

# Plataforma de Gestão de Espaços numa Escola de Ensino Superior

## *Platform for Space Management in a Higher Education School*

Caio Cesar Hideo Nakai  
UTFPR, Brasil  
Instituto Politécnico de Bragança  
Portugal  
a42882@alunos.ipb.pt

José Eduardo Fernandes, Maria João V. Pereira  
Centro de Investigação em Digitalização e Robótica  
Inteligente  
Instituto Politécnico de Bragança  
Portugal  
jef@ipb.pt, mjoao@ipb.pt

*Resumo* — Nas instituições de ensino superior há uma cada vez maior variedade de atividades de ensino e investigação a desenvolver que necessitam de espaços apropriados e adaptados às necessidades. Sendo que algumas ações têm prioridade e restrições mais exigentes que outras há que fazer uma gestão eficiente dos recursos nomeadamente dos espaços a reservar. Nesse sentido, é apresentado neste artigo uma plataforma de gestão de espaços aplicada a uma escola do ensino superior, mas que pode ser facilmente adaptada a outras instalações.

*Palavras Chave* – *Gestão de espaços; aplicações de gestão escolar;*

*Abstract* — In higher education institutions there is an increasing variety of teaching and research activities to be developed that need appropriate spaces and adapted to needs. Since some actions have more demanding priorities and restrictions than others, it is necessary to make an efficient management of resources, namely the spaces to be reserved. In this sense, this article presents a space management platform applied to a higher education school but which can be easily adapted to other installations.

*Keywords* - *Space management; school management applications*

### I. INTRODUÇÃO

A escola de ensino superior, onde foi desenvolvido este trabalho, dispõe de uma área global de 14 000m<sup>2</sup>, comporta 112 gabinetes de docentes, 1 auditório, 2 anfiteatros, 25 salas de aula, 5 salas de informática, 1 biblioteca e 20 laboratórios, que ocupam uma área global superior a 3000m<sup>2</sup>, para além de zonas de convívio e de apoio técnico/administrativo [1]. Dada a dimensão das instalações e as adaptações que ocorrem para suprir as necessidades de cada ano letivo em termos de espaços e equipamentos, torna-se evidente a dificuldade em manter as informações de todos os espaços atualizadas. Atualmente, para solicitar a reserva de um espaço, é necessário entrar em contacto com a funcionária responsável pela gestão de espaços que, consultando os espaços não ocupados para aulas, regista a nova reserva. Em relação às informações dos espaços, atualmente, apenas o nome do espaço e o número de lugares são

armazenados em uma plataforma online para efeitos de elaboração de horários escolares.

As limitações derivadas deste processo e dificuldades sentidas levaram à decisão de o reformular, recorrendo para tal à incorporação no processo de uma nova aplicação informática que permitisse a automatização e simplificação das ações dos diversos intervenientes.

Para aprimorar o processo atual de reserva dos espaços da escola, foi estabelecido que a aplicação tomaria a forma de uma ferramenta web que possibilitasse a gestão dos espaços, dos pedidos de reserva e que armazenasse informações sobre recursos e pessoas associadas a cada espaço. Esse processo de reserva teria de levar em conta os espaços já ocupados semanalmente pelas aulas. Todas as restantes reservas feitas na plataforma não têm um carácter recorrente. Mais ainda, foi estabelecido que, na interação com os utilizadores, a navegação pelos espaços existentes seria realizada com base nas plantas do edifício relativas a salas e pisos.

Este documento retrata os desafios da conceção, a implementação e a instalação deste sistema. O resto do documento está estruturado da seguinte forma: a secção II retrata trabalhos ou sistemas relacionados; a secção III apresenta a conceção do sistema expondo a análise de requisitos, arquitetura do sistema e tecnologias associadas; a secção IV debruça-se sobre o desenvolvimento do sistema, testes e instalação do sistema; a secção V apresenta os resultados obtidos e a secção VI conclui apontando possibilidades de evolução do sistema em trabalhos futuros.

### II. ESTADO DA ARTE

Segundo [2], a gestão de espaços é considerada um módulo de um sistema Computer-Aided Facility Management (CAFM). Um CAFM é um sistema que armazena, analisa e apresenta a informação sobre as instalações de uma empresa [3]. Os trabalhos [4] [5] são ambos CAFM e partem do mesmo princípio que é manter informações precisas sobre a utilização dos espaços, plantas e localização dos recursos. No que diz

respeito à gestão de espaços, contribuem com resultados positivos que evidenciam os benefícios de um sistema de gestão espaços e instalações.

Além dos trabalhos citados acima, foram encontradas duas soluções comerciais parecidas: “touchONE” [6] e “Skedway” [7]. O touchONE é um sistema de reserva de salas, locais de trabalho, hotéis, grandes instalações corporativas, universidades, etc. O sistema utiliza painéis de reserva interativos na frente de cada espaço, que indica a ocupação do espaço no momento e possui integração com outros sistemas como o Office 365 e o G-Suite. Também é possível utilizar através de aplicações mobile e web. Além disso, o sistema também oferece funcionalidades como check-in através de QR codes, que se encontram nos painéis interativos, relatar problemas no espaço, suporte para 17 idiomas diferentes e outras. Para comprar esta solução é necessário entrar em contacto com a empresa, o website não possui nenhuma informação em relação ao custo da solução. A solução “Skedway” é uma plataforma para gestão de espaços e secretárias. A plataforma também possui hardware dedicado para utilização da plataforma, assim como disponibiliza aplicações mobile e web. Entre as funcionalidades estão: integrações com Office 365 e o G-Suite, check-in através de QR codes, relatar problemas no espaço, calendário para encontrar a data para a reserva, etc. A solução é vendida através de assinaturas mensais e o valor varia de acordo com a quantidade de espaços e secretárias, existe um plano gratuito, entretanto é limitado a apenas 3 espaços.

### III. CONCEÇÃO

Apesar das duas soluções comerciais apresentadas serem utilizadas por grandes empresas como por exemplo, a Microsoft e possuírem diversas funcionalidades, foi decidido desenvolver um novo sistema de raiz feito à medida da escola. Optou-se por uma aplicação monolítica dado que se pretende uma aplicação específica, com poucos utilizadores sem necessidade de grande escalabilidade nem flexibilidade de ações. Entre os objetivos do sistema que foi desenvolvido estavam a integração com a Application Programming Interface (API) dos sumários (os sumários é uma aplicação já existente onde se registam os horários das aulas), a navegação com base nas plantas da escola e a autenticação através do servidor Lightweight Directory Access Protocol (LDAP), este servidor também realiza a autenticação em outras plataformas da escola. Essas três funcionalidades são muito específicas e não existem nas soluções comerciais apresentadas. Além disto, desenvolver a própria solução permite que sejam adicionadas funcionalidades consoante a necessidade.

#### A. Perfis de utilizador e Requisitos Funcionais

O sistema possui três tipos de utilizadores, sendo eles o funcionário, o gestor de espaços e gestor do sistema. Todos estes tipos de utilizadores serão autenticados no sistema através do LDAP, após o qual poderão realizar ações apropriadas ao seu tipo de utilizador:

- O funcionário representa o utilizador, por exemplo, um professor, cujo principal interesse é poder solicitar e

confirmar a reserva de um espaço. Adicionalmente, poderá consultar informações dos espaços, consultar horário de ocupação dos espaços; solicitar reserva dos espaços; cancelar a própria solicitação de reserva dos espaços e verificar o estado das próprias reservas;

- O gestor de espaços é o responsável por aprovar, rejeitar e editar as solicitações de reservas dos espaços sob a sua responsabilidade. Nesse contexto, pode alterar o valor dos atributos dos espaços, gerir os recursos alocados aos espaços, consultar o estado de todas as solicitações pendentes de reservas; consultar o estado e reservar espaços;
- O gestor do sistema é o responsável, de forma geral, por gerir o sistema: pode gerir os espaços, os pisos e as áreas. É o gestor de sistema que controla o perfil de gestor de espaços e atribui a responsabilidade de gerir cada espaço.

A essência do suporte dos requisitos funcionais pelo sistema é esquematizada pelo diagrama de casos de uso (com elementos de modelação dispostos de forma a facilitar a visão geral) mostrado na Fig.1.

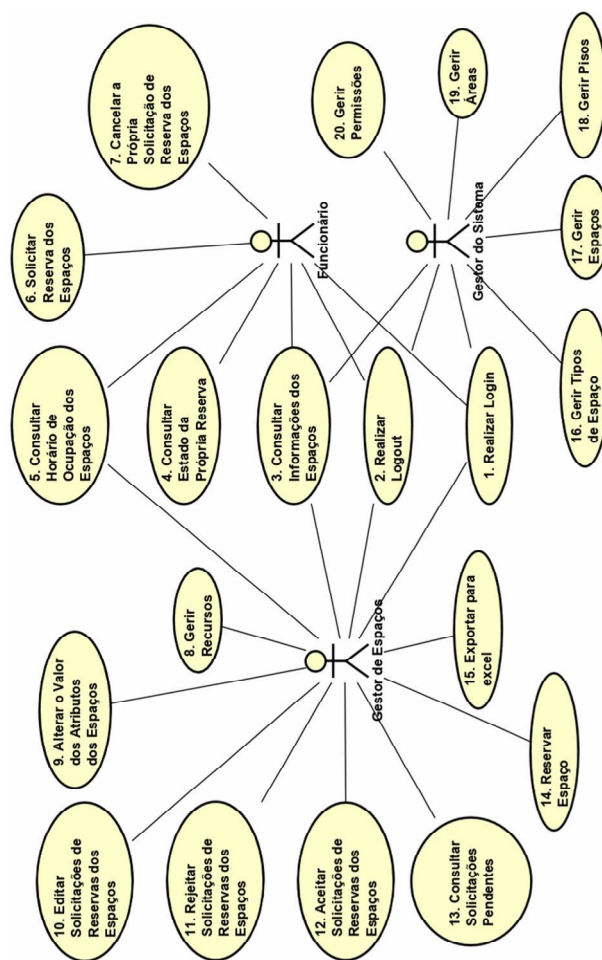


Figura 1. Diagrama de casos de uso

A visão geral das classes conceituais e suas relações é exposta pelo diagrama de domínio mostrado na Fig. 2. Para lidar com os diferentes pisos da escola, foi criada uma classe conceitual para o piso, possibilitando que caso haja a construção de um novo piso na escola, seja possível adicioná-lo ao sistema, tornando-o mais flexível. Cada piso pode possuir zero ou mais áreas. As áreas são diferentes regiões de um determinado piso e podem possuir zero ou mais espaços. O espaço é a classe conceitual que possui a maior quantidade de relacionamentos. Cada espaço possui exatamente um tipo (por exemplo "sala de aula"), o espaço pode possuir zero ou vários recursos materiais (por exemplo "ar condicionado"). O espaço também possui relacionamentos com o funcionário: possui nenhum ou um funcionário responsável, possui nenhum ou muitos funcionários técnicos e, no mínimo, um ou muitos gestores de espaços. Cada espaço possui exatamente um tipo (por exemplo "sala de estudo").

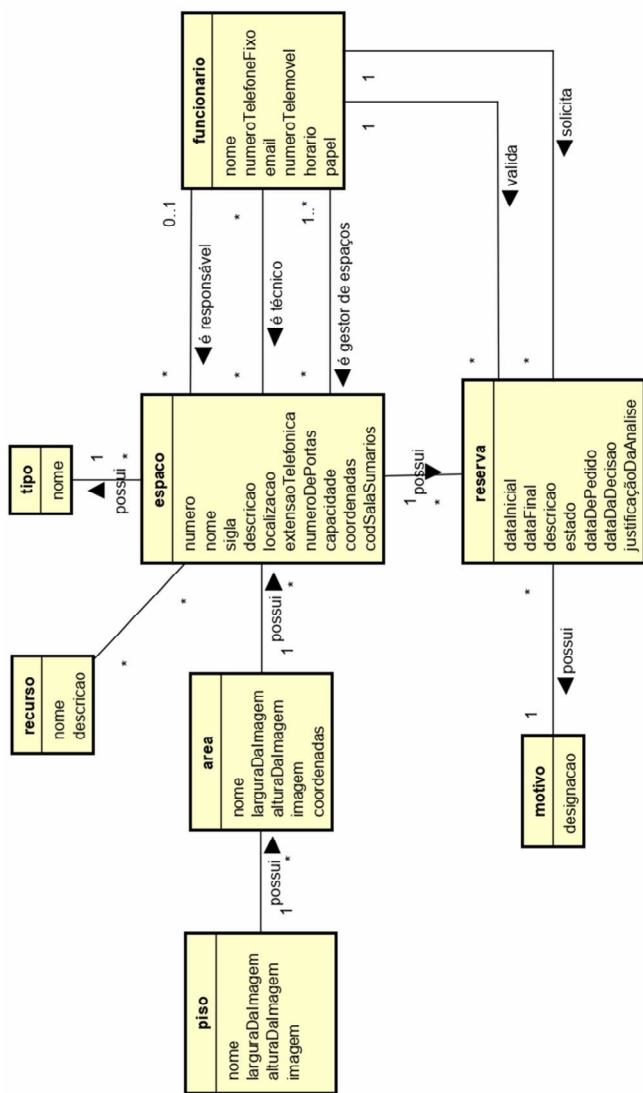


Figura 2. Diagrama de domínio

## B. Arquitetura e Tecnologias

A arquitetura monolítica mostrada na Fig.3 é a arquitetura do sistema desenvolvido. Foi escolhido este tipo de arquitetura porque a quantidade de utilizadores não é significativa e para aplicações com poucos utilizadores as arquiteturas monolíticas constituem uma abordagem mais expedita [8]. Além disso, o desenvolvimento e deploy são mais simples, por existir apenas uma única base de código, existem menos problemas relacionados com a integração e configuração [9]. Na Fig.3 podemos ver que o servidor comunica com o servidor LDAP, pois é ele que realiza a autenticação dos utilizadores. Além disso, o servidor também comunica com a API Sumários para coletar os eventos e registá-los na base de dados.

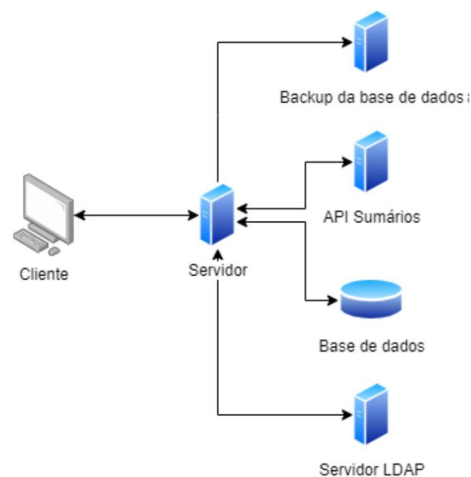


Figura 3. Arquitetura do sistema

Da diversidade de soluções tecnológicas, foram feitas escolhas para a implementação da arquitetura proposta para o sistema. Comparando o React ao Angular e ao Vue, outros dois frameworks JavaScript populares, nenhum deles se destaca o suficiente para que possam ser recomendados para a maioria dos casos de uso em termos de desenvolvimento web [10]. A decisão para implementar a aplicação em React foi tomada devido ao fato de já existir um conhecimento prévio da biblioteca. No que respeita à escolha do Node.js, entre as vantagens da utilização do Node.js estão: uma comunidade de programadores ativa; o node package manager (a maior biblioteca de software do mundo); e a utilização do JavaScript tanto no front-end quanto no back-end, facilitando e acelerando o desenvolvimento. A escolha de utilizar o MongoDB deu-se principalmente pela eficiência no acesso aos dados, pelo ganho de flexibilidade para realizar alterações caso necessário, pois facilita a integração com outros sistemas e por tornar o desenvolvimento mais ágil devido a facilidade de aprendizagem.

## IV. IMPLEMENTAÇÃO

A metodologia de desenvolvimento seguiu uma abordagem iterativa e incremental com sprints duas semanas, ao longo das quais, foram desenvolvidos e refinados vários aspetos do sistema, em particular das funcionalidades esperadas com base numa arquitetura de sistema estabilizada. Apresenta-se de seguida alguns dos pontos mais relevantes do desenvolvimento

da aplicação, alguns dos quais respeitantes a funcionalidades principais, tais como a navegação por planta e gestão de reservas, e outros aspetos relacionados com o deploy, testes e documentação.

### A. Desenvolvimento

#### 1) Navegação por planta

Entre os pontos a serem destacados está a implementação das imagens selecionáveis para navegar entre os diferentes espaços da escola. Um dos objetivos principais e inovadores deste trabalho consiste no facto da ferramenta se basear nas plantas reais das instalações da escola. Portanto, para consultar as informações ou realizar a solicitação de reserva dos espaços o utilizador seleciona um piso e escolhe a área onde o espaço está localizado, como é mostrado na Fig.4. Após selecionar a área, ele seleciona o espaço desejado clicando na sua localização na planta da escola. Para auxiliar o utilizador a encontrar o espaço desejado, ao passar o rato por cima dos espaços/áreas o nome do espaço é mostrado ao lado. Além disso, a ferramenta também suporta as operações de CRUD (create, read, update and delete) para os pisos, áreas e espaços. Para implementar as funcionalidades citadas acima, foi utilizada a tag HTML “path” para desenhar as áreas e espaços selecionáveis nas imagens, dessa forma ao definir o path, as áreas selecionáveis na imagem, são sempre as mesmas independente do seu tamanho. No entanto, definir o path, ou seja, as áreas que serão selecionáveis na imagem, é um tanto quanto complexo para se fazer manualmente. Para isso são utilizados editores de imagens ou editores de Scalable Vector Graphics. Sendo assim, para o gestor do sistema adicionar novas áreas ou espaços é necessário que ele indique o path, ou seja, é necessário utilizar algum editor como dito anteriormente para obter este valor. Como isso não é uma tarefa trivial de ser feita pela primeira vez, foi adicionado um vídeo que explica como obter o path utilizando o editor de imagens open source GIMP.

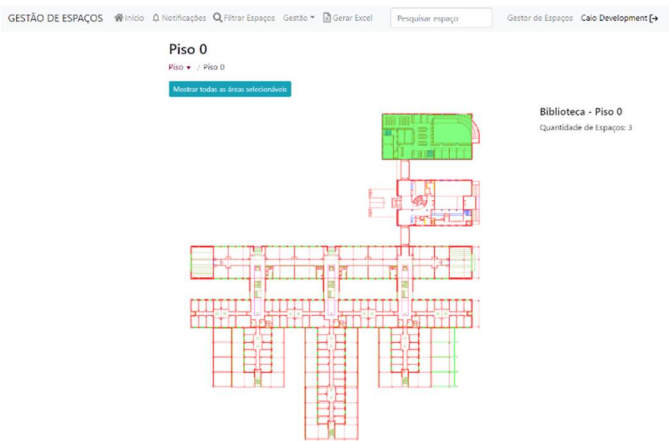


Figura 4. Navegação baseada em planta

#### 2) Horários

A implementação da tabela de horários de um espaço (mostrada na Fig.5) é outro ponto a ser destacado. Cada coluna da tabela representa uma data específica e as linhas representam um intervalo de horário de trinta minutos. As células da tabela

podem ser selecionadas de três formas diferentes: ao clicar diretamente na célula, ao clicar no horário para selecionar toda a linha e ao clicar na data para selecionar toda a coluna. A tabela também exibe as reservas e solicitações de reservas, as quais são posicionadas nas linhas referentes aos horários que foram reservadas/solicitadas. A tabela foi implementada de raiz, isto é, não foram utilizadas bibliotecas de terceiros, pois nenhuma das bibliotecas encontradas possuía as funcionalidades e layout esperados. Além disso, ao desenvolver os próprios componentes torna-se mais fácil acrescentar novas funcionalidades visto que não é necessário estudar o código desenvolvido por outra pessoa para verificar se é possível realizar as alterações desejadas.

Figura 5. Tabela de horários

#### 3) Procura de espaço por filtros

Para implementar a funcionalidade de procurar espaços utilizando filtros como tipo de espaço, capacidade, recursos, técnicos, responsável e disponibilidade de horários foi necessário recorrer a queries mais complexas. A página de procurar espaços com filtros é mostrada na Fig.6. Apesar disso, foi possível implementar de forma muito simples, com expressões regulares e métodos JavaScript de objetos e arrays.

Figura 6. Procura de espaços através de filtros

#### 4) Gestão de reservas

A implementação da gestão de solicitações de reservas também é um ponto interessante a destacar. A Fig. 7 é a janela de gestão de reservas que pode ser acedida pelos gestores de espaço. Para fazer a implementação das “toasts”, as “caixas”

verdes mostradas na Fig. 7, foi utilizado um componente da biblioteca React Bootstrap. Esta biblioteca também fornece vários outros componentes, como botões, inputs, formulários e outros, sendo que muitos deles também foram utilizados em outras partes da aplicação. As solicitações de reserva podem possuir três cores diferentes, sendo elas: verde, significa que a notificação pode ser aceite e não possui conflito com nenhuma outra solicitação; vermelha, significa que a notificação não pode ser aceite de imediato, pois há um pedido de reserva em conflito e por último, amarela, que significa que a solicitação está em conflito com uma reserva já aceite. Além disso, também é possível filtrar as solicitações por utilizador e espaço assim como ordenar as solicitações pelas mais antigas, mais recentes e por nome.



Figura 7. Gestão de reservas

### 5) Estatísticas

Foi também implementada uma funcionalidade que permite o download de ficheiros Excel das reservas aprovadas que permite que possam ser feitas análises interessantes como por exemplo: qual a média de duração das reservas; quantas reservas foram feitas durante o semestre; quem são as pessoas que mais fizeram reservas; quais são os espaços mais reservados; quais são os horários mais reservados; qual o tempo médio que um determinado espaço fica reservado por dia. Além destas análises, podem ser feitas muitas outras. Essas informações são úteis pois podem auxiliar nas tomadas de decisões, por exemplo, verificar que um espaço é muito pouco utilizado para dar aulas e decidir utilizá-lo para outra finalidade.

### B. Deploy e Testes

Para realizar o deploy da ferramenta foi utilizado o Nginx, um servidor proxy reverso e HTTP, tanto para o front-end como para o back-end. A escolha se deu pelo facto do Nginx ser mais fácil de ser utilizado quando comparado ao Apache, outro servidor HTTP. Além disso, assim como o Apache, o Nginx é muito popular e é utilizado por grandes empresas como a Uber, Airbnb, Instagram, etc. O Nginx foi configurado numa Virtual Machine (VM) com o sistema operativo Ubuntu 20.04. Nesta VM encontram-se os projetos do front-end e do back-end. É importante que o back-end corra exactamente nessa VM, pois a autenticação feita pelo LDAP exige que as requisições sejam feitas por endereços IP. A aplicação foi então disponibilizada no endereço próprio para realização dos testes.

Para utilizar o protocolo Hypertext Transfer Protocol Secure (HTTPS) e tornar a comunicação entre o cliente e o servidor

mais segura, foram utilizados certificados TLS emitidos pela autoridade de certificação gratuita “Let’s encrypt”. O protocolo TLS possui o objetivo principal de fornecer privacidade e integridade dos dados durante a comunicação entre duas aplicações [11]. Os certificados emitidos pela “Let’s encrypt” são válidos durante 90 dias, dessa forma chaves roubadas e certificados emitidos incorretamente são válidos por menos tempo. O facto de os certificados serem válidos durante um período curto não é um problema, pois eles são renovados automaticamente antes de serem expirados.

A aplicação React foi desenvolvida utilizando o ambiente create react app, sendo esta a forma suportada oficialmente e sugerida na documentação gerida pelo Facebook. Com este ambiente é possível gerar a versão minimizada dos pacotes da aplicação com um único comando e desta forma, fazer o deploy do front-end é uma tarefa simples e rápida. Após gerar a versão minimizada do projeto só é preciso reiniciar o servidor Nginx para que o front-end seja atualizado. Para fazer o deploy da aplicação Node.js foi utilizado o gerenciador de processos PM2 para Node.js. O PM2 possui um balanceador de carga integrado, permite manter as aplicações ativas o tempo todo e recarregá-las sem downtime. O balanceador de carga funciona ao utilizar o modo cluster. Este é um modo especial ao iniciar uma aplicação Node.js que permite o início de múltiplos processos e distribui a carga entre eles. A quantidade de processos que podem ser utilizados depende do número de núcleos da CPU disponíveis.

Foram realizadas reuniões para testes após a realização do deploy, possibilitando que qualquer máquina com conexão a internet pudesse aceder a ferramenta. Com o sistema acessível, foram testados todos os casos de uso com as pessoas que realmente irão utilizar o sistema, sendo elas os funcionários que realizam a gestão dos pedidos de reserva atualmente e os docentes. Após as primeiras reuniões de teste foram sugeridas algumas alterações para melhorar a experiência do utilizador e tornar a ferramenta mais user-friendly. Após a realização das alterações sugeridas outras reuniões para testes foram feitas para concluir a validação do sistema.

### C. Documentação

A documentação, apesar de muitas vezes esquecida, é uma das partes mais importantes de um projeto de software, pois facilita o entendimento do projeto e reduz drasticamente a quantidade de tempo necessário para as pessoas que não participaram do processo de desenvolvimento, realizem contribuições ou manutenções. Para documentar a API foi utilizada a plataforma colaborativa para desenvolvimento de API’s Postman [12]. A documentação possui todos os endpoints da API, o método Hypertext Transfer Protocol (HTTP) de cada um e uma pequena descrição sobre o que é esperado do endpoint e um exemplo de resposta. Também foram documentados os requisitos necessários para correr os projetos do front-end e do back-end. Para além dos requisitos necessários, também foi documentada a lista de todos os comandos necessários para correr ambos os projetos e outros comandos que possam ser úteis, como por exemplo, como aceder a linha de comando da base de dados, entre outros. Além desta documentação, também foi documentado o processo de deploy das alterações realizadas no código e a localização dos projetos (back-end e front-end), da

configuração do Nginx, da base de dados, dos certificados Transport Layer Security (TLS) e dos backups.

## V. RESULTADOS

O sistema desenvolvido possibilita a gestão dos espaços, pedidos de reserva e armazena as informações dos recursos e pessoas associadas a cada espaço. Além da gestão, o sistema também possibilita que as reservas sejam feitas de forma coordenada com a aplicação dos horários/sumários. Portanto, espera-se que a mesma possa facilitar a gestão das informações e reservas dos espaços da escola.

O facto de a ferramenta permitir o upload das imagens para a navegação, significa que pode ser feito o upload de plantas de outras construções e não apenas da escola em causa, ou seja, torna-se mais fácil a adaptação que seria necessária realizar para utilizar a ferramenta em outras escolas.

Entre o que se acredita que poderia ter sido feito de forma diferente, está a entrega das funcionalidades que foram sendo desenvolvidas ao longo do tempo, a partir do momento em que estavam prontas. Em engenharia de software esta prática é chamada de entrega contínua. Dessa forma, acredita-se que a fase de testes com os funcionários que utilizarão o sistema poderia ter sido mais rápida. Além disso, poderia ter sido implementado testes automatizados, para auxiliar na redução de erros e no esforço exigido para realizar todos os testes necessários.

## VI. TRABALHO FUTURO

A fim de aprimorar a ferramenta desenvolvida de forma geral, são sugeridas as implementações para trabalhos futuros: automatização do deploy (pois apesar de ter sido implementado um script shell para facilitar ainda mais o processo deploy, a automatização pode reduzir ainda mais o tempo necessário); automatização de testes (a automatização dos testes contribui para a redução dos erros); recuperação automática dos backups da base de dados (pois apesar do automatismo para gerar os backups e a necessidade de recuperação não ser algo recorrente, caso algum problema ocorra e seja necessário fazer a recuperação dos dados, a mesma tem que ser feita de forma manual).

## AGRADECIMENTOS

Este trabalho foi apoiado pela FCT - Fundação para a Ciência e Tecnologia no âmbito do Projeto: UIDB / 05757/2020. Agradecimentos ao Instituto Politécnico de Bragança por fornecer os recursos necessários para realização deste trabalho.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] "Descrição Geral da Instituição". Acessado em: 30 de Janeiro de 2021. [Online]. Disponível: <https://estig.ipb.pt/index.php/estig/a-estig>
- [2] P. Poór, "Strengthening of boundary processes in a company using specific computer-aided facility management software," 2014 IEEE 12th International Symposium on Applied Machine Intelligence and Informatics (SAMi), Herlany, Slovakia, 2014, pp. 257-261, doi: 10.1109/SAMI.2014.6822418.
- [3] T. Schurle, A. Boy, D. Fritsch, "Geographic Information Systems and Facility Management," International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 32, no. 4, 1998, pp. 562-568.
- [4] C. D'Urso, "Information Integration for Facility Management," in IT Professional, vol. 13, no. 6, pp. 48-53, Nov.-Dez. 2011, doi: 10.1109/MITP.2011.100.
- [5] Iadanza, Ernesto, Luschi, Alessio. (2019). "An integrated custom decision-support computer aided facility management informative system for healthcare facilities and analysis. Health and Technology", 2020, vol. 10, pp. 135-145, doi: 10.1007/s12553-019-00377-6.
- [6] "TouchONE". Acessado em 4 de Fevereiro de 2021. [Online]. Disponível: <https://www.touchone.eu/EN/Homepage>
- [7] "Skedway". Acessado em 4 de Fevereiro de 2021. [Online]. Disponível: <https://skedway.com/pt/>
- [8] M. Villamizar et al., "Evaluating the monolithic and the microservice architecture pattern to deploy web applications in the cloud," 2015 10th Computing Colombian Conference (10CCC), Bogota, 2015, pp. 583-590, doi: 10.1109/ColumbianCC.2015.7333476.
- [9] K. Gos, W. Zabierowski, "The Comparison of Microservice and Monolithic Architecture," 2020 IEEE XVIth International Conference on the Perspective Technologies and Methods in MEMS Design (MEMSTECH), Lviv, Ukraine, 2020, pp. 150-153, doi: 10.1109/MEMSTECH49584.2020.9109514.
- [10] E. Wohlgethan, "Supporting Web Development Decisions by Comparing Three Major JavaScript Frameworks: Angular, React and Vue.js", Hochschule für angewandte Wissenschaften Hamburg, 2018.
- [11] E. Rescorla, T. Dierks, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, Ago. de 2008. doi: 10.17487/RFC5246. URL: <https://rfc-editor.Org/rfc/rfc5246.txt>.
- [12] "Postman". Acessado em 4 de Fevereiro de 2021. [Online]. Disponível: <https://www.postman.com/>