteriormente 'Google Cloud Messaging') e Google App Engine; dispositivo virtual Android (emulador) para executar e depurar aplicações no Android Studio.

Depois de uma aplicação ser compilada com o Android Studio, pode ser publicada na Google Play Store. A aplicação deve estar de acordo com a política de conteúdo do desenvolvedor da Google Play Store. [12]

2.13. Kotlin

Kotlin é uma linguagem de programação multiplataforma, estaticamente tipada e de propósito geral com inferência de tipos. Foi projetado para interoperar totalmente com o Java, e a versão JVM da biblioteca padrão do Kotlin depende da Java Class Library, mas a inferência de tipo permite que sua sintaxe seja mais concisa. O Kotlin visa principalmente a JVM, mas também compila para JavaScript (por exemplo, para aplicativos Web front-end usando React) ou código nativo via LLVM (por exemplo, para aplicativos iOS nativos compartilhando lógica de negócios com aplicativos Android). Os custos de desenvolvimento de linguagem são suportados pela JetBrains, enquanto a Kotlin Foundation protege a marca Kotlin.

Em 7 de maio de 2019, o Google anunciou que a linguagem de programação Kotlin agora é sua linguagem preferida para desenvolvedores de aplicativos Android. Desde o lançamento do Android Studio 3.0 em outubro de 2017, o Kotlin foi incluído como uma alternativa ao compilador Java padrão. O compilador Android Kotlin produz bytecode Java 8 por padrão (que é executado em qualquer JVM posterior), mas permite que o programador escolha o Java 9 até 18, para otimização, ou permite mais recursos; tem suporte de interoperabilidade de classe de registo bidirecional para JVM, introduzido no Java 16, considerado estável a partir do Kotlin 1.5.

Kotlin tem suporte para a web; compilando para JavaScript, enquanto o Kotlin/JS mais recente (baseado em IR) está em beta a partir da versão 1.5.30. Kotlin/Native (por exemplo, suporte ao silício da Apple) é considerado beta desde a versão 1.3.[13]

3. Corpo

3.1. Conceção

3.1.1. Requisitos

- A solução deverá permitir o registo de salas e a alocação máxima disponível por sala.
- Deverá permitir, a cada momento, a parametrização da % de alocação permitida face às regras covid-19 implementadas para cada sala.
- Utilizadores e permissões deverão ser parametrizáveis no *backoffice* da aplicação mediante respetiva permissão. Na primeira vez que um utilizador entrar na aplicação tem de alterar a password.
- Os utilizadores devem receber um email de confirmação de registo antes de puderem utilizar a solução, sendo que o registo será efetuado pelo administrador da plataforma. Deve ser considerado o upload de um ficheiro para criação de utilizadores em *bundle*.
- Os utilizadores devem poder fazer a inscrição na sala, alocando um *slot* horário para uma determinada data.
- Deve ser possível cancelar uma reserva em qualquer altura.
- Se uma reunião passar o tempo previsto e houver tempo disponível imediatamente a seguir, então, deve ser possível prolongar o tempo de reserva da sala.
- Ao reservar a Sala o utilizador receberá informação da capacidade da mesma.
- Quando a data/hora da marcação estiver a chegar, o utilizador deve receber uma notificação.
- Os utilizadores devem poder cancelar a sua reserva da sala.
- Deve ser possível parametrizar em *backoffice* o tempo mínimo de limpeza de sala, que deve ser contabilizado para todas as reservas, se o tempo for parametrizado como 0, desconsidera-se a limpeza.

-
- Para as notificações geradas pela solução deve ser considerado o modelo de *push notifications*.

Aplicação Web:

- A aplicação web servirá apenas para parametrização dos espaços e utilizadores e visualização de *dashboards* com a seguinte informação:

- % de alocação diária, com visão mensal;
- # de reservas por range de datas;
- # de utilizadores registados;
- % de salas mais utilizadas face à capacidade;
- "realtime Data" com a marcação das salas permitindo a gestão mais eficiente dos mesmos;
- Necessidade atual de limpeza de sala.

Aplicação Mobile – Utilizador:

- A aplicação mobile deverá permitir a autenticação do utilizador confirmado
- Deverá permitir a seleção da sala pretendida, através da leitura de QR code ou através da pesquisa da sala, que permita a sua reserva da mesma
- Deverá permitir o cancelamento da reserva, ou alteração de reservas futuras.

Aplicação Mobile – Sala:

Faz parte desta solução a colocação de um equipamento (telemóvel ou tablet) na entrada da sala que disponibilize o calendário semanal com visualização das reservas efetuadas e permita a leitura do QR code para marcação da sala de forma rápida.

- Deve permitir a configuração da sala correspondente.
- Deve permitir manter o ecrã sempre ligado sem necessidade de desbloqueio.

- O *refresh* da informação na visível no ecrã deve ser online, ou seja, sempre que exista uma marcação/cancelamento de reserva ou abertura/fecho de sala, a aplicação deve mostrar em real time o respetivo estado atual.

Bónus:

- Considerando que a sala possa ficar indisponível, por motivos de força maior, a aplicação deverá permitir ao administrador inativar o espaço e gerar uma notificação para todos os utilizadores com reservas onde informe que a sala ficou indisponível mais a justificação do administrador e no ecrã presente na sala, deve constar a mesma informação.

- O pessoal da limpeza não tem de "entrar" na aplicação. Contudo, a título de Bónus, podem permitir o acesso à aplicação por parte do pessoal da limpeza, de modo que recebam notificações e possam definir que uma sala já está limpa e desinfetada.

3.1.2. Checklist de requisitos

Requisitos Funcionais - Geral	Validado
Criar Utilizadores	X
Ativar/Inativar utilizadores	X
Associar utilizadores a permissões da aplicação	X
Associar utilizadores a pelo menos um centro	X
Obrigar a atualização de password no primeiro login	X
Enviar email ao utilizador após a sua criação na aplicação	X
Criar utilizadores por <i>bulkinsert</i> através de upload de ficheiro de excel com estrutura pré-definida	X
Criar centros geográficos (centro de Viseu, Fundão, etc)	X
Registar salas e associar ao centro geográfico	X
Definir alocação máxima disponível por sala	X
Permitir inativar o centro	X
Permitir inativar salas	X
Parametrizar em <i>backoffice</i> o tempo mínimo de limpeza de sala, que deve ser contabilizado para todas as reservas	X
Disponibilizar <i>dashboard</i> com (% de alocação diária, com visão mensal, # de reservas por range de datas, # de utilizadores registados, % de salas	X

mais utilizadas face à capacidade,"realtime Data" com a marcação das salas permitindo a gestão + eficiente dos mesmos, necessidade atual de limpeza de sala)	
Aplicação Mobile - Utilizador	
Permitir a leitura da sala através de QR Code	X
Fazer reserva na sala mediante disponibilidade da mesma	X
Fazer a alteração da reserva na sala	X
Alterar a reserva na sala, mediante disponibilidade da mesma	X
Cancelar as reservas atuais ou futuras	X
Garantir que as reservas passadas, não podem ser removidas, exceto pelo administrador da aplicação	X
Informar utilizador da capacidade máxima da sala, no momento da reserva	X
Aplicação Mobile - Sala	
Permitir a configuração da sala correspondente	X
Manter o ecrã sempre ligado sem necessidade de desbloqueio	X
Refrescar a informação visível no ecrã automaticamente	X
Bónus	
Permitir ao administrador, quando inativa o espaço, gerar uma notificação para todos os utilizadores com reservas onde informe que a sala ficou	

indisponível + justificação do administrador e no ecrã presente na sala, deve constar a mesma informação	
O pessoal da limpeza não tem de "entrar" na aplicação. Contudo, a título de Bónus, podem permitir o acesso à aplicação por parte do pessoal da limpeza, de modo que recebam notificações e possam definir que uma sala já está limpa e desinfetada.	

3.1.3. Modelo de dados

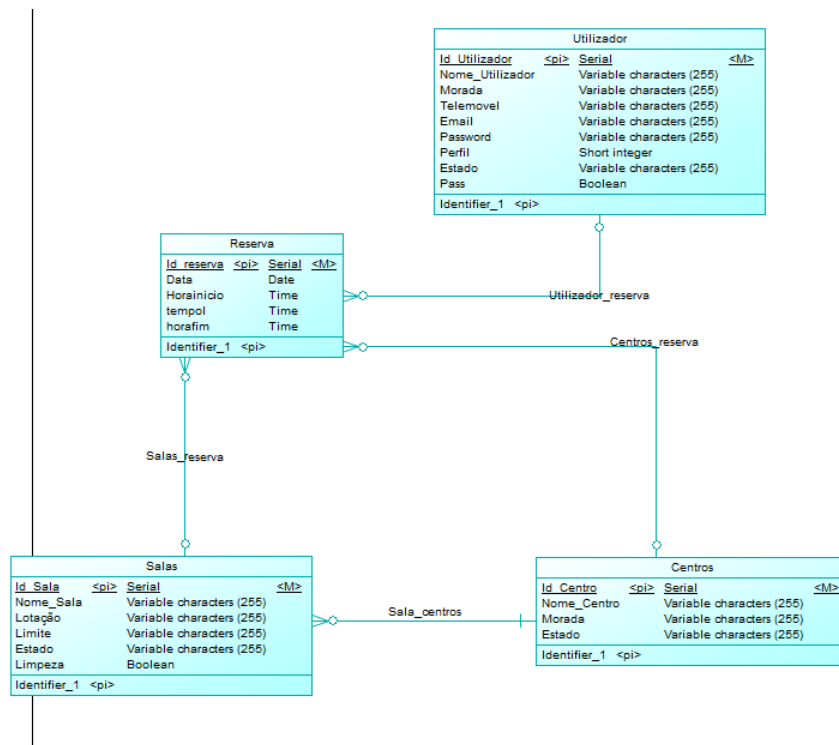


Figura 1 - Modelo de Entidade-Relacionamento

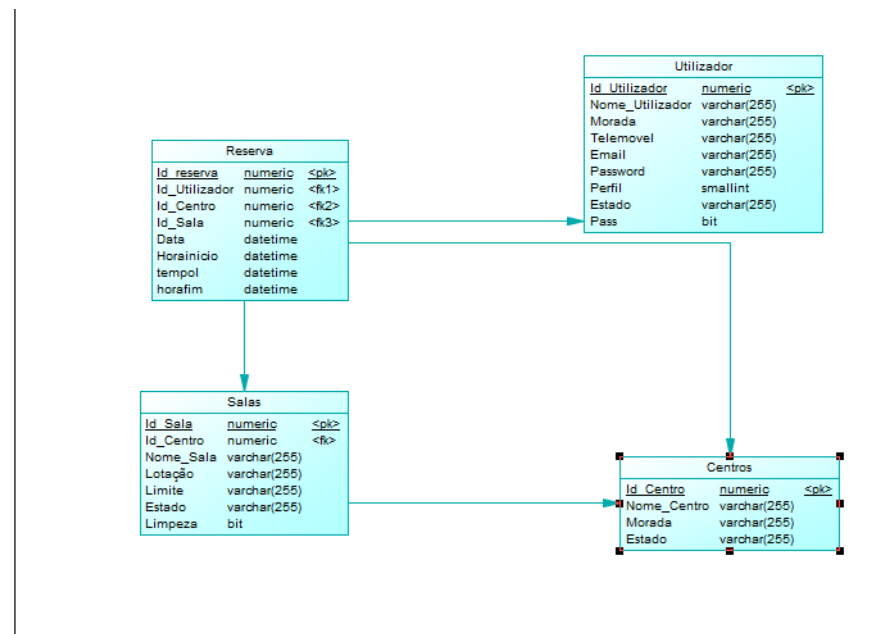


Figura 2 - Modelo Físico

3.2. Aplicação Mobile

A aplicação Mobile foi desenvolvida na linguagem de programação Kotlin, no ambiente de desenvolvimento Android Studio.

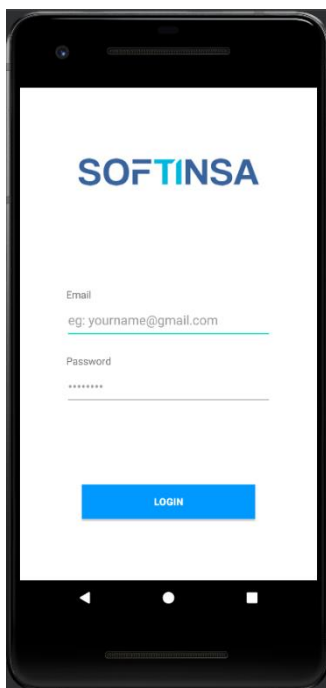


Figura 3 - Login (Mobile)

Nesta atividade temos a página de login, onde se preenchem os dados das credenciais (Email e Password) e clica-se no botão Login de modo a aceder à página inicial da aplicação.

O Login é um “post” com as variáveis email e password. Se o response code for igual a 200, guardamos o id do utilizador numa shared preferences. É realizada a verificação se é a primeira vez que o utilizador está a fazer login, se tal for verdade, o utilizador é redirecionado para a página de alterar a password, senão o utilizador é redirecionado para o menu da aplicação.

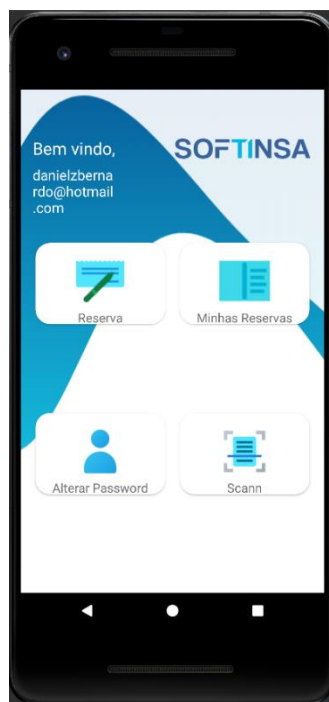


Figura 4 - Menu

Através da atividade “Menu”, podemos aceder às seguintes atividades: “Reserva”, “Minhas Reservas”, “Alterar Password” e “Scann”.



Figura 5 - Lista Centros

Ao aceder à atividade “Reserva” é nos disponibilizada a lista dos centros, ao seleccionarmos uma sala, somos redirecionados para a página onde se faz a reserva.

A smartphone screen displaying a form titled "Reservar sala". The form has a blue header bar with a back arrow and the title. Below the header, there are three input fields: "Data da reunião" with the value "20-07-2022", "Início da reunião" with the value "09:00", and "Fim da reunião" with the value "12:00". Each field has a dropdown arrow on the right. At the bottom of the form is a blue button labeled "CONFIRMAR". The phone's navigation bar is visible at the bottom.

Figura 6 - Reservar Sala

Nesta atividade podemos fazer a reserva da sala seleccionada, escolhemos a data da reunião, a hora de início da reunião e a hora de fim da reunião.

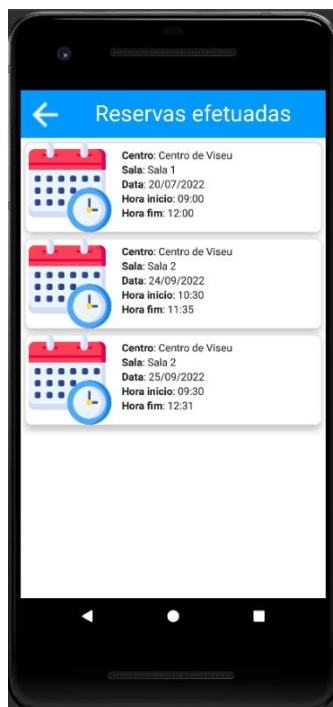


Figura 7 - Reservas Efetuadas

Ao acedermos às “Minhas Reservas”, através do menu, somos redirecionados para a página das reservas efetuadas onde podemos visualizar as nossas reservas e a sua informação.



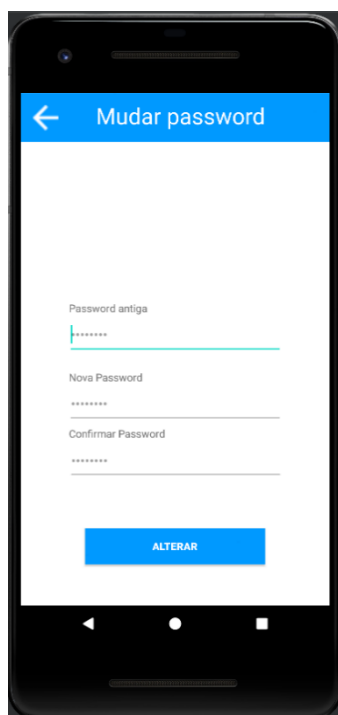
Figura 8 - Detalhes Reserva

Ao selecionarmos uma das reservas presentes na atividade “Reservas Efetuadas” podemos observar os detalhes da mesma, e nos permite Editar e/ou Cancelar a Reserva.



Figura 9 - Editar Reserva

Ao selecionarmos “Editar Reserva”, somos redirecionados para uma página que nos permite alterar a data da reunião, a hora de início da reunião e a hora de fim da reunião.



**Figura 10 - Mudar Password
(Mobile)**

Ao seleccionarmos “Alterar Password” no menu, a app redireciona o utilizador para uma página que o permita mudar a password da sua conta, pedindo que introduza a password antiga e introduza e confirme a sua nova password.

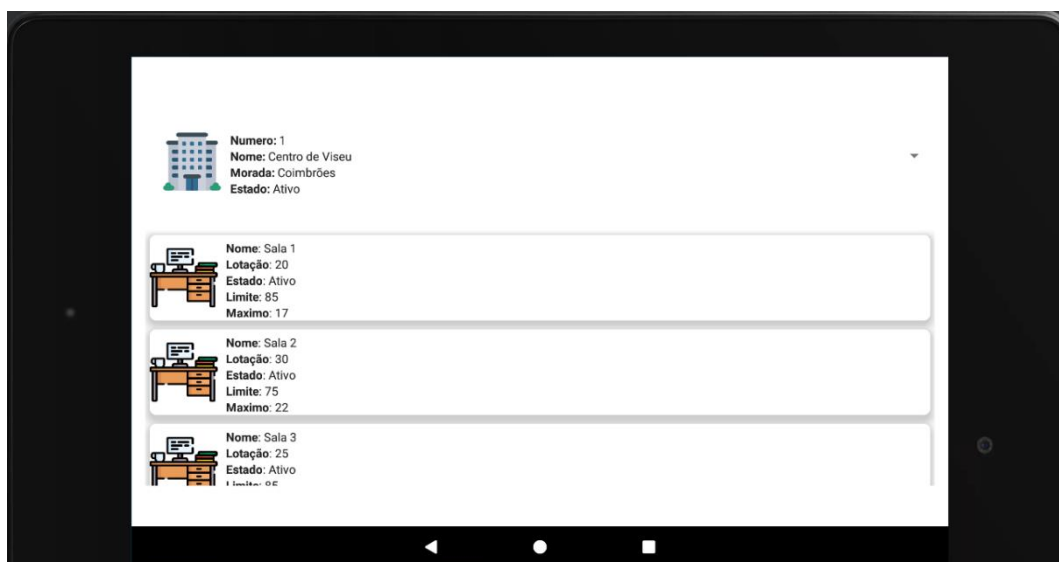


Figura 11 - Exemplo de tela de dispositivo Fixo

Ao olharmos para a tela do dispositivo fixo de um centro, podemos visualizar as salas presentes no mesmo e a sua respetiva lotação, estado, limite percentual de pessoas, e máximo de pessoas que podem estar presentes na mesma.

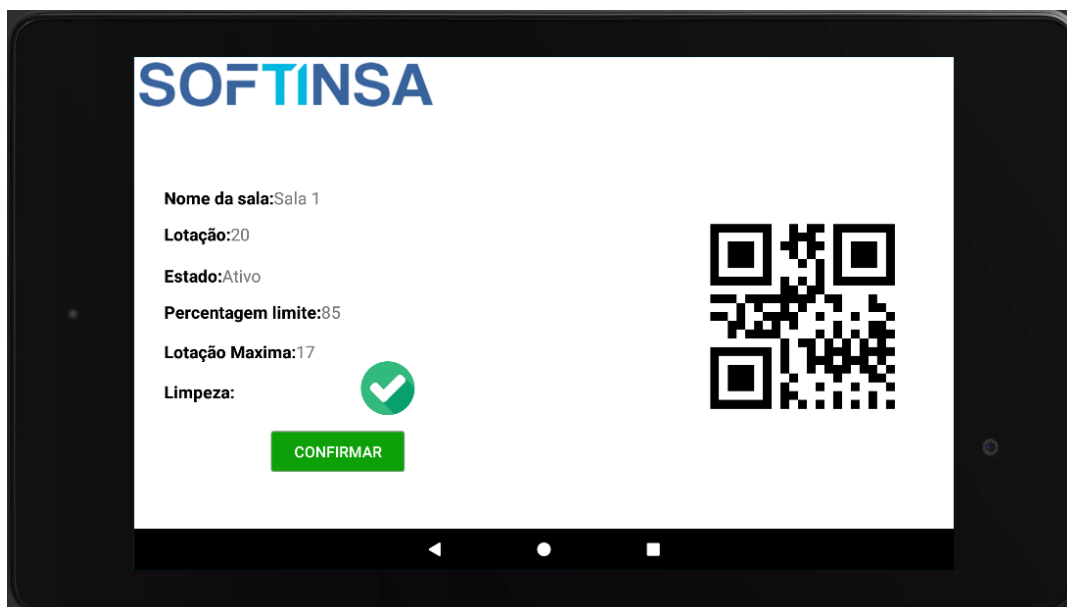


Figura 12 - Exemplo Sala 1 dispositivo Fixo

Ao selecionarmos uma das salas presentes no ecrã fixo do centro, somos redirecionados para a página da respetiva sala, onde podemos visualizar os seus detalhes de informação (lotação, estado, limite percentual de pessoas, e máximo de pessoas) e o seu QR code que nos redireciona para a página de Reserva da mesma.

3.3. Aplicação Web

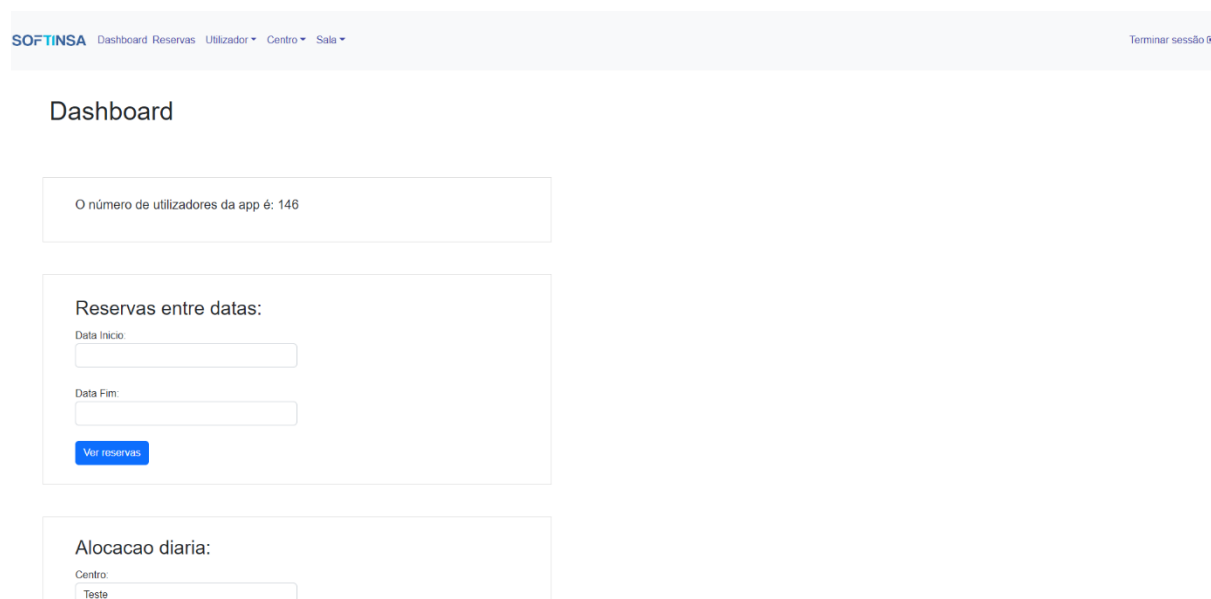
A aplicação Web foi desenvolvida em HTML, CSS e JavaScript, com recurso às tecnologias e ferramentas Bootstrap, Node.js, Express.js, React, Heroku, JWT, no ambiente de desenvolvimento Visual Studio Code.



The image shows a login form for SOFTINSA. At the top, the SOFTINSA logo is displayed in blue. Below the logo is a white rectangular box containing the login fields. Inside the box, there is an 'Email:' label followed by a text input field, a 'Password:' label followed by a text input field, and a blue 'Login' button. Below the box, the text '©Todos os direitos reservados.' is centered.

Figura 13 - Login (Web)

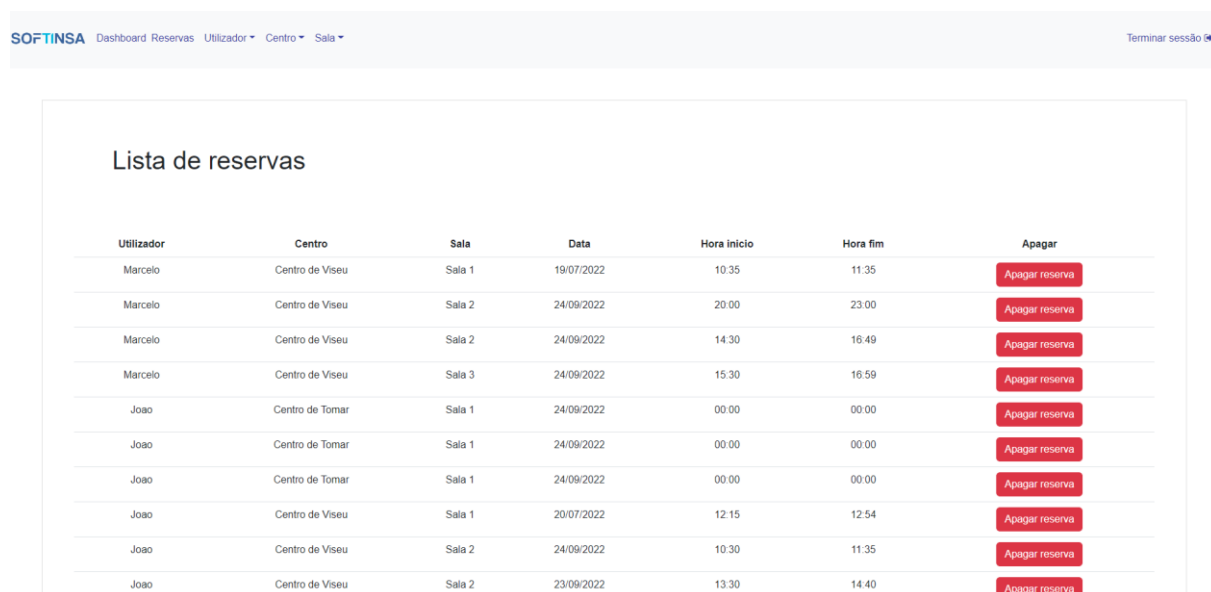
Na página de Login preenchem-se as caixas com os dados das credenciais (Email e Password) e clica-se no botão Login de modo a aceder à página inicial da aplicação.



The image shows the SOFTINSA dashboard. At the top, there is a header bar with the SOFTINSA logo on the left, navigation links 'Dashboard', 'Reservas', 'Utilizador', 'Centro', and 'Sala' in the middle, and a 'Terminar sessão' link on the right. Below the header, the word 'Dashboard' is displayed. The main content area contains three white boxes. The first box shows 'O número de utilizadores da app é: 146'. The second box is titled 'Reservas entre datas:' and contains two date input fields labeled 'Data Inicio:' and 'Data Fim:', followed by a blue 'Ver reservas' button. The third box is titled 'Alocacao diaria:' and contains a text input field labeled 'Centro:' with the value 'Teste' entered.

Figura 14 - Dashboard

Na página inicial encontramos o dashboard da aplicação Web, e a partir desta página podemos aceder todas as outras páginas onde se encontram as funcionalidades da aplicação.



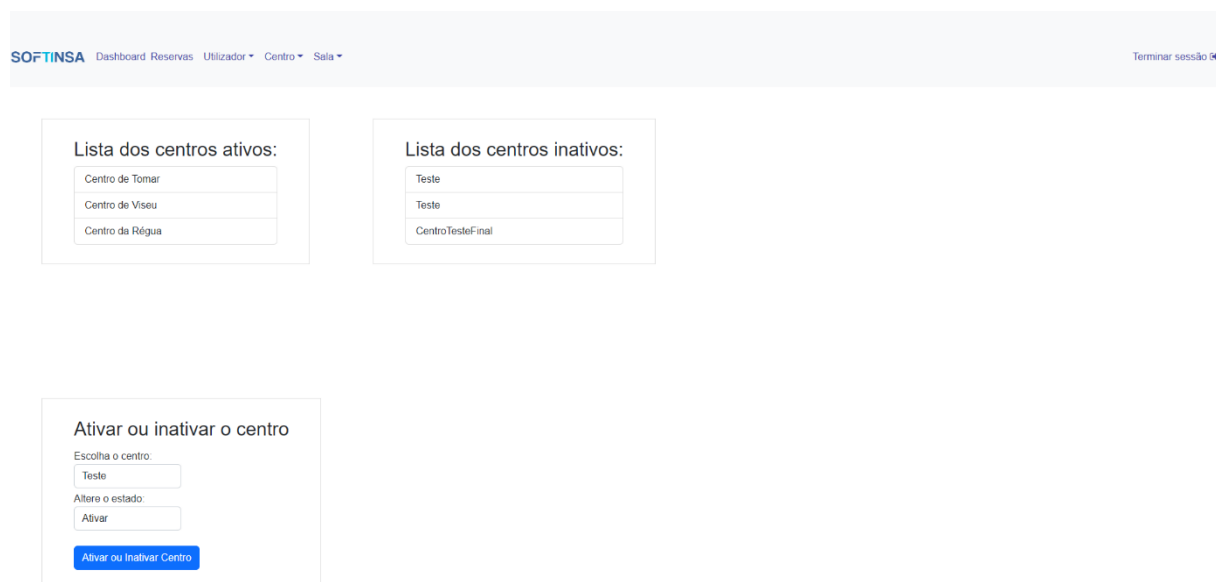
SOFTINSA Dashboard Reservas Utilizador Centro Sala Terminar sessão

Lista de reservas

Utilizador	Centro	Sala	Data	Hora início	Hora fim	Apagar
Marcelo	Centro de Viseu	Sala 1	19/07/2022	10:35	11:35	Apagar reserva
Marcelo	Centro de Viseu	Sala 2	24/09/2022	20:00	23:00	Apagar reserva
Marcelo	Centro de Viseu	Sala 2	24/09/2022	14:30	16:49	Apagar reserva
Marcelo	Centro de Viseu	Sala 3	24/09/2022	15:30	16:59	Apagar reserva
Joao	Centro de Tomar	Sala 1	24/09/2022	00:00	00:00	Apagar reserva
Joao	Centro de Tomar	Sala 1	24/09/2022	00:00	00:00	Apagar reserva
Joao	Centro de Tomar	Sala 1	24/09/2022	00:00	00:00	Apagar reserva
Joao	Centro de Viseu	Sala 1	20/07/2022	12:15	12:54	Apagar reserva
Joao	Centro de Viseu	Sala 2	24/09/2022	10:30	11:35	Apagar reserva
Joao	Centro de Viseu	Sala 2	23/09/2022	13:30	14:40	Apagar reserva

Figura 15 - Lista de Reservas

Ao selecionar a opção “Lista de reservas” acedemos à respetiva página, onde temos disponibilizada toda a informação referente às reservas guardadas na base de dados, e se apresenta a opção “Apagar reserva”, que elimina a mesma.



SOFTINSA Dashboard Reservas Utilizador Centro Sala Terminar sessão

Lista dos centros ativos:

Centro de Tomar
Centro de Viseu
Centro da Régua

Lista dos centros inativos:

Teste
Teste
CentroTesteFinal

Ativar ou inativar o centro

Escolha o centro:

Teste

Altere o estado:

Ativar

Ativar ou Inativar Centro

Figura 17 - Lista dos Centros

Ao acedermos ao menu dropdown “Centro” podemos abrir a página “Criar Centro” que permite, ao utilizador, criar um novo centro introduzindo o seu nome e morada.

Ao acedermos ao menu dropdown “Centro” podemos abrir a página “Lista dos Centros” onde é apresentada a lista dos centros ativos, a lista dos centros inativos, e a funcionalidade de ativar/inativar um centro.

The screenshot shows the 'Criar aqui a sala' (Create room here) form. At the top, there is a navigation bar with the SOFTINSA logo and a breadcrumb trail: Dashboard > Reservas > Utilizador > Centro > Sala. A 'Terminar sessão' (Log out) link is on the right. The form itself is titled 'Criar aqui a sala' and contains the following fields: 'Nome:' (text input), 'Escolha o centro:' (dropdown menu with 'Teste' selected), 'Lotação:' (text input), 'Limite da Sala:' (text input), and 'Estado:' (text input). A blue 'Criar Sala' button is at the bottom left of the form.

Figura 18 - Criar Sala

Ao acedermos ao menu dropdown “Sala” podemos abrir a página “Criar Sala” que permite ao utilizador criar uma sala, atribuindo os dados nome, centro, lotação, limite da sala e estado.

The screenshot shows the 'Ativar ou inativar a sala' (Activate or deactivate room) form. The navigation bar is identical to the previous figure. The form is titled 'Ativar ou inativar a sala' and contains the following fields: 'Escolha o centro:' (dropdown menu with 'Teste' selected), 'Escolha a sala:' (text input), and 'Altere o estado:' (text input). A blue 'Guardar alterações' (Save changes) button is at the bottom left of the form.

Figura 19 - Ativar/Inativar Sala

Ao acedermos ao menu dropdown “Sala” podemos abrir a página “Ativar/Inativar Sala” que nos permite alterar o estado de uma respetiva sala.

SOFTINSA Dashboard Reservas Utilizador Centro Sala Terminar sessão

Defina o limite da sala

Escolha o centro:

Teste

Escolha a sala:

Limite:

Guardar alterações

Figura 21 - Definir Limite de Sala

Outra funcionalidade presente nas opções do menu dropdown “Sala” é a definição do limite percentual de pessoas numa sala específica.

SOFTINSA Dashboard Reservas Utilizador Centro Sala Terminar sessão

Defina o tempo de limpeza

Escolha o tempo de limpeza:

00:05

Gravar alterações

Figura 20 - Definir Tempo de Limpeza

Nesta página podemos definir o tempo de limpeza de uma sala.

SOFTINSA Dashboard Reservas Utilizador Centro Sala Terminar sessão

Escolha a sala que deve ser limpa

Escolha o centro:

Teste

Escolha a sala:

Aviso de limpeza enviado!

Figura 23 - Enviar Aviso de Limpeza

Nesta página escolhemos o centro e uma sala específica e enviamos um aviso de limpeza da mesma.

SOFTINSA Dashboard Reservas Utilizador Centro Sala Terminar sessão

Registe aqui o utilizador

Nome:

Morada:

Telemovel:

Email:

Perfil:

Admin

Estado:

Ativo

Registar Utilizador

Figura 22 - Registar Utilizador

Ao acedermos ao menu dropdown “Utilizador” podemos aceder à página “Registar Utilizador” onde preenchemos um formulário com os dados do utilizador (Nome, morada, telemóvel, email, perfil e estado), sendo posteriormente criado na base de dados.

SOFTINSA

Dashboard Reservas Utilizador Centro Sala

Terminar sessão

Lista de utilizadores

Utilizador	Centro	Morada	Telemovel	Email	Estado	Ativar	Inativar
andre		casaandre	922222222	andre@gmail.com	Inativo	Ativar	Inativar
daniel		casa	911111111	daniel@gmail.com	Inativo	Ativar	Inativar
andre		casaandre	922222222	an@gmail.com	Ativo	Ativar	Inativar
a		a	911111111	daniel@gmail.com	Ativo	Ativar	Inativar
manel ze		rua random	915656565	novo@hotmail.com		Ativar	Inativar
andre		casaandre	922222222	joaquim@gmail.com	Ativo	Ativar	Inativar
				manel@gmail.com		Ativar	Inativar
				manel1@gmail.com		Ativar	Inativar
				manel2@gmail.com		Ativar	Inativar
manel1 ze		rua random	915656565	novo1@hotmail.com		Ativar	Inativar

Figura 25 - Lista de Utilizadores

Ao acedermos ao menu dropdown “Utilizador” podemos aceder à página “Lista de Utilizadores” onde podemos visualizar a lista dos utilizadores presentes na base de dados e os seus respetivos detalhes.

SOFTINSA

Dashboard Reservas Utilizador Centro Sala

Terminar sessão

Read CSV file

Escolher Ficheiro

Não foi escolhido nenhum ficheiro

There are no records to display

IMPORT CSV

Figura 24 - Bulk Insert

Ao acedermos ao menu dropdown “Utilizador” podemos aceder a esta página que nos permite importar um ficheiro CSV com dados de utilizadores.

SOFTINSA Dashboard Reservas Utilizador Centro Sala Terminar sessão

Associe um centro ao utilizador

Escolha o utilizador:

Altere o centro:

Mudar centro do utilizador

Figura 26 - Associar Centro ao Utilizador

Ao acedermos ao menu dropdown “Utilizador” podemos aceder à página que permite associar um centro a um utilizador.

SOFTINSA Dashboard Reservas Utilizador Centro Sala Terminar sessão

Mude aqui a permissão do utilizador

Escolha o utilizador:

Altere o perfil do utilizador:

Mudar permissão

Figura 27 - Mudar permissão utilizador

Ao acedermos ao menu dropdown “Utilizador” podemos aceder à página que permite ao utilizador administrador alterar as permissões de um respetivo utilizador.

4. Conclusão

Este projeto foi, sem dúvida, o maior desafio que tivemos que ultrapassar ao longo do nosso percurso acadêmico e para conseguirmos concretizar este projeto com sucesso foi necessária uma persistência constante na aprendizagem destas tecnologias Full Stack e uma forte comunicação e trabalho de equipa entre todos os membros.

Todos os requisitos (menos os Bónus) foram implementados e todos os objetivos foram atingidos. Apesar de todas as funcionalidades terem sido implementadas, uma coisa que poderia ser melhorada no nosso projeto seria o design de ambas as aplicações (Mobile e Web), pois estas não cumprem todas as regras de usabilidade, o que torna a experiência do utilizador menos satisfatória e/ou eficiente.

Ao realizarmos este projeto, sentimos que estamos bastante mais capazes de enfrentar futuros desafios no mundo do trabalho.

5. Bibliografia

- [1] - <https://en.wikipedia.org/wiki/HTML>
- [2] - <https://en.wikipedia.org/wiki/CSS>
- [3] - <https://en.wikipedia.org/wiki/JavaScript>
- [4] - [https://en.wikipedia.org/wiki/Bootstrap_\(front-end_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework))
- [5] - <https://en.wikipedia.org/wiki/Node.js>
- [6] - <https://en.wikipedia.org/wiki/Express.js>
- [7] - [https://en.wikipedia.org/wiki/React_\(JavaScript_library\)](https://en.wikipedia.org/wiki/React_(JavaScript_library))
- [8] - <https://en.wikipedia.org/wiki/Heroku>
- [9] - https://en.wikipedia.org/wiki/JSON_Web_Token
- [10] - <https://en.wikipedia.org/wiki/PostgreSQL>
- [11] - <https://en.wikipedia.org/wiki/SQL>
- [12] - https://en.wikipedia.org/wiki/Android_Studio
- [13] - [https://en.wikipedia.org/wiki/Kotlin_\(programming_language\)](https://en.wikipedia.org/wiki/Kotlin_(programming_language))