

Parents Relationship Management

Projeto em Informática da Licenciatura em Engenharia Informática

Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

Junho, 2020

Alexandre Lopes

André Amarante

Edgar Morais

Francisco Fontinha

Orientador: Prof. Doutor Diogo Gomes

Sumário

As associações de pais são uma forma organizada e institucional legalmente prevista de pais e encarregados de educação fazerem ouvir a sua voz e participar no processo educativo dos filhos.

Geralmente as associações não utilizam plataformas específicas para comunicar ou guardar informações, pois a grande maioria das plataformas, nomeadamente CRMs, não tem um plano gratuito ou são demasiado complexas para o tipo de utilização que as associações de pais esperam realizar nessas plataformas. Sendo assim, acabam por utilizar apenas folhas de cálculo para guardar dados e a comunicação é feita pessoalmente, por email, telemóvel ou uma rede social, o que a torna pouco eficiente e leva a que a informação fique dispersa e pouco organizada.

Este relatório descreve o desenvolvimento de uma plataforma *open-source* e totalmente gratuita desenhada especificamente para associações de pais, independentemente do tipo de academia em que atuam.

A plataforma permite aos utilizadores criarem casos de forma organizada para discutir facilmente questões que considerem relevantes, importar e exportar dados, verificarem e registarem pagamentos de cotas e efetuar transições de cargos, tudo em formato digital e num só lugar.

Tabela de Conteúdos

Sumário.....	ii
Tabela de Conteúdos	1
Tabela de Figuras	4
Introdução	6
Contexto.....	6
Objetivos	6
Limitações.....	7
Estrutura do relatório	8
Estado da arte.....	9
Aspetos em comum com serviços de CRM	9
Fatores de diferenciação	9
Análise de Requisitos.....	11
Requisitos Funcionais.....	11
Requisitos Não Funcionais	12
Requisitos de Usabilidade	12
Requisitos de Desempenho.....	12
Requisitos de Segurança e Integridade de Dados	12
Requisitos de Interface com Sistemas Externos.....	13
Atores.....	13
Casos de Utilização	13
Arquitetura	17
Modelo do domínio.....	17
Modelo tecnológico	18
Versão inicial.....	18

Versão final.....	19
Implementação.....	21
Componentes da Implementação	21
Google Firebase.....	21
Aplicação Web	27
Gravatar	27
Funcionalidades	28
Configuração da associação.....	28
Registo	34
Autenticação	38
Perfil	39
Cotas	45
Casos	47
Cargos / transição de mandato.....	51
Notificações	52
Envio de emails	52
Exportação de dados.....	53
Testes e resultados	55
Protótipos	55
Protótipo Estático	55
Protótipo Funcional	57
Testes de usabilidade.....	60
Testes automáticos	60
Conclusão	61
Resultados.....	61
Benefícios e limitações da solução	62
Trabalho futuro.....	63

Referências	64
Apêndices	67
Apêndice A: Complementos do projeto	67

Tabela de Figuras

Figura 1 - Casos de utilização de utilizadores	14
Figura 2 - Caso de utilização do Administrador do Sistema.....	15
Figura 3 - Modelo do domínio	17
Figura 4 - Modelo tecnológico (versão inicial)	18
Figura 5 - Modelo tecnológico (versão final)	19
Figura 6 - Página inicial de configuração.....	28
Figura 7 - Formulário de configuração de uma associação pré-existente	29
Figura 8 - Formulário de configuração de uma nova associação	30
Figura 9 - Exemplo de ficheiro JSON de parâmetros adicionais.....	31
Figura 10 - Exemplo de ficheiro JSON de cargos	32
Figura 11 - Ficheiro de importação de membros	33
Figura 12 - Ficheiro de importação de educandos	33
Figura 13 - Página de registo.....	34
Figura 14 - Página de login após o registo bem-sucedido	35
Figura 15 - Página de confirmação do pagamento da primeira cota	36
Figura 16 - Página de aprovação de registos.....	37
Figura 17 - Janela de detalhes do registo	37
Figura 18 - Página de login	38
Figura 19 - Página de perfil	39
Figura 20 - Consultar informações de um educando.....	40
Figura 21 - Adicionar um educando	40
Figura 22 - Página de definições avançadas	41
Figura 23 - Confirmação da alteração de email	42
Figura 24 - Confirmação da eliminação de conta	43
Figura 25 - Página de administração	44
Figura 26 - Painel de exportação de dados	44
Figura 27 - Página de cotas.....	45
Figura 28 - Criação de um novo registo de pagamento.....	46
Figura 29 - Mensagem de confirmação das alterações ao sair da página de cotas	47
Figura 30 - Página de casos.....	48
Figura 31 - Página de casos com casos arquivados visíveis	48
Figura 32 - Janela de criação de um novo caso	49

Figura 33 - Página individual de um caso.....	49
Figura 34 - Atualização de membros de um caso	50
Figura 35 - Janela de atribuição de cargos.....	51
Figura 36 - Página de cargos	52
Figura 37 - Protótipo da dashboard	55
Figura 38 - Protótipo da página de casos.....	56
Figura 39 - Protótipo da página de um caso	57
Figura 40 - Protótipo da página de instalação	58
Figura 41 - Protótipo da página de login	58
Figura 42 - Protótipo da página inicial.....	59

Introdução

As associações de pais são associações constituídas por pais e encarregados de educação que se querem fazer ouvir e participar mais ativamente no processo de educação dos seus filhos/educandos. Estas são independentes do estado, partidos políticos, organizações religiosas ou quaisquer outras organizações. São autónomas relativamente à definição das suas normas internas e estatutos, eleição dos seus órgãos sociais, de gestão e de administração e à elaboração dos seus planos de atividades [1].

As associações têm como principal objetivo a defesa e promoção dos interesses dos encarregados de educação relativamente à educação dos seus filhos/educandos. Este objetivo inclui o desenvolvimento de ações em conjunto com direções de escolas e professores de forma a promover a formação tanto dos pais como dos alunos. Também poderão promover atividades de apoio à família e à sua participação na vida dos alunos.

Contexto

As associações de pais, geralmente, não possuem nenhum local designado para reuniões nem nenhuma plataforma centralizada onde seja possível divulgar informação, discutir casos ou partilhar ficheiros sobre a administração, problema ou evento que esteja a ser organizado pela associação. Isto leva a uma maior dificuldade de comunicação entre os membros da associação e acaba por influenciar negativamente a exposição e eficácia das ações e discussões realizadas na associação.

O nosso projeto centra-se precisamente nesta necessidade, uma vez que está direcionado de forma a resolver ou melhorar as condições de comunicação e gestão da associação. O nosso projeto consiste numa plataforma web adaptada a plataformas móveis que permite a divulgação de eventos, discussão de casos, partilha de ficheiros e gestão dos membros e dos seus cargos. A nossa plataforma está adaptada à rotatividade elevada dos membros da associação e permite a exportação de dados guardados na plataforma, de forma a que a associação não perca dados caso deixe de utilizar esta plataforma.

Objetivos

Devido a alguns aspetos específicos relativos às associações de pais, a nossa plataforma deverá ser adaptada de forma a que esteja de acordo com esses aspetos. Este projeto tem uma natureza *open-source*, pelo que o código do projeto estará disponível publicamente. Para facilitar a instalação da plataforma também faz parte dos objetivos a realização de um manual de instruções da plataforma.

As associações de pais, geralmente, não possuem fundos suficientes para a hospedagem de uma plataforma num servidor ou na nuvem, pelo que a plataforma terá de ser hospedada num serviço gratuito.

A plataforma permitirá a criação de casos. Estes poderão ser privados ou públicos, pelo que, no caso de ser privado, será possível filtrar os utilizadores que têm acesso às informações desse caso. Também será possível realizar comentários/observações e adicionar anexos/ficheiros relacionados com o caso.

Para que um encarregado de educação passe a ser membro da associação, é necessário o pagamento de uma cota anual. Outro objetivo da nossa plataforma é permitir manter uma lista das cotas que tenham de ser pagas pelos encarregados de educação e garantir que apenas os membros com as cotas em dia possam utilizar a plataforma normalmente.

A alta rotatividade de membros e cargos deverá ser considerada, de forma a que os utilizadores consigam alterar o seu cargo e permissões da plataforma de forma rápida e eficaz. Além disso a exportação e importação de dados também é um requisito importante. Desta forma, é possível importar dados na instalação da plataforma de forma a que ela possua, desde o início, uma lista de encarregados de educação e educandos. Isto também permite a possibilidade de uma associação poder manter os dados da plataforma em outro formato (ao exportarem a informação) no caso de decidirem de a deixar utilizar.

Limitações

Ao longo do desenvolvimento deste projeto deparámo-nos com algumas dificuldades que acabaram por limitar e alterar o resultado e os requisitos do projeto. Entre elas está a limitação de custo monetário que tinha sido imposta no início do projeto.

No início do projeto, um requisito proeminente que nos tinha sido imposto estava relacionado com a integração do sistema com o MB WAY relativamente ao pagamento de cotas. De acordo com este requisito, tentámos contactar os responsáveis pelo MB WAY para obtermos informações sobre a integração desse serviço no nosso projeto. Nessa altura não obtivemos resposta, pelo que começámos a investigar outras formas de pagamento que permitissem preencher esse requisito. Decidimos então, como plano alternativo e em conjunto com o orientador do projeto, alterar o método de pagamento de cotas para PayPal. No desenvolvimento do mecanismo de pagamentos por PayPal, reparámos que o plano gratuito dos serviços da Firebase que estávamos a utilizar para suportar a nossa plataforma não permitem a chamada a APIs externas, pelo que também não foi possível implementar o pagamento de

cotas por PayPal. Isto constitui uma limitação aos requisitos e funcionalidades planeados inicialmente na nossa plataforma.

As circunstâncias criadas pela pandemia de Covid-19 também constituem algumas limitações nos processos de trabalho. A falta de reuniões presenciais poderia resultar em algumas falhas de comunicação que afetassem o projeto, embora este problema tivesse sido facilmente colmatado através de contacto frequente via *Slack* e reuniões por videoconferência. A ausência de *pair programming*, especialmente no desenvolvimento do *frontend*, também dificultou a implementação de algumas funcionalidades.

Estrutura do relatório

Além da introdução, este relatório terá mais 6 capítulos.

No segundo capítulo, serão analisadas e comparadas soluções de sistemas já existentes que sejam comparáveis ao nosso projeto. No terceiro capítulo serão descritos os requisitos recolhidos nas fases iniciais do projeto. No quarto capítulo será descrita e justificada a arquitetura escolhida para o nosso sistema. Depois, no capítulo 5, iremos descrever a implementação das várias funcionalidades desenvolvidas no nosso projeto, abordando tanto uma vista mais generalizada das funcionalidades, como uma vista mais funcional das tecnologias utilizadas. No capítulo 6 vamos descrever mecanismos de teste utilizados e, por fim, no capítulo 7 vamos descrever uma visão generalizada do projeto, tal como os seus resultados e outras considerações finais que consideramos relevantes.

No final do relatório, encontram-se ainda apêndices, nos quais estão incluídos o vídeo promocional do projeto, o website de apresentação do produto, bem como de uma demonstração em vídeo.

Estado da arte

Na fase de elaboração de este projeto foi realizada uma pesquisa relativa a outros serviços e plataformas similares à nossa plataforma, de forma a tomarmos conhecimento de desenvolvimentos e abordagens que pudessem ser úteis na realização deste projeto. Não encontramos nenhum serviço com um objetivo equivalente ao que o nosso projeto pretende atingir, pelo que não possuímos nenhum trabalho desenvolvido anteriormente neste tópico.

Tal como foi aludido no documento da proposta do tema deste projeto, a plataforma desenvolvida tem alguns aspetos que são comuns a serviços de CRM (*client relationship management*). Isto levou-nos a observar alguns serviços de CRM para que pudéssemos adaptar alguns dos seus requisitos ao domínio do nosso projeto.

Aspetos em comum com serviços de CRM

Após observarmos alguns serviços de CRM existentes [2] [3] [4], foi possível fazer o levantamento de algumas funcionalidades que seriam favoráveis para o objetivo do nosso projeto:

- Importação e exportação de dados de clientes através de folhas de cálculo;
- Filtrar informação de casos;
- Guardar ficheiros relacionados a casos;
- Marcar eventos (reuniões presenciais).

Estes aspetos são adaptáveis ao domínio do nosso projeto. Enquanto que nas CRMs observadas estes aspetos correspondem ao domínio das vendas e gestão de empresas, estes também são aplicáveis à gestão de uma associação de pais.

Fatores de diferenciação

Também é possível observar diferenças entre os vários serviços de CRM e o nosso projeto. Estas existem devido às particularidades de cada domínio e dos objetivos das plataformas:

- **Gratuito** - Os serviços de CRM são utilizados por empresas, pelo que estas conseguem suportar os custos de utilização do serviço. No entanto, as associações de pais não conseguem suportar os custos de utilização de um serviço tão especializado, pelo que a nossa aplicação será de instalação e utilização gratuitas;
- **UI intuitiva e focada** - Os serviços de CRM são utilizados para propósitos de gestão de vendas, pelo que possuem várias funcionalidades altamente especializadas para esse propósito. No entanto, a gestão de uma associação de pais não requer ferramentas tão especializadas e/ou complexas, pelo

que uma aplicação com funcionalidades mais simples e focadas no domínio de associações de pais é mais adequada para a sua gestão;

- **Hosting local e focado** - Os serviços de CRM têm incluído no seu serviço a hospedagem da aplicação e das bases de dados necessárias. No entanto, a nossa aplicação será hospedada nos serviços gratuitos da Google Firebase, pelo que fica dependente da disponibilidade dos serviços da Google.

Análise de Requisitos

A enumeração dos requisitos do sistema foi feita através de algum conhecimento geral que a própria equipa já tinha sobre o contexto das associações de pais, de forma a listar requisitos favoráveis, bem como em conjunto com o orientador do projeto, ele próprio presidente de uma associação de pais, que levantou vários problemas e limitações correntes que a plataforma deveria abordar.

Requisitos Funcionais

- O Administrador do Sistema, que poderá ou não ser um Encarregado de Educação, pode fazer instalação do sistema através do manual e *scripts* fornecidos;
- Uma associação de pais pode ter um logótipo diferenciado, sendo este alterável pelo no momento da instalação do sistema ou, posteriormente, dentro da aplicação;
- Os órgãos sociais são responsáveis por admitir novos encarregados de educação no sistema;
- Dentro de uma associação existem diferentes membros com cargos diferentes, que são visíveis pelos outros utilizadores, e estes cargos também devem poder ser atribuídos pelos órgãos sociais;
- Pais ou encarregados de educação interessados em aderir deverão preencher um formulário de inscrição que é visto pelos órgãos sociais;
- Após a inscrição de um novo utilizador/encarregado de educação ser aceite, este pode realizar login na plataforma;
- Após a importação dos membros da associação no momento da instalação, estes recebem um email para notificar de que foram importados para a plataforma com um link único para proceder ao login;
- No momento da instalação, a plataforma permite a importação de dados dos membros correntes, caso se trate de uma associação já oficialmente formada;
- No momento da instalação, a plataforma permite a importação de parâmetros adicionais de modo a ter uma configuração personalizada para cada associação e o seu contexto;
- Os vários utilizadores podem criar casos que estes considerem relevantes. Estes casos podem ser públicos ou privados de forma a preservar a privacidade dos utilizadores relativamente às situações abordadas nos casos;
- Em casos privados, é necessário adicionar-lhes utilizadores para que esses passem a ser visíveis apenas a essa seleção restrita de encarregados de educação;

- Utilizadores podem adicionar observações/comentários/notas a casos a que tenham acesso;
- Os utilizadores podem alterar e eliminar observações/comentários/notas da sua autoria;
- Os órgãos sociais podem exportar as informações da aplicação para um formato conhecido, como CSV ou PDF;
- Os utilizadores podem carregar documentos/anexos a casos a que tenham acesso;
- O sistema tem um mecanismo de registo de cotas com confirmação tanto do emissor como do recetor, servindo estes registos de prova de pagamento.

Requisitos Não Funcionais

Requisitos de Usabilidade

- A plataforma deve ter uma interface apelativa e organizada com fontes e cores que facilitem a legibilidade da informação;
- A plataforma contar com um mecanismo de pesquisa que permita aos utilizadores encontrar recursos na plataforma (outros utilizadores, casos, documentos da associação, etc.) de forma fácil e rápida;
- Interface da plataforma deve ser intuitiva e de fácil utilização;
- A plataforma deve ter um mecanismo de pagamento fácil e rápido de cotas.

Requisitos de Desempenho

- A aplicação deve estar disponível a maior parte do tempo (pelo menos, 90% das vezes).

Requisitos de Segurança e Integridade de Dados

- Um utilizador não pode ter acesso a dados confidenciais e privados de outros utilizadores registados na plataforma;
- Os casos mais sensíveis que envolvam alunos têm de ser apenas e só visíveis e discutidos por utilizadores com acesso a esses casos;
- A plataforma não pode ter acesso aos dados bancários do cliente, disponibilizando formas de pagamento externas e seguras como, por exemplo, MB WAY ou PayPal;
- Relativamente a um novo utilizador, deve ser garantida a sua privacidade e ficar claro o fim a que se destinam os dados necessários para o seu registo como, por exemplo, email, contacto telefónico, NIF ou data de nascimento.

Requisitos de Interface com Sistemas Externos

- Informação deve ser armazenada numa base de dados segura, funcional e estável;
- A plataforma deve ser hospedada num ambiente estável e gratuito;
- Deve ser possível uma instalação configurável que permita ao administrador fornecer parâmetros adicionais para a associação e instalar a plataforma de acordo com as necessidades da mesma.

Atores

O alvo principal da plataforma são as associações de pais que procuram uma melhor solução para a gestão de toda a informação relacionada com a organização. Estas são, normalmente, constituídas por órgãos sociais e outros encarregados de educação.

- **Órgão social** - Encarregado de educação responsável pela associação de pais que é encarregue de gerir e comunicar com outros encarregados de educação e com a escola de forma a resolver questões que surjam e que envolvam os pais, os alunos e a escola.
- **Encarregado de Educação** - Encarregado de educação responsável por um ou mais estudantes que frequentam um dado estabelecimento de ensino e que quer ter a certeza de que não surgem problemas que possam prejudicar os seus educandos tomando um papel ativo na vida escolar dos mesmos.

Para além dos dois tipos de atores numa associação de pais, é necessário o papel de um Administrador do Sistema para inicializar a plataforma e gerir a mesma.

- **Administrador do Sistema** - Encarregado de educação que encontrou e sugeriu a utilização da aplicação para ajudar a gerir os assuntos da associação de pais e que, por já ser familiar com a utilização de várias tecnologias, se tornou responsável pela sua instalação e monitorização/gestão.

Casos de Utilização

A seguinte figura representa o modelo de casos de utilização dos vários atores na plataforma. Mais abaixo é detalhado cada caso de utilização individualmente.

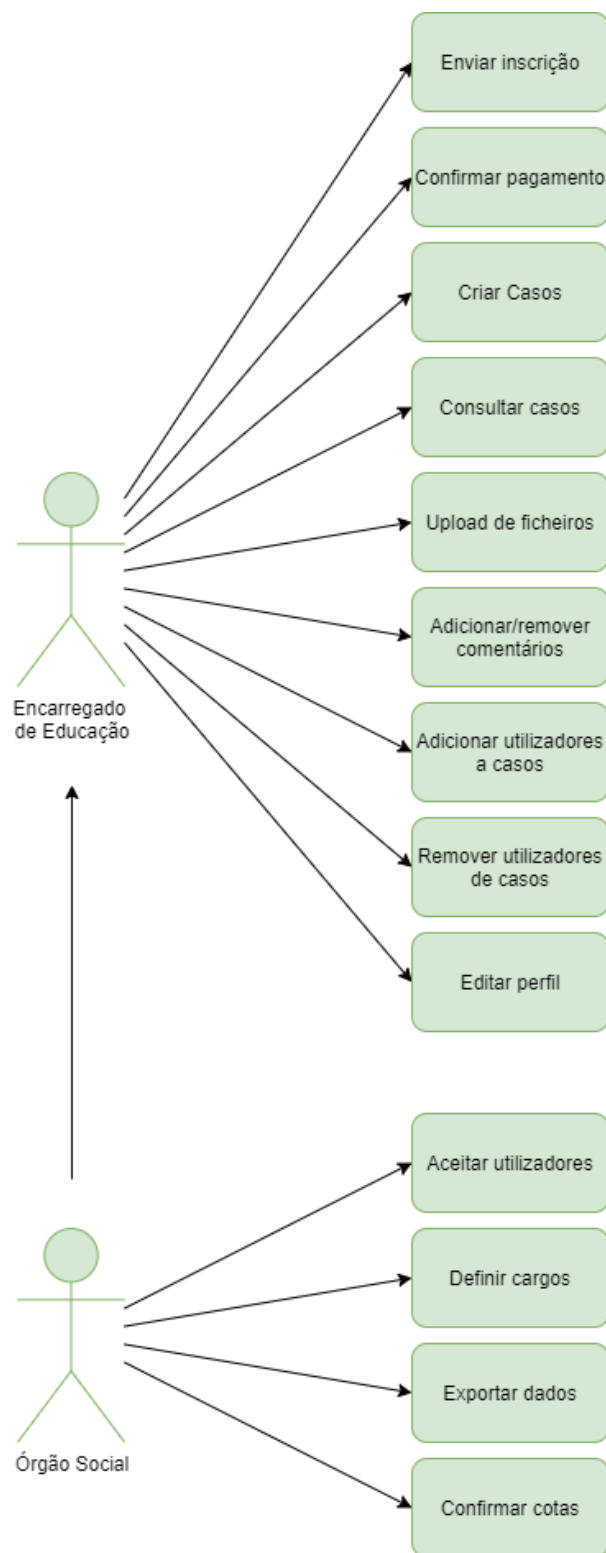


Figura 1 - Casos de utilização de utilizadores

- **Enviar inscrição** - O utilizador envia a inscrição para a associação para passar a ser um possível membro da mesma e usufruir da plataforma;
- **Confirmar pagamento** - O utilizador confirma o pagamento da cota do ano letivo presente;
- **Criar casos** - O utilizador cria um caso para tratar de qualquer problema que surja podendo editá-lo a qualquer momento;
- **Consultar casos** - O utilizador consulta todas as informações (incluindo comentários/observações) de um caso ao qual tenha acesso;
- **Upload de ficheiros** - Fazer *upload* de ficheiros relevantes a casos;
- **Adicionar/remover comentários** - O utilizador pertencente a um caso adiciona um comentário/observação e pode também eliminar comentários da sua autoria;
- **Adicionar utilizadores a casos** - O utilizador autor de um caso/órgão social adiciona outro utilizador a esse mesmo caso (para que ele passe a ter acesso);
- **Eliminar utilizadores a casos** - O utilizador criador de um caso/órgão social remove outro utilizador desse mesmo caso (para que ele deixe de ter acesso);
- **Editar perfil** - O utilizador edita as informações do seu perfil pessoal;
- **Aceitar utilizadores** - Os órgãos sociais da associação de pais aceitam inscrições de utilizadores para entrar na associação de pais;
- **Definir cargos** - O Administrador do Sistema ou qualquer órgão social da associação define cargos a atribuir aos utilizadores. Estes cargos também podem ser transferidos com um mecanismo similar a uma tomada de posse;
- **Exportar dados** - Os órgãos sociais da associação exportam dados relativos a casos ou a utilizadores para um formato externo;
- **Confirmar cotas** - Os órgãos sociais da associação registam pagamentos realizados por outros encarregados de educação;

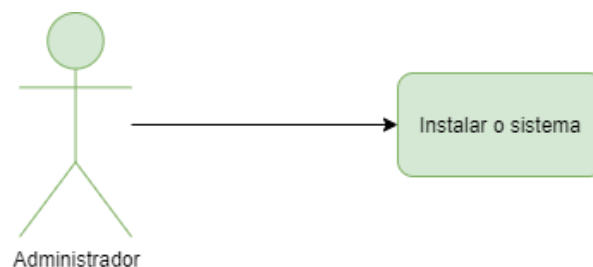


Figura 2 - Caso de utilização do Administrador do Sistema

- **Instalar o sistema** - O Administrador do Sistema instala o sistema seguindo o manual de instruções disponibilizado.

Arquitetura

Neste capítulo vamos apresentar os modelos do domínio e das tecnologias utilizadas para desenvolver o projeto, bem como descrever e justificar as escolhas feitas em ambas as situações. Desta forma, pretendemos aprofundar os aspetos relacionados com a estrutura do sistema desenvolvido.

Modelo do domínio

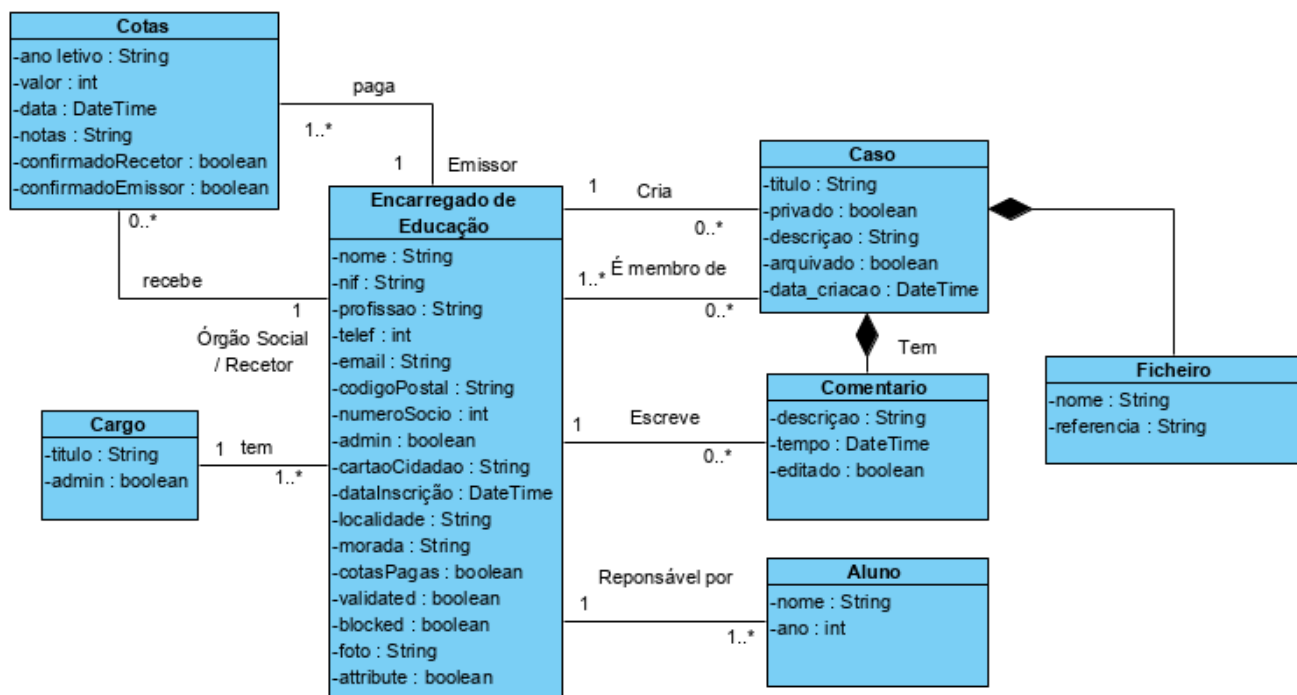


Figura 3 - Modelo do domínio

Na figura acima, está apresentado o diagrama do domínio principalmente utilizado para a modelação da base de dados utilizada no projeto. Como é possível observar, esta contém vários elementos relacionados com o domínio académico.

A entidade central que corresponde aos utilizadores é a entidade Encarregado de Educação. Esta é referenciada em quase todas as outras entidades. Um encarregado de educação possui um ou vários educandos/alunos pelo qual é responsável, daí a relação entre eles. Além disso, os encarregados de educação podem criar casos, pelo que ficam relacionados a eles como autores. Os casos, no caso de serem privados, possuem uma lista de membros que os podem visualizar. Sendo assim, os casos também estão relacionados aos encarregados de educação através dessa lista de membros. Os

comentários realizados nos casos também possuem um encarregado de educação registado como seu autor. Um caso pode ter vários ficheiros associados a si, relativos a documentos externos.

Os utilizadores podem ser apenas membros normais ou possuir um cargo que lhes dá permissões especiais, pelo que a entidade Encarregado de Educação está relacionada com a entidade Cargo, de forma a estabelecer essa relação semântica.

Para o registo de cotas é necessário que o pagamento seja confirmado tanto por um órgão social/recetor como pelo membro que realiza o pagamento, ou seja, o emissor. Isto traduz-se nas duas relações existentes entre as entidades Cotas e Encarregado de Educação.

Modelo tecnológico

Nesta secção vamos analisar o diagrama tecnológico do nosso projeto tal como comparar duas versões do mesmo. Tendo sido a primeira versão a realizada na *milestone 2*/elaboração deste projeto e a segunda versão realizada no final do projeto com alterações relativas a mudanças de requisitos e tecnologias utilizadas.

Versão inicial

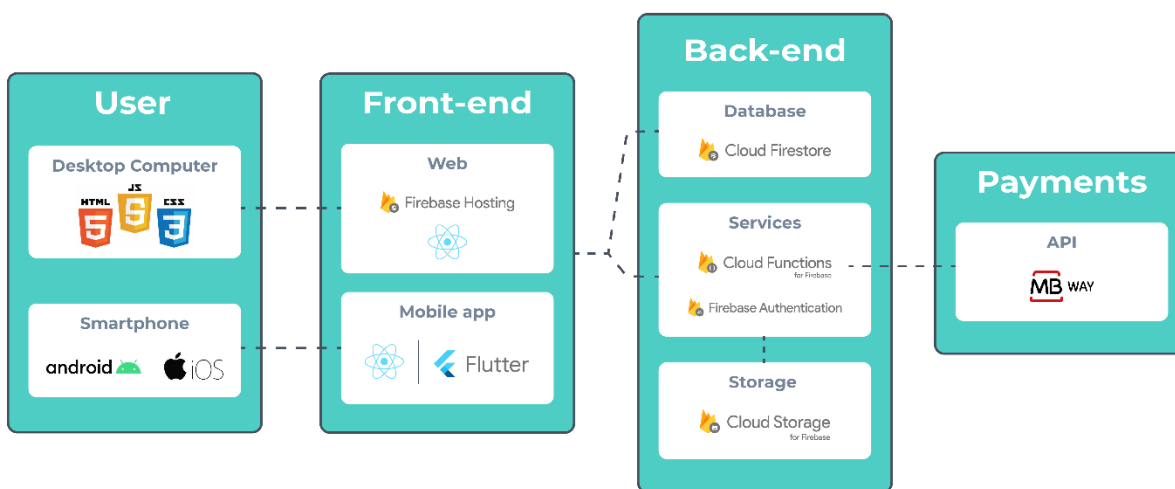


Figura 4 - Modelo tecnológico (versão inicial)

Esta versão inicial do diagrama conta com a utilização de HTML, CSS e JavaScript (e React) para a realização da aplicação web e de Flutter ou React Native para a realização de uma aplicação móvel. O *backend* utilizaria os vários serviços/módulos fornecidos pela Google Firebase (*backend-as-a-service*), sendo estes:

- Cloud Firestore, utilizado como base de dados;
- Firebase Hosting, utilizado para o *hosting* da aplicação;
- Cloud Functions, utilizado para realizar operações mais ou menos complexas que podem envolver outros módulos (como a Firestore e Authentication);
- Firebase Authentication, utilizado como serviço de autenticação;
- Cloud Storage, utilizado como serviço de armazenamento para ficheiros e fotografias.

Por fim, decidimos utilizar MB WAY como serviço de pagamentos de cotas, tal como estava especificado nos requisitos do projeto.

Versão final

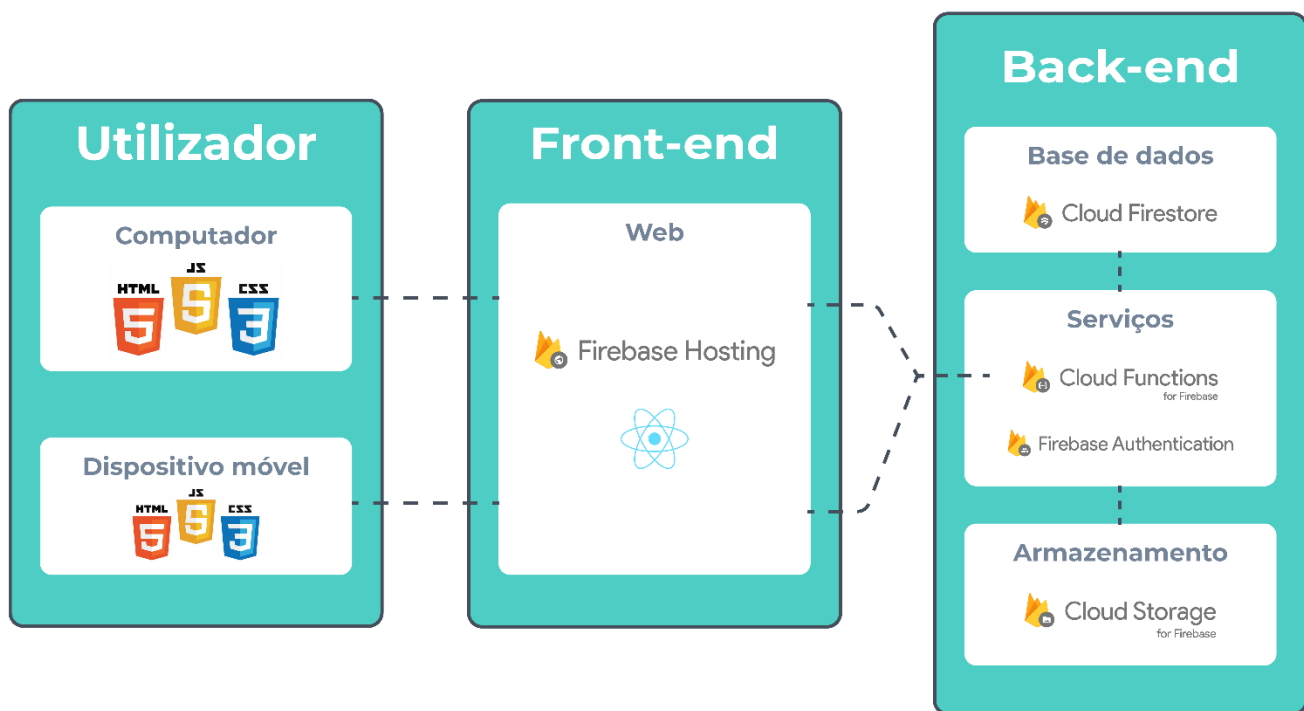


Figura 5 - Modelo tecnológico (versão final)

Durante o projeto foram realizadas mudanças relativamente aos seus requisitos, pelo que foi necessária a realização de algumas alterações. As principais mudanças realizadas comparativamente à primeira versão do diagrama foram as seguintes:

- A eliminação do módulo de pagamentos com MB WAY, que ocorreu devido a uma mudança de requisitos do projeto durante o desenvolvimento. As razões desta mudança foram especificadas na secção relativa às limitações do projeto no capítulo de Introdução deste relatório;
- A mudança em que foi decidido que não se iria realizar uma aplicação móvel independente, mas se iria adaptar a aplicação web para dispositivos móveis. Esta alteração foi realizada principalmente devido a restrições de tempo. Ao realizarmos esta mudança, foi tido em conta que o requisito de utilização móvel continuaria a ser satisfeito mesmo após a alteração ter sido realizada, garantindo que a aplicação web é responsiva.

Implementação

Para a implementação, a nossa abordagem começou por discutir em equipa tecnologias que nos permitissem ir ao encontro dos requisitos do projeto e que permitissem também desenvolver as nossas capacidades. Neste capítulo vai apresentado o processo de desenvolvimento do nosso projeto. Esta secção será dividida em várias subsecções que tratarão, cada uma, de forma mais detalhada, um mecanismo ou módulo utilizado na conceção do projeto ou de detalhes de implementação de uma funcionalidade.

Componentes da Implementação

Google Firebase

Uma vez que um dos requisitos era que a plataforma fosse gratuita, tínhamos de adotar soluções que nos permitissem hospedar, armazenar dados, desenvolver e fazer o *deploy* da nossa aplicação sem quaisquer custos monetários. Foi tendo este aspeto em consideração, e também o facto de ter uma grande comunidade, suporte e documentação, bem como as sugestões do nosso orientador, que optámos por enveredar pelos serviços fornecidos pela Google Firebase [5] que, em termos gerais, se apresenta como uma plataforma *backend-as-a-service* que nos permitiu desenvolver vários módulos necessários para o projeto, desde uma API (Cloud Functions), uma base de dados (Cloud Firestore), hospedar o projeto para que pudesse ser disponibilizado ao público (Firebase Hosting) e outras operações, através dos seus microsserviços. De notar que a própria Firebase tem diferentes planos com diferentes preços, pelo que desenvolvemos segundo o plano gratuito do serviço, que satisfaz as nossas necessidades, apesar das suas limitações. Nesta secção, vamos enumerar e especificar o uso de cada um dos serviços utilizados, de forma a clarificar o papel de cada um deles no funcionamento da plataforma.

Hosting

Firebase Hosting [6] é o serviço da Firebase que permite a hospedagem da plataforma propriamente dita, isto é, dos ficheiros HTML, CSS e Javascript (mas não limitado a, podendo hospedar quer conteúdos estáticos, quer dinâmicos), para que possa ser acedida pelos utilizadores através de um URL. O facto de ser gratuito e pertencer ao ecossistema da Firebase foram as duas principais razões

que nos levaram a hospedar a nossa plataforma neste serviço, mas este tem outras vantagens que conferem alguma robustez, desempenho e facilidade de utilização:

- Em termos de segurança, o serviço corre os protocolos HTTPS e SSL, permitindo-nos fornecer conteúdo pela rede de forma segura e sem configuração adicional;
- Apesar de o Hosting gerar domínios da Firebase para o projeto, também é possível adicionar os nossos próprios domínios que podem ser fácil e rapidamente validados através de certificados SSL;
- Fazer o *deployment* (instalação) da aplicação é muito simples: basta correr um comando na linha de comandos e a nossa aplicação está pronta a usar através de um navegador web;
- O serviço mantém um histórico das versões da aplicação que foram sendo *deployed* ao longo do tempo, permitindo voltar a versões anteriores se necessário;
- É possível fazer a hospedagem de múltiplos sites que, embora partilhem os recursos da Firebase, podem ser independentes entre eles (com os seus próprios ficheiros, recursos, etc.);
- A ausência de necessidade de manter um servidor próprio, pelo que não temos de ser responsáveis pela sua manutenção, desempenho e disponibilidade.

Cloud Functions

Neste projeto foram utilizadas as Cloud Functions disponibilizadas pela Google Firebase para realizar o acesso da aplicação à base de dados. Isto inclui as várias operações CRUD disponíveis às entidades presentes na base de dados, sendo que estas *cloud functions* funcionam de forma semelhante a uma API que gere o fluxo de dados entre o *frontend* e o *backend*. Além deste papel, as *cloud functions* também foram utilizadas para enviar emails de notificação e de autenticação aos utilizadores e para executar exportações de dados através de ficheiros, quer em formato CSV, quer em formato PDF.

Cloud Functions representam um serviço FAAS (*function as a service*) [7]. Este tipo de serviços consiste em execuções de segmentos de código de *backend* em forma de funções, sem que seja necessária a gestão de um servidor próprio para o efeito ou de uma aplicação continuamente ativa num servidor remoto. Isto permite que seja apenas necessário o *upload* do código relativo às funções para realizar o seu *deployment*. Geralmente, estes serviços permitem que a execução das funções seja desencadeada através de vários tipos de *triggers*. Alguns destes são operações na base de dados e operações de autenticação. No nosso projeto, tendo em conta que, como já referimos, as funções funcionam de forma similar a uma API, as funções criadas são iniciadas através de pedidos HTTPS.

Como já aludimos, a implementação do nosso *backend* através de *cloud functions* proporciona várias vantagens relativamente a outras implementações [8]. De entre estas, estão:

- O custo de utilização deste serviço é reduzido relativamente a outras opções de *backend*;
- A não necessidade de manter um servidor físico, pelo que não somos responsáveis pela sua manutenção;
- Embora não represente um problema grave no domínio do nosso projeto, a utilização de *cloud functions* faz com que não nos tenhamos de preocupar com problemas de escalabilidade, pelo que o serviço fica responsável por realizar essas operações, sendo a escalabilidade automática;
- O *deployment* do *backend* através de *cloud functions* é mais fácil e rápido do que a realização de um *deployment* de um servidor completo.

Apesar desta escolha de arquitetura possuir várias vantagens, também implica várias desvantagens e limitações [8], que incluem:

- A existência de inicializações a frio (*cold starts*) diminui, por vezes consideravelmente, a performance das operações de *backend*, quando estas são utilizadas de forma muito dispersa ao longo do tempo. Isto significa que, se a aplicação não for utilizada de forma regular ao longo do tempo as primeiras operações realizadas terão um tempo de inicialização mais elevado;
- As funções possuem um tempo limitado de execução, por invocação, pelo que não devem possuir operações muito pesadas computacionalmente;
- Devido à utilização de um ou múltiplos servidores físicos desconhecidos não existe a possibilidade de guardar um estado em memória persistente entre várias invocações de funções;
- Não existe a possibilidade para realizar otimizações de performance através de mudanças de design do servidor;
- Realizar testes, especialmente de integração, será mais complicado pelo facto de as funções serem executadas em um ou vários servidores desconhecidos. Isto faz com que seja impossível simular o ambiente de *cloud* em que o código será executado.

Cloud Firestore

A Cloud Firestore [9] é uma base de dados NoSQL orientada a documentos e é a base de dados da nossa plataforma. É nesta base de dados que estão guardadas todas as informações e dados da plataforma, o que inclui dados de utilizadores e educandos, informações sobre casos, cargos e transições de mandatos, registo de cotas e outros, de forma persistente. É importante notar que,

quando há informações apagadas pelos utilizadores através da interface (nomeadamente de entidades como casos, educandos, registos de cotas ou dos próprios utilizadores e das suas contas), a informação, embora não visível por estes, continua persistente na base de dados para efeitos de arquivo devido a, embora não limitado, obrigações legais.

A Cloud Firestore está integrada no ecossistema da Firebase, sendo a maior razão pela opção por este serviço, mas oferece outras vantagens que conferem ao nosso sistema mais capacidades:

- O facto de ser NoSQL e orientada a documentos confere maior desempenho e permitiu-nos desenvolver com maior flexibilidade, relativamente a modelos relacionais, no que toca ao armazenamento de dados, sendo possível, por exemplo, estruturar os dados de forma hierárquica com documentos organizados em coleções, sendo que os documentos podem conter em si outras coleções com outros documentos. Além disso, a possibilidade de armazenar listas dentro de documentos foi uma grande vantagem na modelação e estruturação do nosso modelo de dados, algo que não é possível em bases de dados relacionais (algo que teria de ser contornado com referências a outras tabelas com chaves estrangeiras, por exemplo);
- A capacidade da Firestore suportar operações *offline*, através de mecanismos de *caching*, permitindo que as aplicações continuem a responder independentemente de eventual latência na rede, por exemplo; fornecendo ainda mecanismos de sincronização entre dispositivos assim que voltem a estar *online*;
- Como já referido, o facto de estar integrado com outros serviços da Firebase, nomeadamente as Cloud Functions, permite que o nosso cliente (*frontend*) não tenha de aceder diretamente à base de dados nem operações complexas cada vez que é necessário realizar acessos à mesma, já que esses acessos e operações são tornadas abstratas pelas Cloud Functions, como referido acima;
- À semelhança de outros serviços, também com a Cloud Firestore a escalabilidade é automática, embora dentro dos limites do plano gratuito, permitindo replicação de dados automática, consistência, operações atómicas e transações.

Contudo, a Cloud Firestore também algumas desvantagens [10]. Apesar das suas opções para fazer *queries* recorrendo a filtragem e ordenação, há ainda algumas operações que não são possíveis, nomeadamente pesquisar por texto ou por subtexto nos valores dos documentos (para além de ser *case sensitive*), tendo as *queries* que ser feitas pelo texto exatamente que procuramos. Além disso, algumas opções de *querying* não podem ser combinadas. Alternativamente, terá de ser feito processamento adicional do lado do cliente, o que pode levar a quebras de desempenho. Mesmo assim, no desenvolver deste projeto, não tivemos grandes constrangimentos no que toca a este aspeto, mas reconhecemos que pode vir a ser uma desvantagem caso queiramos melhorar o produto no futuro. Por último, embora

seja vantajoso que os dados estejam armazenados num servidor remoto que não é da nossa responsabilidade, já que não temos de nos preocupar com a sua gestão, poupando-nos tempo, recursos e custos, também pode ter um outro lado em que os dados não estão sob o nosso controlo direto e, se algo catastrófico acontecer aos servidores da Google, estaremos dependentes da sua capacidade para reagir e recuperar desse tipo de situações.

Cloud Storage

Outro microsserviço da Firebase que adotámos foi a Cloud Storage [11]. Este módulo consiste num serviço robusto, simples e acessível que permite o armazenamento e acesso a partir da nuvem de objetos, principalmente ficheiros, como imagens, vídeos, áudio e outros.

A necessidade de utilização deste serviço passou principalmente pelo armazenamento de imagens. A nossa plataforma permite que os utilizadores carreguem fotos para o seu perfil e para os seus educandos, e que os administradores / órgãos sociais da associação forneçam também um logótipo para a mesma. Quando um utilizador carrega uma foto ou um novo logótipo para a associação estes ficheiros são carregados para a nuvem e posteriormente acessíveis através de um URL público para que possa ser mostrado na interface aos utilizadores.

Para além de estar integrado no ecossistema da Firebase, a opção pela Cloud Storage também foi tomada tendo em conta as vantagens que fornece:

- O custo de utilização deste serviço é gratuito dentro de certos limites, limites esses nos quais as nossas necessidades, no âmbito do projeto, são satisfeitas;
- A não necessidade de manter um servidor físico, pelo que não temos de ser responsáveis pela sua manutenção;
- A gestão de uploads e downloads por parte do serviço é dinâmica, flexível e robusta, na medida em que o serviço pode, por exemplo, retomar downloads que tenham sido suspensos por, por exemplo, problemas de rede, o que evita que os nossos utilizadores tenham menos preocupações com a largura de banda e o tempo da operação;
- Embora o domínio do nosso projeto represente as atividades de uma associação de pais, que, em teoria, possuem poucas necessidades de escalabilidade devido ao relativamente reduzido volume de dados que manipulam, a utilização da Cloud Storage permite que não nos tenhamos de preocupar com problemas de escalabilidade, pelo que o serviço fica responsável por realizar esse ajuste, sendo a escalabilidade automática, embora dentro dos limites impostos pelo plano gratuito da Firebase.

Authentication

O Firebase Authentication [12] foi utilizado para facilitar ao máximo o processo de autenticação dos utilizadores. Este fornece serviços de *backend* fáceis de usar e bibliotecas de UI prontas para autenticar utilizadores na plataforma. Além disso, também é garantida a segurança destes processos de autenticação, já que este serviço conta com mecanismos para identificar e monitorizar que utilizadores entram e saem (*logout*) da plataforma, e funcionalidades como limitar o acesso a determinados domínios da aplicação consoante o tipo de utilizador e o seu estado de autenticação, bem como a possibilidade de restringir a criação de contas diferentes com o mesmo email mesmo quando usado em provedores de identidade (mencionados de seguida) diferentes.

Relativamente aos métodos de autenticação, o Firebase Authentication permite ainda a autenticação por diversas vias como por senha, telefone ou até mesmo através de outros provedores de identidade como Google ou Facebook. Esta integração é crucial para manter a simplicidade da autenticação do lado dos utilizadores, já que podem usar provedores externos com estes e que já utilizam para se autenticarem de forma mais rápida e simples.

Através destes serviços, adotámos três tipos de autenticação, todas elas baseadas num formato sem *password*, ou seja, nunca é necessário criar ou escrever uma palavra-passe no Registo ou Login da aplicação. Os três tipos de autenticação adotados foram os seguintes.

- **Link por email** - Desta forma, é apenas necessário escrever o email do utilizador (o que usou para se registar) para o qual é enviado um link gerado automaticamente. Ao clicar nesse link, o utilizador é redirecionado para a página da aplicação com o login efetuado.
- **Google** - É também possível ligar a conta atual do utilizador a uma conta do Google e, a partir daí, passa a ser possível fazer login através da autenticação do Google.
- **Facebook** - O mesmo é possível com o Facebook, podendo ser feito login com a autenticação da rede social.

Apesar da facilidade de utilização deste módulo e das suas funcionalidades, no caso particular da integração com a autenticação do Facebook, foram necessários passos adicionais [13] comparativamente à integração com a Google, já que a rede social tem a sua própria plataforma de desenvolvedores, na qual foi necessário registar a nossa plataforma usando as chaves do nosso projeto no Firebase. Estes passos levaram a que tivéssemos investido algum tempo adicional a investigar os

requisitos do próprio Facebook para a integração com a sua API de autenticação no nosso projeto. O mesmo é válido para outros provedores de identidade, como a Microsoft, cuja integração com esta não implementámos devido a limitações e pré-requisitos da empresa que envolvem alguns custos.

Aplicação Web

React

Para o desenvolvimento da aplicação web, nomeadamente do *frontend* do sistema, a equipa decidiu usar React [14], uma biblioteca JavaScript *open-source* e desenvolvida pela Facebook, focada no desenvolvimento de interfaces de utilizador, com uma estrutura baseada em componentes. Um componente é uma das unidades básicas do React, similar ao conceito de *Object* em Java. Um Componente pode ser definido como classe ou função, receber parâmetros através da *keyword props* e manter um estado interno (*state*). Aquando de atualizações a variáveis do estado usadas diretamente na interface visual do utilizador, a atualização da mesma é feita de forma automática (daí o nome de React), sendo possível, portanto, atualizar apenas os componentes alterados na interface sem ter de recarregar toda a página completamente. Esta tecnologia, por estar orientada a componentes e permitir definição de novos pelos *developers*, tem a vantagem da possível reutilização de componentes ao longo dos projetos.

A escolha do React para este projeto passou principalmente pelo conjunto de características anteriormente mencionadas, que, aliás, atribuem a esta biblioteca uma boa fama para o uso de desenvolvimento de interfaces de utilizador, e pelo contacto prévio de parte da equipa de *frontend* com a tecnologia.

Gravatar

O Gravatar é um serviço onde um utilizador pode associar uma ou mais imagens/avatars ao seu email [15]. Isto com o objetivo de o utilizador poder utilizar uma imagem/avatar global para todas as suas contas em vários serviços, pressupondo que esses serviços estão integrados com o Gravatar.

Após os utilizadores se associarem ao serviço de Gravatar com o seu email, a sua imagem pode ser obtida colocando apenas um link de acesso como atributo “*source*” de uma imagem.

Este serviço foi sugerido pelo orientador do projeto e escolhido devido à sua simplicidade de uso e flexibilidade, que existe devido aos argumentos que, quando passados no URL da imagem, podem causar o aparecimento de diferentes imagens. Estes diferentes comportamentos incluem um pedido com uma imagem por omissão caso o utilizador não tivesse imagem definida no Gravatar e um pedido que devolve apenas imagens com um certo *rating* de maturidade (um exemplo de *rating* pode ser *g* que significa uma imagem adequada para exibição em qualquer site com qualquer audiência).

No nosso projeto em particular, quando um utilizador se regista, é calculada uma *hash* MD5 com base no seu email e criado um URL para o seu Gravatar com esta *hash* [16]. Se o utilizador já tiver um Gravatar para esse email, esse URL vai corresponder ao seu Gravatar, que vai ser mostrado no seu perfil. Caso contrário, o avatar do utilizador irá corresponder ao avatar a um avatar por omissão. Também nos educandos dos utilizadores foi utilizado Gravatar. Contudo, como os educandos não têm email, é-lhes atribuído inicialmente a imagem por omissão geral do Gravatar (que não precisa de nenhuma *hash* com base num email), disponibilizada através do URL <https://www.gravatar.com/avatar/00000000000000000000000000000000>.

Funcionalidades

Configuração da associação



Figura 6 - Página inicial de configuração

Após o *deployment* da aplicação, é apresentada uma página em que o utilizador terá de escolher entre configurar uma associação completamente nova ou uma associação já existente. A primeira opção é para alguém que queira fundar uma associação fisicamente e que queira que os membros adotem esta plataforma logo desde a sua formação. A segunda opção é dirigida aos órgãos sociais de uma associação já formada e que esteja já a atuar num estabelecimento de ensino e que desejem transitar a sua forma de trabalho para a nossa plataforma.

Em qualquer uma das opções, o utilizador será redirecionado para um formulário onde será incentivado a preencher as informações da associação, como o nome, morada, email de contacto, o valor da cota ou ainda, opcionalmente, carregar um logótipo. A grande diferença entre os formulários de ambas as opções reside na forma como recolhemos os primeiros dados de membros da associação.

Configuração

Nome da Associação de Pais
Associação de Pais e Encarregados de Educação da Escola...

Descrição (opcional)
Descrição breve da associação e/ou alguns objetivos.

Morada (sede/escola)
Rua Dr. Ricardo Silva, 276 R/C Dto

Localidade
Aveiro

Código Postal
1234-567

Email da Associação
associacao@exemplo.pt

Telemóvel / Telefone (opcional)
200345678

IBAN (opcional)
PT50 1234 4321 12345678901 72

Valor da Quota (€)
5

Logótipo da associação/escola (opcional) ⓘ
Escolha uma imagem... Browse

Membros da Associação ⓘ
Escolha um ficheiro... Browse

Alunos e Encarregados de Educação ⓘ
Escolha um ficheiro... Browse

Parâmetros particulares do contexto da Associação ⓘ
Escolha um ficheiro... Browse

Cargos da Associação ⓘ
Escolha um ficheiro... Browse

☐ Li e aceito a [Política de Privacidade](#).

Instalar

Figura 7 - Formulário de configuração de uma associação pré-existente

No caso de ser uma associação já existente, é solicitado que o utilizador carregue ficheiros em formato CSV com os dados dos seus membros e os seus educandos, como vamos detalhar mais adiante.

Configuração

Sobre si ⓘ

O seu nome: João Oliveira

O seu email: joao@associacao.pt

Cargo: Cargo a desempenhar na associação

A Associação

Nome da Associação de Pais: Associação de Pais e Encarregados de Educação da Escola...

Descrição (opcional): Descrição breve da associação e/ou alguns objetivos.

Morada (sede/escola): Rua Dr. Ricardo Silva, 276 R/C Dto

Localidade: Aveiro

Código Postal: 1234-567

Email da Associação: associacao@exemplo.pt

Telemóvel / Telefone (opcional): 200345678

IBAN (opcional): PT50 1234 4321 12345678901 72

Valor da Quota (€): 5

Logótipo da associação/escola (opcional) ⓘ: Escolha uma imagem... Browse

Parâmetros particulares do contexto da Associação ⓘ: Escolha um ficheiro... Browse

Cargos da Associação ⓘ: Escolha um ficheiro... Browse

☐ Li e aceito a [Política de Privacidade](#).

Instalar

Figura 8 - Formulário de configuração de uma nova associação

Já no formulário de uma nova associação, como ainda não terá membros, apenas é pedido ao utilizador que o está a preencher que forneça o seu nome, email e cargo que irá desempenhar na associação. Estes são necessários para que seja possível criar uma primeira conta na plataforma (uma conta para quem está a preencher o formulário) para que possa começar a explorá-la e aceitar novos utilizadores (já que novos registos terão de ser aprovados por utilizadores com permissão de administrador, como é o caso da conta do utilizador que preencher o formulário; mais adiante, iremos detalhar o mecanismo de registo e respetiva aprovação).

Configuração dinâmica e importação de dados

Uma das grandes premissas da nossa plataforma é o facto de ser configurável ao começo, permitindo a sua adaptação a qualquer tipo de estabelecimento de ensino, seja esta uma escola do ensino obrigatório, um clube de futebol ou um conservatório artístico. Para isto, tanto no formulário de configuração de uma nova associação como no de uma associação existente, está disponível uma área em que é possível carregar um ficheiro de formato JSON com informações acerca dos seus

membros e educandos e que sejam particulares do contexto em que a associação se insere, não estando, por isso, previstos nos formulários. O ficheiro deverá ter a seguinte estrutura:

```
{
  "EE" :
    {
      "um parametro" : "texto"
    },
  "aluno" :
    {
      "Grau" : "inteiro", "Regime" : "texto", "Modalidade" : "texto",
      "Pólo" : "texto", "Média" : "decimal"
    }
}
```

Figura 9 - Exemplo de ficheiro JSON de parâmetros adicionais

Como podemos ver acima, o ficheiro JSON fornecido terá de conter ele próprio um JSON com parâmetros para os membros da associação (denotados por “EE”) e outro JSON para os seus educandos (denotados por “aluno”). Dentro de cada um destes JSON, cada chave deverá ser o nome do parâmetro e o seu valor o seu tipo. Por exemplo, num clube desportivo, a associação de pais pode pretender guardar a modalidade dos educandos e essa informação poderá ser textual (o nome da modalidade). No momento da escrita deste relatório, a nossa plataforma permite três tipos distintos: “texto”, “inteiro”, “decimal”. De notar ainda que, apesar de no formulário, a submissão deste ficheiro ser obrigatória, se a administração da associação não pretender fornecer novos parâmetros, pode optar por submeter o ficheiro com o JSON vazio ({ }). Da mesma forma, também é possível este ficheiro ser submetido apenas com um JSON para os membros ou um só para os educandos, consoante as necessidades da associação, não sendo obrigatório fornecer parâmetros para ambos num mesmo ficheiro. Qualquer outra designação para membros e educandos além de “EE” e “aluno”, respetivamente, ou tipos que não sejam um dos mencionados acima, não serão aceites e será mostrado um aviso a indicar que o ficheiro de parâmetros submetido é inválido, não sendo possível prosseguir com o processo de configuração enquanto o ficheiro não for retificado.

Outro dado que solicitamos que sejam importados nos formulários de configuração é relativo aos cargos. Esta importação, tal como na importação de parâmetros, é feita fornecendo um ficheiro JSON com a seguinte forma:

```

{
  "Presidente do Conselho Executivo" :
    {
      "admin" : "true"
    },
  "Vice-Presidente" :
    {
      "admin" : "true"
    },
  "Tesoureiro(a)" :
    {
      "admin" : "true"
    },
  "Vogal" :
    {
      "admin" : "true"
    },
  "Associado(a)" :
    {
      "admin" : "false"
    }
}

```

Figura 10 - Exemplo de ficheiro JSON de cargos

Os cargos de uma associação definem uma divisão hierárquica e na plataforma essa divisão estabelece quem tem permissões de administrador (dirigidas essencialmente aos membros dos órgãos sociais da associação) e quem não tem (nomeadamente, os sócios “normais”) com base nos cargos dos seus utilizadores. Assim, o ficheiro JSON deverá incluir os nomes dos cargos como chave em que a cada um irá corresponder outro JSON com a chave “admin” o valor “true” ou “false”, consoante o cargo seja dirigido a utilizadores com permissões de administrador ou não, respetivamente. Claro que, embora tenhamos mencionado que uma possibilidade seria a atribuição de permissões de administrador a utilizadores de órgãos sociais, esta configuração permite que o utilizador faça a sua própria atribuição, bastando inserir “true” apenas nos cargos que desejar conceder essas permissões. Uma vez realizada esta configuração, os utilizadores com um determinado cargo terão as permissões associadas a esse cargo. De notar que o carregamento deste ficheiro é obrigatório e não será permitido que o JSON (quer o JSON “global” ou o JSON associado a cada cargo) seja vazio. Além disso, qualquer outra designação para além de “admin” para o nome da permissão, ou “true” ou “false” para o seu valor, não será aceite, situação na qual será mostrado um aviso a indicar que o ficheiro submetido é inválido, não sendo possível prosseguir com o processo de configuração enquanto o ficheiro não for retificado. Quanto à designação dos cargos, esta é, naturalmente, da responsabilidade do utilizador, tendo a

obrigação de indicar o nome dos cargos da sua associação de forma correta, bem como fazer corresponder corretamente cada cargo à sua permissão.

Por último, é solicitado ainda ao utilizador que está a realizar esta configuração inicial que forneça dois ficheiros em formato CSV: um com os dados dos membros e outro com os dados dos seus educandos. As áreas para carregar estes ficheiros estão apenas disponíveis no formulário de configuração de uma associação já formalmente formada, já que é nessa situação que existem já membros com educandos no estabelecimento de ensino onde atua. O formato dos ficheiros é o seguinte:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	Número d	Cargo	Quotas pagas	Nome	NIF	Cartão Cid	Email	Telemóve	Profissão	Morada	Código Post	Localidad	Admin	um parametro	
2	54353	Presidente	true	Diogo Gomes	894274893	CC1	dgomes@pi-	987585	Professor	Rua do Col	0000-000	Aveiro	true	v1	
3	98978	Vice-Presi	false	Vera Teixeira	895758866	CC2	vera@pi-ass	6895655	Investigadora	Avenida d	5555-555	Espinho	true	v2	
4	33244	Tesoureiro	true	Luísa Santos	937498483	nd	luisa.santos@	3975489	Professor	Rua do Col	4444-444	Estarreja	true	v3	
5	65362	Vogal	true	Olivério Bapt	894274893	CC4	oliverio@pi-	97545409	GNR	Avenida d	1234-321	Estarreja	true	v4	
6	92839	Associado	false	Mário Silva	937498483	CC5	mario.silva@	765243	Professor	Avenida d	7777-888	Espinho	false	v5	
7	12764	Associado	true	Joana Marçal	895758866	CC6	joana_marca	6895655	Gestora	Rua do Col	4555-535	Aveiro	false	v6	

Figura 11 - Ficheiro de importação de membros

	A	B	C	D	E	F	G	H
	Número d	Encarregado	Nome	Ano	Modalidade	Grau	Regime	Pólo
	54353	Diogo Gomes	João		4 Instrumento		1 Normal	Aveiro
	98978	Vera Teixeira	Laura		6 Artes Visuais		2 Normal	Estarreja
	33244	Luísa Santos	Maria		6 Artes Visuais		2 Extraordir	Aveiro
	65362	Olivério Bapt	Santiago		5 Dança		2 Normal	Aveiro
	12764	Joana Marçal	Bernardo		3 Instrumento		1 Normal	Estarreja
	12764	Joana Marçal	Rita		8 Dança		3 Extraordir	Estarreja

Figura 12 - Ficheiro de importação de educandos

Como podemos ver, os CSV dos membros e dos alunos seguem o estilo de folhas de cálculo, em que a primeira linha (o cabeçalho) contém a designação dos dados. Ambos os ficheiros devem contar com parâmetros obrigatórios (no dos membros são os da coluna A até à M, inclusive enquanto que no dos alunos são os parâmetros das colunas A a D, inclusive) e com parâmetros adicionais.

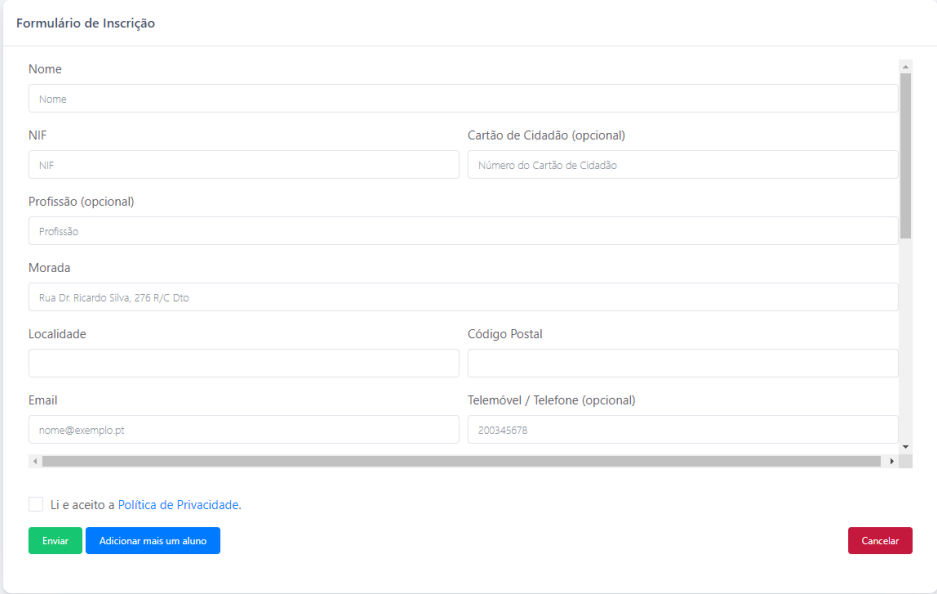
Começando pelos parâmetros obrigatórios, gostaríamos de destacar, nos membros, a coluna C, “Quotas pagas”, que irá indicar se o membro tem ou não as cotas em dia. Isto é importante, pois o membro só será admitido na plataforma caso tenha as cotas em dia, como iremos explicar com mais detalhes na secção dos registos e sua aprovação, mais adiante. Além disso, deve ser fornecida na coluna M, “Admin”, a informação, “true” ou “false”, sobre a permissão que esse membro deve ter na plataforma. Naturalmente, essa permissão terá de coincidir com a permissão que foi associada ao cargo

desse membro (que é especificado na coluna B da imagem) no ficheiro JSON de cargos detalhado anteriormente. Ainda quanto aos cargos, há ainda uma coluna que deve conter a designação do cargo de cada membro. Essa designação terá de estar incluída no JSON dos cargos importado e detalhado acima, isto é, qualquer ficheiro CSV de membros que tenha membros com cargos não especificados nesse JSON de cargos, é recusado pela plataforma. Relativamente aos educandos, de notar que é necessário, além do nome e do ano de escolaridade, que sejam fornecidos o número de sócio e o nome do respetivo encarregado de educação (os mesmos fornecidos no ficheiro dos membros) para que possamos fazer o cruzamento dos dados entre o ficheiro dos educandos e o dos membros de modo a conseguirmos associar cada educando ao seu encarregado de educação.

Relativamente aos parâmetros adicionais, estes devem ser os mesmos parâmetros descritos no ficheiro JSON de parâmetros que detalhamos acima. Caso haja inconsistências entre este JSON e os CSV no que diz respeito aos parâmetros (por exemplo, parâmetros no JSON ausentes no CSV e vice-versa ou com designações diferentes), a plataforma não irá permitir prosseguir com a configuração. Isto é válido tanto para os parâmetros adicionais dos encarregados de educação como para os dos educandos.

Registo

O formulário de registo de um novo utilizador divide-se em dois setores, encarregado de educação e educando, ambos dinâmicos nos dados necessários à inscrição.



Formulário de Inscrição

Nome
Nome

NIF
NIF

Cartão de Cidadão (opcional)
Número do Cartão de Cidadão

Profissão (opcional)
Profissão

Morada
Rua Dr. Ricardo Silva, 276 R/C Dto

Localidade
Código Postal

Email
nome@exemplo.pt

Telemóvel / Telefone (opcional)
200345678

☐ Li e aceito a [Política de Privacidade](#).

Enviar Adicionar mais um aluno Cancelar

Figura 13 - Página de registo

Além de um conjunto obrigatório de dados, como nome do encarregado de educação e email, nome do aluno, etc., este formulário também inclui os parâmetros adicionais importados de um JSON no processo de configuração da associação (anteriormente mencionado), o que confere uma configurabilidade adequada para associações com diferentes necessidades. Existe também a obrigatoriedade de preencher dados de, pelo menos, um aluno, mas com possibilidade de adicionar mais alunos através do botão no final da página.

A submissão do formulário está dotada de verificação dos campos inseridos ou falta deles, com verificações, por exemplo, do formato do código postal e email. O uso de um email indevido, como um já em uso por outro utilizador na plataforma, também é alvo de controlo com mensagem informativa ao utilizador.

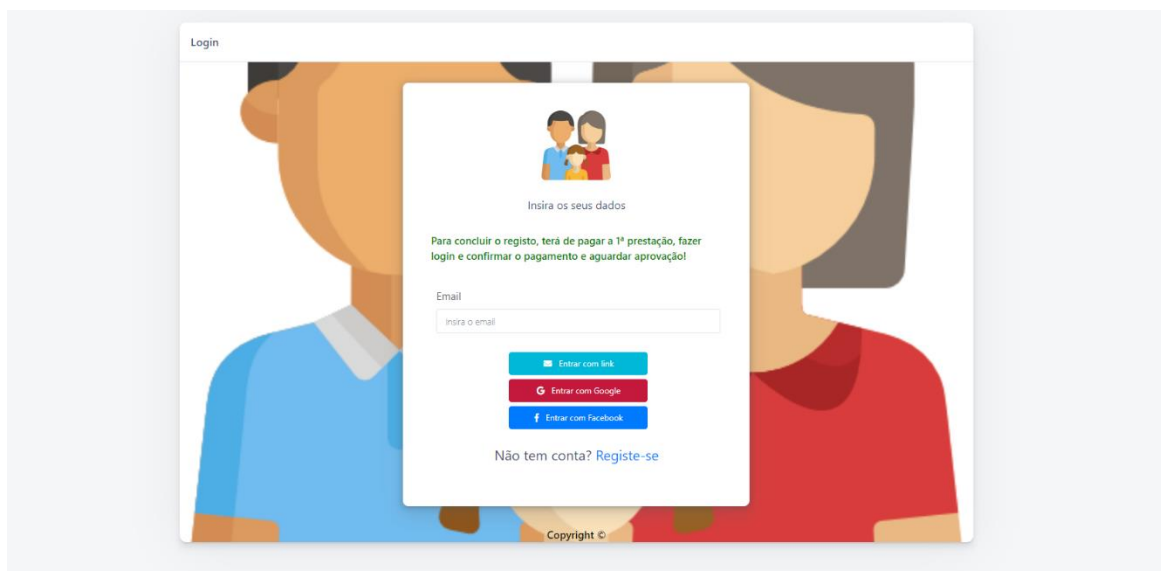


Figura 14 - Página de login após o registo bem-sucedido

O sucesso na submissão do registo leva o utilizador de volta à página inicial de login, onde está presente uma mensagem que refere o procedimento a tomar para poder ser aprovado pelos órgãos sociais da associação. O utilizador receberá também um email de confirmação de registo, com essa mesma informação.

Confirmação de pagamento

Após submeter o pedido de registo na plataforma, um utilizador deve proceder ao pagamento da primeira cota da associação. Após este pagamento, deve efetuar o login, que o levará para uma

página informativa onde apenas pode realizar *logout* ou confirmar, uma e só uma vez, que efetuou o pagamento da cota inicial.

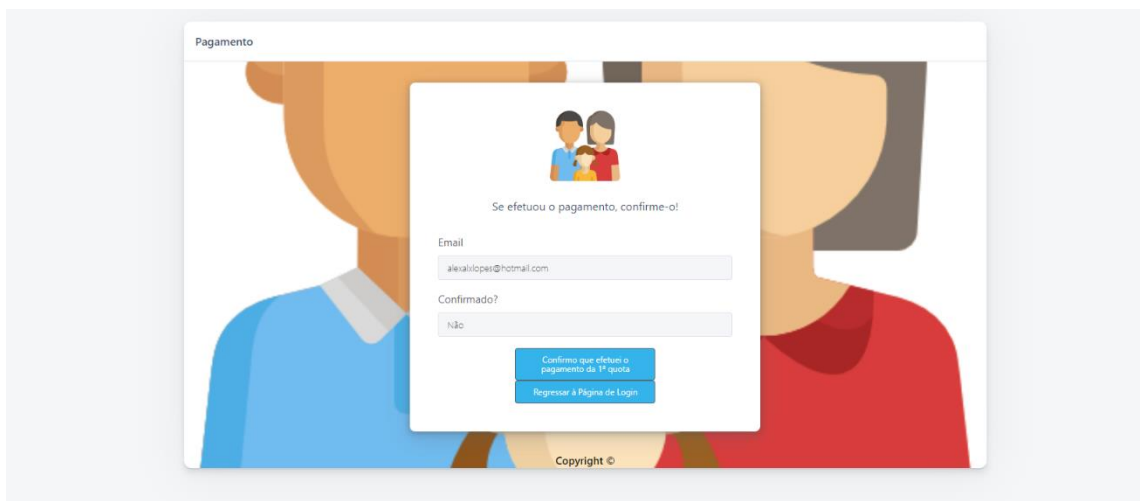


Figura 15 - Página de confirmação do pagamento da primeira cota

Todos estes passos encontram-se mencionados tanto na mensagem mostrada na página de *login* após o envio do formulário de registo, como no email de confirmação de registo enviado ao utilizador (que contém um link de redirecionamento para o login na plataforma). A ação de confirmação de pagamento da cota leva a que, na página de aprovação de registos (apenas visível para administradores/órgãos sociais), o pedido passe da zona de pedidos sem cota paga para a zona de registos com cota já paga (mais detalhes acerca da página de aprovação de pedidos encontram-se na próxima secção).

Após confirmação do pagamento da cota, mas antes da aprovação de membros pelos órgãos sociais, ao efetuar o login, o utilizador é redirecionado para a página de confirmação de pagamento. No entanto, após a aprovação, o login passa a dar acesso a toda a plataforma.

Aprovação

A página de aprovação de registos apenas está disponível para administradores/órgãos sociais e serve para aceitar ou recusar utilizadores na associação.

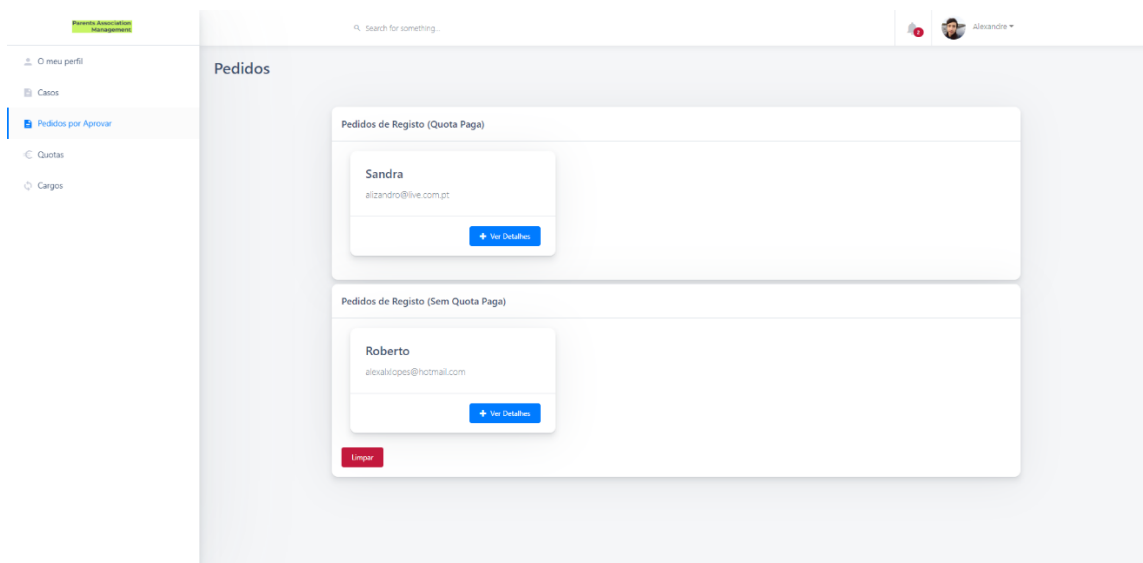


Figura 16 - Página de aprovação de registos

Está dividida em duas zonas: uma para registos que ainda não confirmaram o pagamento da cota inicial e outra para os que já o fizeram. Esta primeira zona contém um botão “limpar” que elimina automaticamente registos que ainda não pagaram cota e cuja data de inscrição ultrapassou um limite de dias definido na página de administração (que, por omissão, é de sete dias).

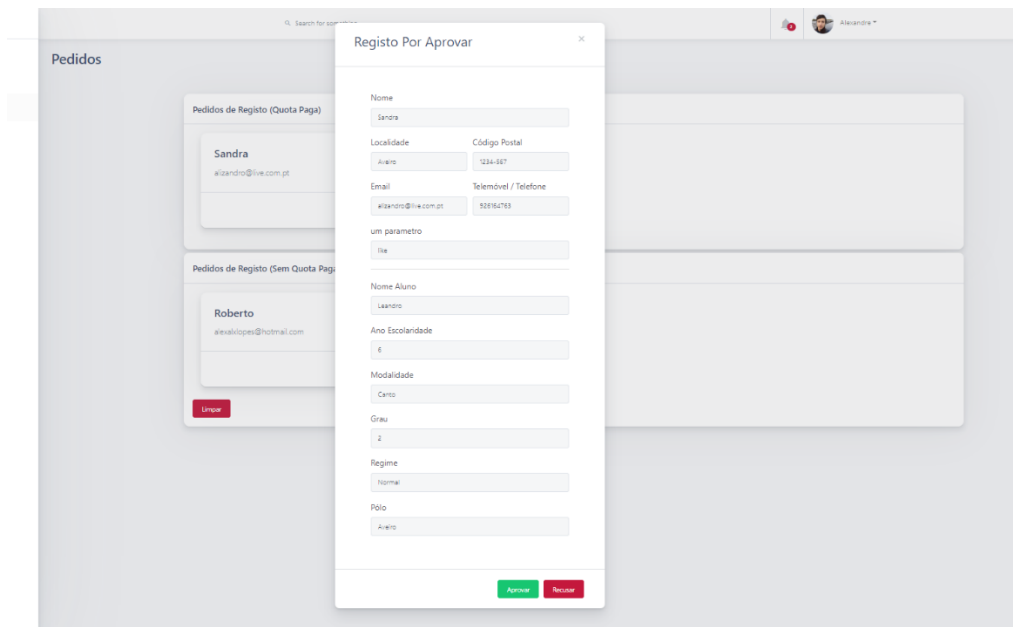


Figura 17 - Janela de detalhes do registo

Registos de utilizadores que já confirmaram o pagamento da cota inicial passam para a segunda zona mencionada, onde já não estão sujeitos a esta eliminação. Os registos têm um botão “ver detalhes” que mostra todos os dados preenchidos no formulário de inscrição, bem como os botões “aceitar” e “recusar”. O clique no botão “recusar” leva à eliminação do utilizador e ao envio de um email de rejeição para o mesmo a informá-lo dessa situação. Já o botão “aceitar”, além de enviar um email de aprovação ao utilizador, atribuí-lhe permissões de acesso à plataforma, concluindo o processo de registo.

Autenticação

O processo de autenticação pode acontecer através de link, Google ou Facebook, já anteriormente mencionados. Este processo está sujeito à verificação da existência da conta na base de dados para tomada de decisões neste processo. Isto é, se se verificar que o utilizador não tem registo na plataforma, uma mensagem com esta informação ser-lhe-á mostrada. Se o utilizador tem registo, mas não foi ainda aprovado por membros dos órgãos sociais, será redirecionado para a página de confirmação de pagamento de cota, onde poderá ou não confirmar esse pagamento caso o tenha ou não feito previamente.

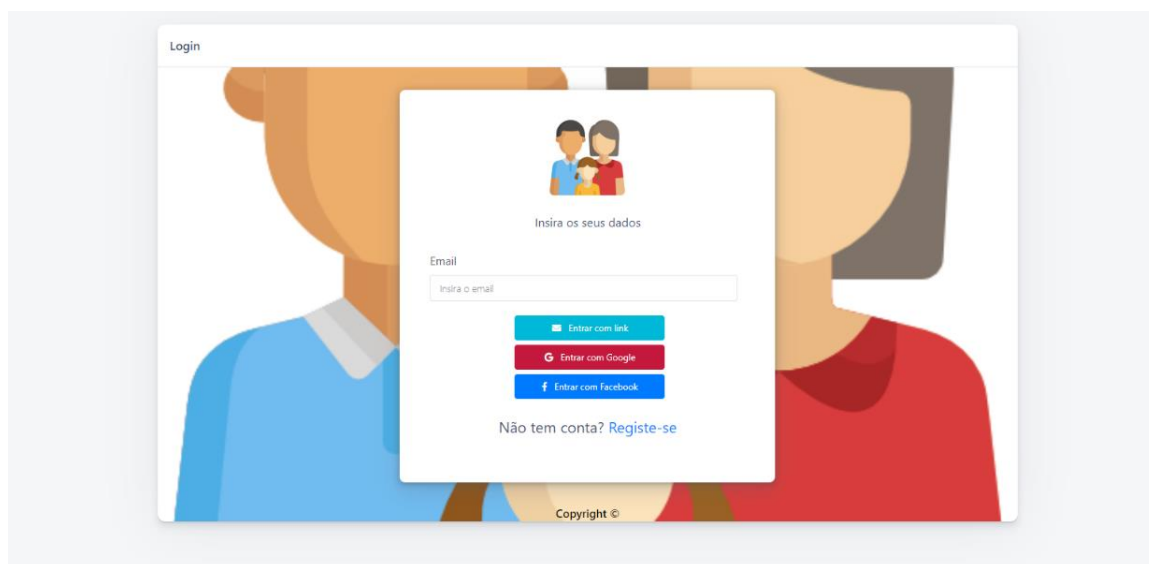


Figura 18 - Página de login

Por fim, se o utilizador tem registo e já foi aprovado na plataforma, efetuará o login com sucesso, sendo redirecionado para a plataforma, com acesso congruente com as suas permissões (se for membro dos órgãos sociais terá permissões de administrador e acesso a páginas e outras permissões exclusivas de administradores, como por exemplo, a criação de registos de cotas).

Perfil

Assim que o utilizador se autentica com sucesso, é redirecionado para a sua página de perfil.

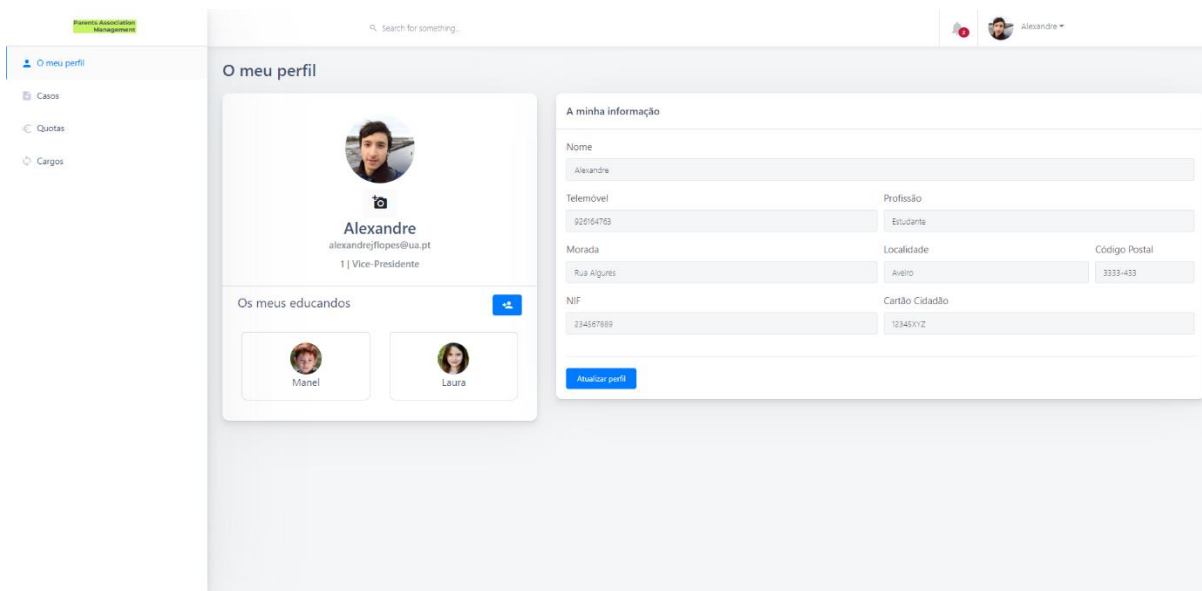


Figura 19 - Página de perfil

Nesta página, é mostrada a imagem de perfil do utilizador e outras informações pessoais, fornecidas aquando do registo (ou provenientes de um CSV importado na configuração inicial) como email, cargo, nome, profissão, NIF, morada, etc., incluindo os seus dados relativos aos parâmetros importados através do JSON fornecido na configuração inicial, sendo possível editar qualquer uma destas informações, exceto o email, que terá de ser alterado numa página diferente como detalharemos mais adiante.

Ainda nesta página inicial do perfil, é possível ver também os educandos do utilizador, identificados por uma imagem/fotografia e nome. Clicando num educando, é aberta uma janela na qual é possível consultar e editar as suas informações, incluindo a sua fotografia e os parâmetros adicionais descritos no JSON fornecido aquando da configuração inicial, como explicado anteriormente.

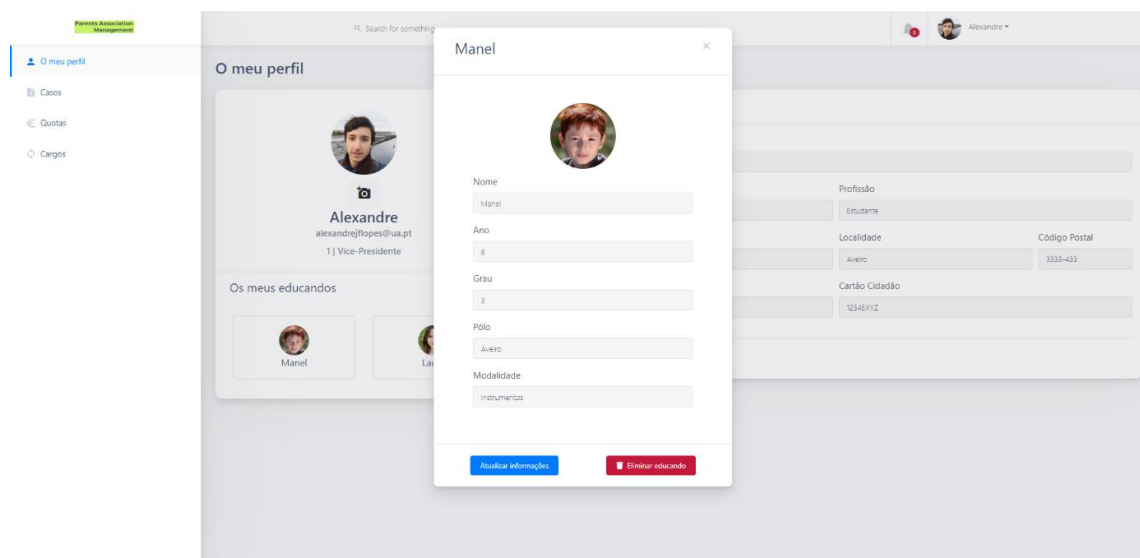


Figura 20 - Consultar informações de um educando

Além de consultar e editar, é possível ainda eliminar educandos, caso o utilizador assim o desejar, deixando, a partir desse momento, de estar visíveis na plataforma. Como esta operação não é reversível, é solicitado ao utilizador, através de uma caixa de confirmação, que a confirme. É importante notar que, tal como acontece com as restantes entidades do nosso sistema (encarregados de educação, casos, registos de cotas, comentários, etc.), ao eliminar os educandos, estes não desaparecem definitivamente da base de dados, deixando apenas de ser visíveis na interface, tal como já mencionado na secção “Cloud Firestore”, a propósito da base de dados.

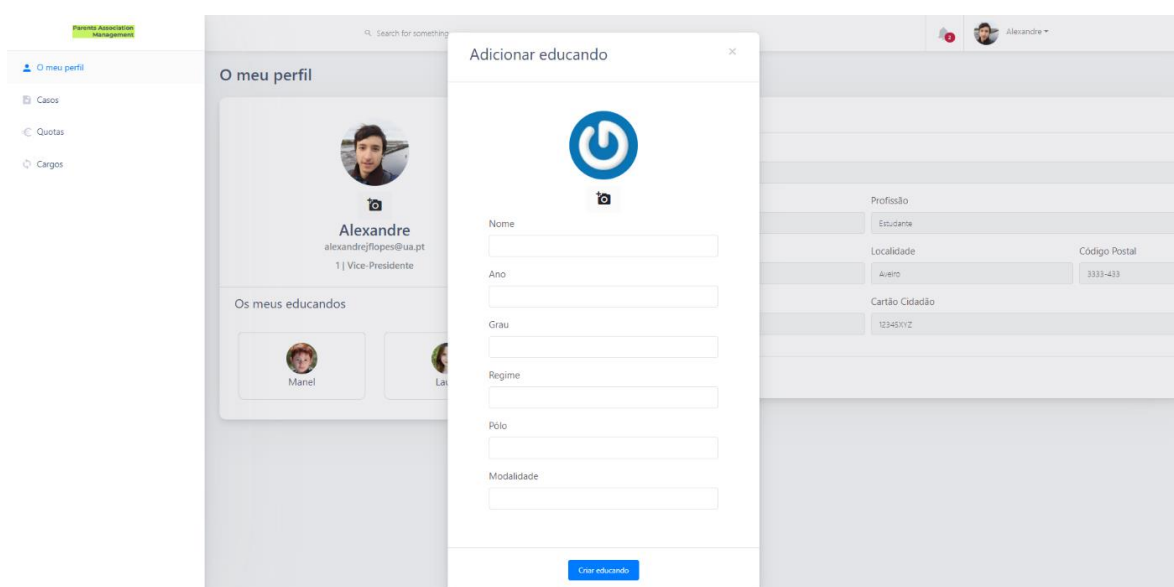


Figura 21 - Adicionar um educando

Por fim, nesta área dos educandos, o utilizador pode também adicionar novos educandos, no qual será aberta uma janela semelhante à de educandos já existentes, embora com as informações por preencher.

Definições avançadas

No canto superior direito da aplicação, estão visíveis a fotografia de perfil e o nome do utilizador. Clicando nesta área, aparece um menu descendente com opções adicionais de entre as quais “Definições avançadas”. Clicando nessa opção, o utilizador poderá ver um conjunto definições que pode configurar no seu perfil.

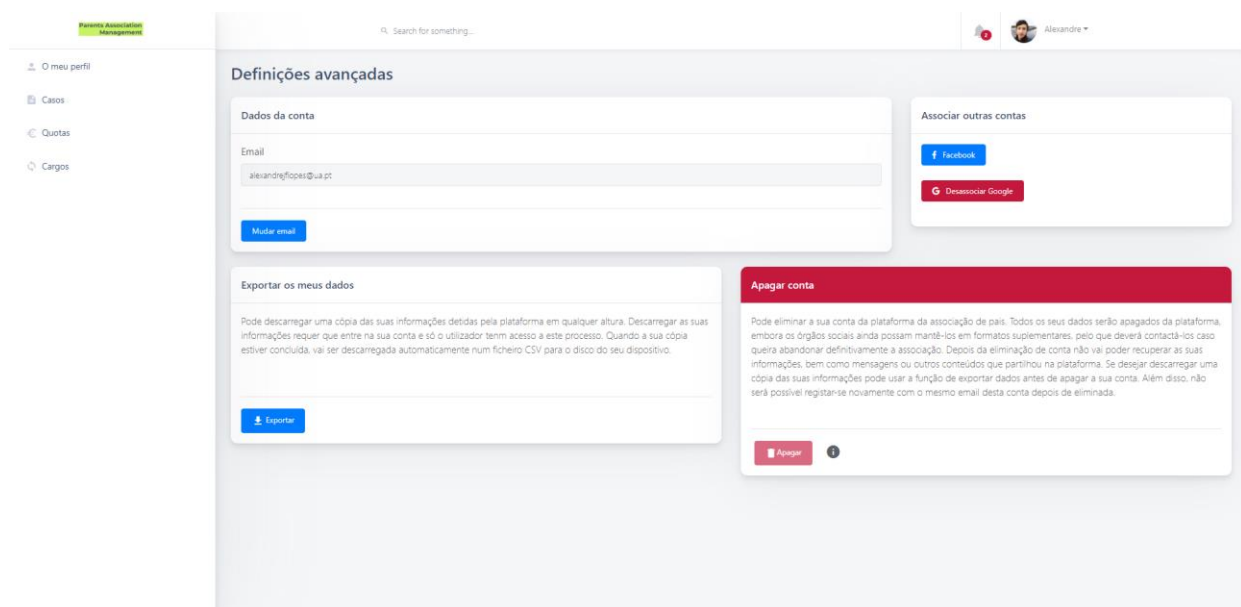


Figura 22 - Página de definições avançadas

Estas configurações refletem aspetos da conta que não são frequentemente acedidos, ao contrário das informações pessoais e dos educandos, e desencadeiam um conjunto maior de operações na plataforma, razões pelas quais optámos por encapsulá-las nesta secção separada, para não tornar a página de perfil principal demasiado confusa e prevenir a eventualidade de o utilizador realizar estas configurações por negligência.

Começamos pela mudança de email. Nesta secção, o utilizador pode verificar novamente o seu email atual e, se desejar, escrever um email diferente que gostaria de associar à sua conta. Como é

uma operação que envolve algum processamento nos bastidores do sistema, já que envolve mudanças na autenticação do utilizador (uma vez que o email é a forma principal de autenticação na plataforma), é-lhe mostrada uma caixa de confirmação para que tenha de confirmar que deseja realmente prosseguir com esta alteração.

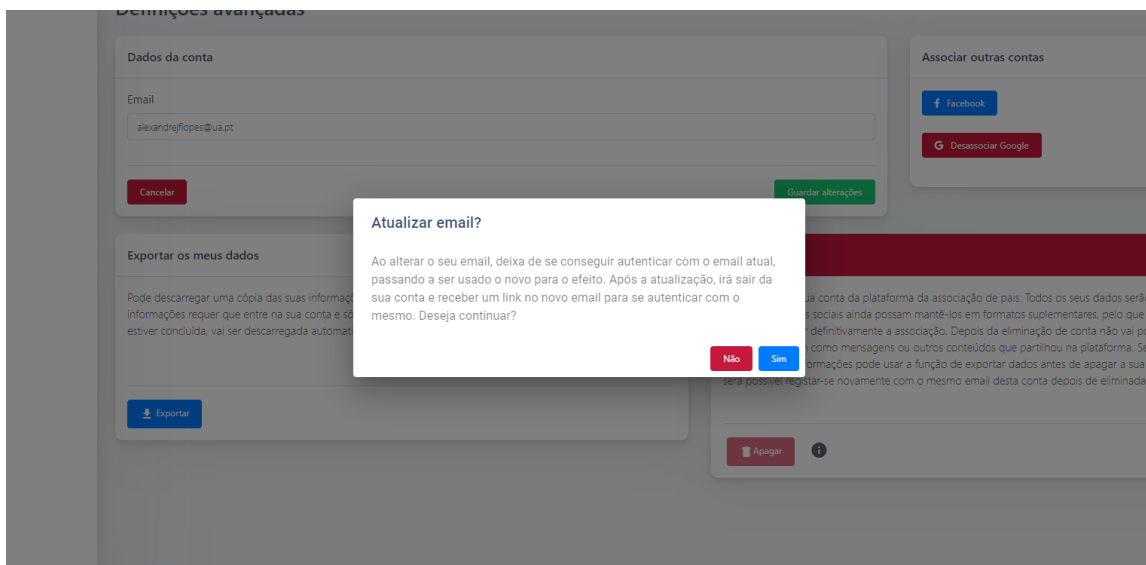


Figura 23 - Confirmação da alteração de email

Como será de esperar, esta operação não acontece se o email fornecido for o mesmo, se estiver num formato inválido ou ainda se corresponder ao email de outro utilizador da plataforma. Uma vez bem-sucedida a operação, a plataforma força o utilizador a sair da sua conta (*logout*), sendo redirecionado para a página de login. Após esta alteração, o utilizador recebe uma mensagem no novo email a confirmar a alteração com um link para se autenticar na plataforma com esse endereço. Naturalmente, com esta alteração, não será possível autenticar-se com o email anterior.

Ao lado do painel da alteração de email, está disponível um outro painel no qual o utilizador pode associar ou desassociar as suas contas do Facebook ou Google à sua conta. Caso associe algum destes provedores de identidade à sua conta, pode, a partir desse momento, autenticar-se com a sua conta na plataforma desse provedor, alternativamente ao email. O inverso se verifica quando o utilizador decide desassociar esses provedores da sua conta: deixa de conseguir autenticar-se na plataforma com essas contas.

Nesta página ainda é possível encontrar o painel no qual o utilizador pode exportar os seus dados da plataforma, descarregados para o seu dispositivo em formato CSV, num ficheiro com nome no formato "User", seguido do nome do utilizador. De entre as informações que são descarregadas,

encontram-se as informações visíveis no seu perfil, incluindo as dos seus educandos. Detalhes acerca dos mecanismos de exportação de dados são fornecidos mais adiante.

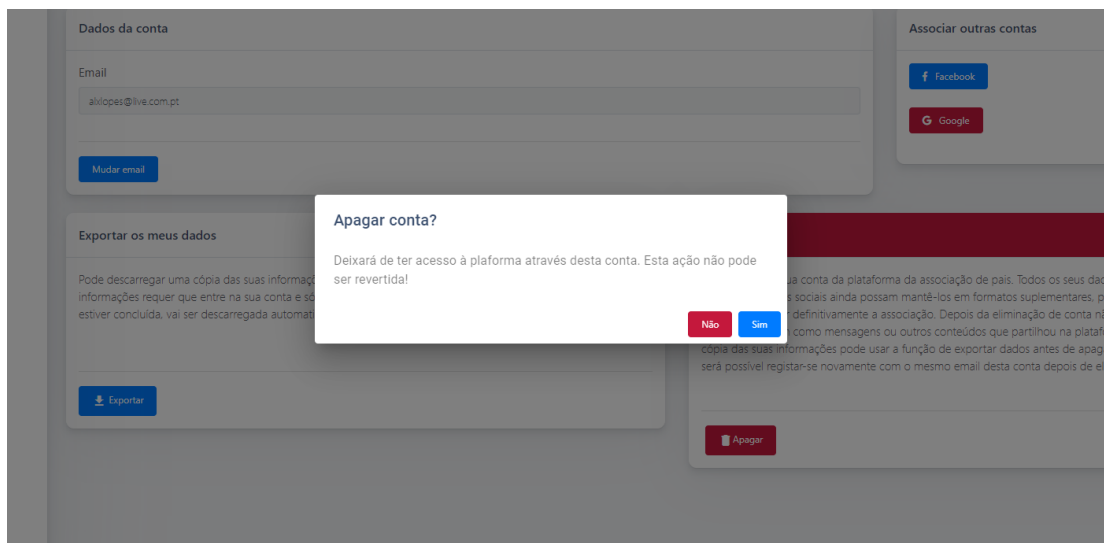


Figura 24 - Confirmação da eliminação de conta

Por fim, o utilizador pode ainda apagar a conta no painel realçado a vermelho. Como é uma operação irreversível, é também mostrada uma caixa de confirmação ao utilizador. Após a operação ter sido bem-sucedida, é automaticamente feito o *logout* do utilizador, que é redirecionado para a página de autenticação (login). Finalmente, o utilizador recebe um email a confirmar a operação e deixa de ter acesso à plataforma. De notar que a opção de eliminar conta só está disponível para utilizadores sem permissões de administrador. Esta decisão foi tomada de forma a prevenir que a plataforma fique sem administradores, inviabilizando a sua utilização, já que apenas administradores podem proceder a um conjunto de ações essenciais para manter o fluxo normal de utilização da aplicação (nomeadamente, a aprovação de registos). Se um utilizador com essas permissões desejar mesmo apagar a sua conta, terá, por exemplo, que ceder o seu cargo a outro utilizador para que lhe possa ser atribuído um cargo sem essas permissões. Mais detalhes acerca da atribuição e transição de cargos serão descritos mais adiante.

Página de administração

Os utilizadores com permissões de administrador têm ainda acesso a uma página de administração.

Administração

Dados da associação

Logótipo

Morada

Localidade

Código Postal

Email da Associação

Contacto telefónico

IBAN

Nome da Associação de Pais

Descrição

Valor da Quota (€)

Dias de persistência dos pedidos de registo

Atualizar informações da associação

Exportar dados globais da plataforma

Membros e Alunos

Pode descarregar uma cópia das informações relacionadas a membros e alunos detidas pela plataforma em qualquer altura. Descarregar as suas informações requer que entre na sua conta com privilégios adicionais, como órgão social, e só o utilizador e outros órgãos sociais têm acesso a este processo. Quando a cópia estiver concluída, vai ser descarregada automaticamente num ficheiro ZIP para o disco do seu dispositivo.

Casos

Pode descarregar uma cópia de todos os casos criados pelos utilizadores da plataforma, incluindo comentários, membros (no caso de casos privados) e informações sobre os documentos associados aos casos. Quando a cópia estiver concluída, vai ser descarregada automaticamente num ficheiro PDF para o disco do seu dispositivo.

Exportar membros e alunos

Exportar casos

Figura 25 - Página de administração

À semelhança da página de perfil, os administradores podem nesta página consultar e editar informações acerca da associação, desde o seu logótipo, email, morada, valor da cota ou até o número de dias após os quais os pedidos de registo sem cota paga devem ser apagados automaticamente, como detalhamos na secção “Aprovação”.

Por fim, nesta página é ainda possível exportar informações armazenadas na plataforma relativas a membros e respetivos educandos e a casos.

Exportar dados globais da plataforma

Membros e Alunos

Pode descarregar uma cópia das informações relacionadas a membros e alunos detidas pela plataforma em qualquer altura. Descarregar as suas informações requer que entre na sua conta com privilégios adicionais, como órgão social, e só o utilizador e outros órgãos sociais têm acesso a este processo. Quando a cópia estiver concluída, vai ser descarregada automaticamente num ficheiro ZIP para o disco do seu dispositivo.

Casos

Pode descarregar uma cópia de todos os casos criados pelos utilizadores da plataforma, incluindo comentários, membros (no caso de casos privados) e informações sobre os documentos associados aos casos. Quando a cópia estiver concluída, vai ser descarregada automaticamente num ficheiro PDF para o disco do seu dispositivo.

Exportar membros e alunos

Exportar casos

Figura 26 - Painel de exportação de dados

Neste painel o utilizador tem a opção de exportar informações dos membros e dos seus educandos em formato CSV. Através desta opção, é descarregado um ficheiro zip que contém um CSV com os dados dos membros e outro com os dados dos seus educandos, à semelhança dos CSV importados na configuração inicial. Além destes dados, o utilizador pode ainda exportar as informações de todos os casos, como título, autor, comentários e membros, para um ficheiro único em formato PDF. Mais detalhes acerca destes mecanismos de exportação são fornecidos na secção “Exportação”, mais adiante.

Cotas

Numa fase inicial, estava prevista a integração da plataforma com um serviço de pagamentos, como MB WAY ou PayPal. No entanto, após investigação, e por conflito com o requisito da gratuidade do sistema, esta solução acabou por ser abandonada e a estratégia repensada.

Desta forma, juntamente com o orientador, decidimos mudar o rumo desta funcionalidade para passar a focar-se num registo de pagamentos, com validação por parte tanto do recetor do numerário, bem como do emissor.

Quotas

Validação de Pagamentos

Data	Recetor	Emissor	Valor	Ano letivo	Confirmado pelo recetor	Confirmado pelo Emissor	Notas/Descrição do Emissor
19/06/2020	Alexandre	Roberto	7	2019/20	<input type="checkbox"/>	<input type="checkbox"/>	

Adicionar registo de pagamentos

Guardar alterações

Figura 27 - Página de cotas

Assim, na solução implementada, é permitido a qualquer utilizador com permissões de administrador/órgão social criar entradas, selecionando o nome do emissor (pessoa que entregou o dinheiro), recetor, valor da cota, ano letivo e data. Estes três últimos campos encontram-se

inicialmente pré-preenchidos de forma a facilitar e agilizar o processo, mas passíveis de serem alterados no momento (o valor da cota, que é, por omissão, 5 euros, podendo ser redefinido por um órgão social na página de administração; o ano letivo, que é calculado com base na data atual, com separação no mês 07, de julho, isto é, se o utilizador estiver, por exemplo, no dia 30/06/2020, o ano letivo será 2019/20, mas se o dia corrente já for 01/07/2020, o ano letivo será 2020/21; e a data, que corresponde à do dia corrente).

The image shows a web application interface. On the left is a sidebar with navigation links: 'O meu perfil', 'Casos', 'Pedidos por Aprovar', 'Quotas' (highlighted), and 'Cargos'. The main area displays a 'Quotas' page with a table. A modal window titled 'Novo Registo Pagamento' is open in the center. The modal contains the following fields: 'Emissor' (a dropdown menu with 'Select' text), 'Recetor' (a dropdown menu with 'Select' text), 'Valor (euros)' (a text input containing '7'), 'Ano Letivo (Exemplo formato: 2019/20)' (a text input containing '2019/20'), and 'Data' (a date input containing '22/06/2020'). At the bottom of the modal are two buttons: 'Cancelar' (red) and 'Gravar' (green). In the background, a table is partially visible with columns: 'Data', 'Recetor', 'Confirmado pelo Emissor', and 'Notas/Descrição do Emissor'. The first row shows the date '19/06/2020' and the name 'Ale...'. There is also a green button labeled 'alterações' on the right side of the background page.

Figura 28 - Criação de um novo registo de pagamento

Após preenchimento e submissão de nova entrada de cotas, a tabela desta página passará a apresentar essa nova entrada, com duas caixas de validação (*checkboxes*), para emissor e recetor, e também um campo para notas do emissor para este, se o desejar, adicionar uma descrição ou outra informação de relevância ao pagamento. É feito o controlo de permissões de alteração nesta página, garantindo que *checkboxes* e notas atribuídas a um certo membro, só por ele podem ser alteradas. Aquando de alterações, é esperado que os utilizadores confirmem as suas intenções de gravação, pelo que existe um botão “salvar alterações” que coloca os dados alterados (se existirem) na base de dados.

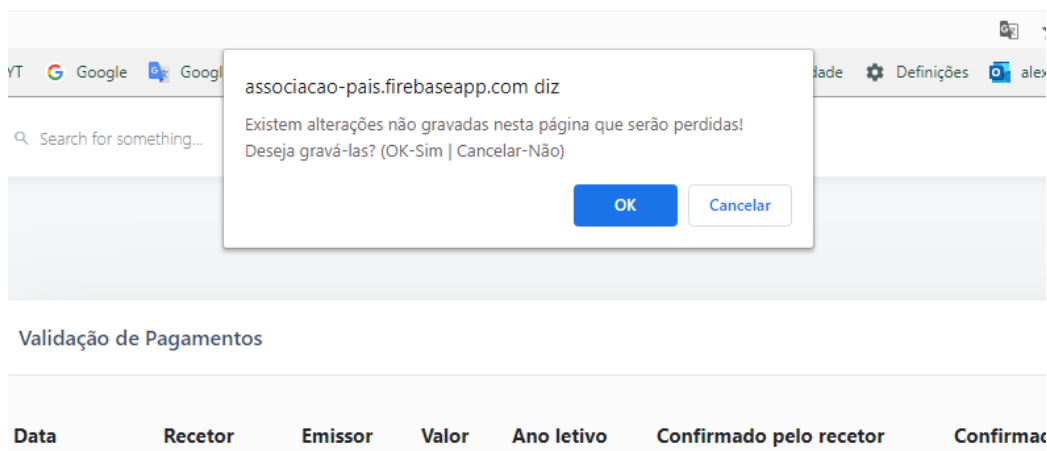


Figura 29 - Mensagem de confirmação das alterações ao sair da página de cotas

Para evitar erros de utilização, quando um utilizador tenta navegar para outra página sem usar o botão de salvar alterações, é feita uma verificação à existência de alterações e em caso positivo, é apresentada uma mensagem ao utilizador a informar de que existem alterações e a perguntar das suas intenções de as gravar ou não na base de dados. A confirmação deste *pop-up* levará à gravação dos dados na base de dados, sem que o utilizador tenha de utilizar o botão “salvar alterações”.

Casos

Na página de casos, são visíveis pequenos cartões de cada caso não arquivado, cada um com título, descrição, foto e nome do autor do caso, data de criação e botões (“ver mais” e “apagar”, sendo que este segundo apenas está visível para o autor do caso e utilizadores com permissões de administrador).

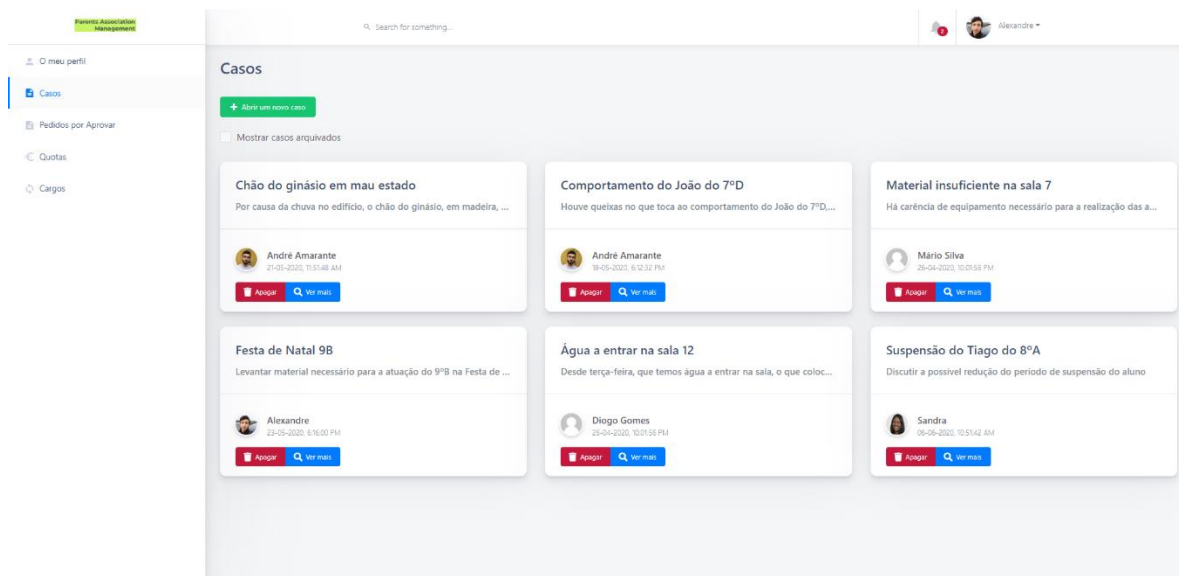


Figura 30 - Página de casos

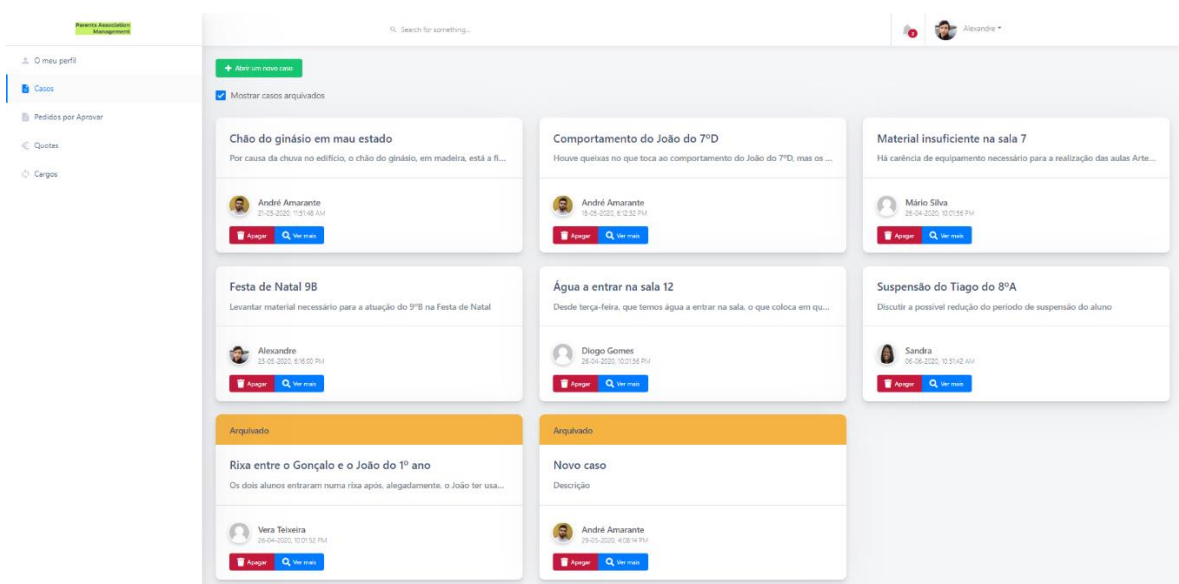


Figura 31 - Página de casos com casos arquivados visíveis

Existe também uma caixa de verificação para mostrar os casos arquivados, que aparecem com uma distinção visível ao utilizador para distinguir este tipo de casos. A cada utilizador pode aparecer uma lista de casos diferente: administradores têm acesso a todos os casos, mas utilizadores sem estas permissões apenas poderão ver os casos públicos ou de que são membros.

Search for something...

Novo caso

casos arquivados

do ginásio em mau estado

da chuva no edifício, o chão do ginásio,

André Amarante

21-05-2020, 11:51:48 AM

Ver mais

de Natal 9B

Água a entrar na sala 12

Suspens

Material

Há carênci

Mã

26-0

Apagar

Novo caso

Título do caso

Breve título do situação que origina o caso.

Descrição (opcional)

Descrição breve da situação que origina o caso.

☐ Este caso é privado.

Fechar Gravar

Figura 32 - Janela de criação de um novo caso

Nesta página, qualquer utilizador pode proceder à criação de um caso, atribuindo-lhe um título (parâmetro obrigatório), uma descrição (opcional) e informar se é ou não um caso privado (apenas acessível a certos membros). Em caso afirmativo deste último ponto, uma lista de todos os membros da associação aparece e o criador do caso seleciona aqueles a que quer dar acesso. Após a criação, o novo caso, passa a aparecer nesta página.

Search for something...

alexandre

O meu perfil

Casos

Pedidos por Aprovar

Quotas

Cargos

Chão do ginásio em mau estado

Criador: André Amarante

Data: 21/05/2020

Arquivado: Não

Membros: Sandra, Alexandre Lopes, André Amarante, Lúlia Santos, Roberto

Adicionar membros Editar título Arquivar / Desarquivar

Descrição

Por causa da chuva no edifício, o chão do ginásio, em madeira, está a ficar muito estragado.

Editar

Discussão

Sandra - 06/06/2020 10:25:12

Já convoquei uma nova reunião com a direção da escola para a semana.

Roberto - 06/06/2020 10:53:40

Vai uma equipa de manutenção ver o piso no fim de semana

Adicionar comentários

Documentos

Reunião com a direção (01/05/2020) - Ata

Adicionar documento

Figura 33 - Página individual de um caso

Ao clicar em “ver mais” associado a um caso, o utilizador é redirecionado para uma página com os detalhes deste caso. O URL associado a cada caso é único para permitir a sua divulgação. Esta página tem também um mecanismo de controlo e redirecionamento para impedir divulgação com utilizadores sem acesso ao caso. Por exemplo, o link utilizado por alguém sem login na página, leva ao redirecionamento para a página de login da plataforma; já o uso por um membro válido da plataforma, mas que não tem acesso ao caso, leva-o a ser redirecionado para a página geral de casos.

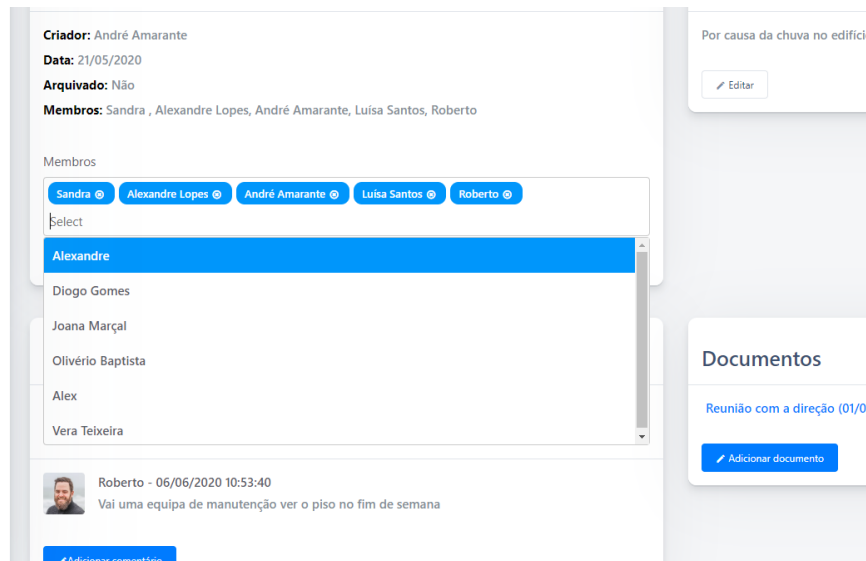


Figura 34 - Atualização de membros de um caso

A página de detalhes de um caso encontra-se dividida em quatro secções: detalhes, descrição, discussão e documentos. A primeira contém as informações gerais do caso como título, criador, data, se está ou não arquivado e membros, bem como botões para proceder à edição do título, arquivar/desarquivar e atualizar lista de membros. Os casos arquivados têm todas as opções de inserção de comentários, documentos e edições desativadas. Como é consistente na plataforma, apenas administradores ou criador do caso conseguem utilizar estes botões. A secção descrição contém apenas esse texto atribuído ao caso e a opção de editar. Já na discussão encontram-se comentários escritos, com conteúdo, data, nome e foto do utilizador, bem como a possibilidade de inserir novos comentários. Os comentários feitos pelo utilizador atual contêm também opção de editar e eliminar. Por fim, a secção de documentos contém links para documentos associados ao caso armazenados externamente, bem como a possibilidade de inserir novos. Nesta inserção, o utilizador tem de colocar um nome e o link em si.

Cargos / transição de mandato

A página de cargos permite a transição de posições nos órgãos sociais, alterando assim as permissões dos utilizadores na plataforma. A criação de nova transição de mandato mostra uma tabela com todos os associados (nome e email), o seu cargo atual e espaço para o novo cargo. Na secção de novos cargos, encontra-se um *dropdown* com a lista de cargos consultada da base de dados e a opção “Sem Mudança” para casos em que cargo atual se mantenha.

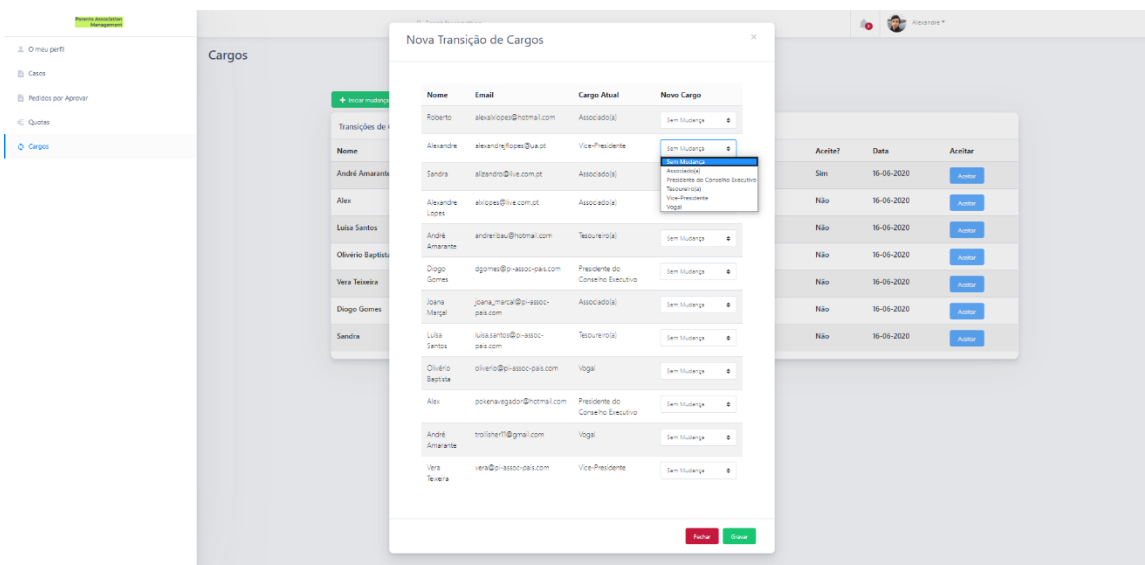


Figura 35 - Janela de atribuição de cargos

A gravação de uma transição de mandato está sujeita às validações em seguida mencionadas. Cargos de órgãos sociais não podem aparecer repetidos (por exemplo dois presidentes do conselho executivo), podendo apenas cargos não administrativos (como “associado(a)” ou outro configurado na criação da associação) serem atribuídos a mais que um utilizador. A tentativa de gravação sem que seja atribuído a nenhum utilizador um novo cargo leva a que seja mostrada uma mensagem de que a transição é inválida por nada mudar. Por fim, nenhuma transição pode levar a que algum cargo fique sem utilizador atribuído. Se for criada uma transição válida, as várias entradas serão apresentadas na tabela inicial desta página de cargos e emails informativos serão enviados aos utilizadores.

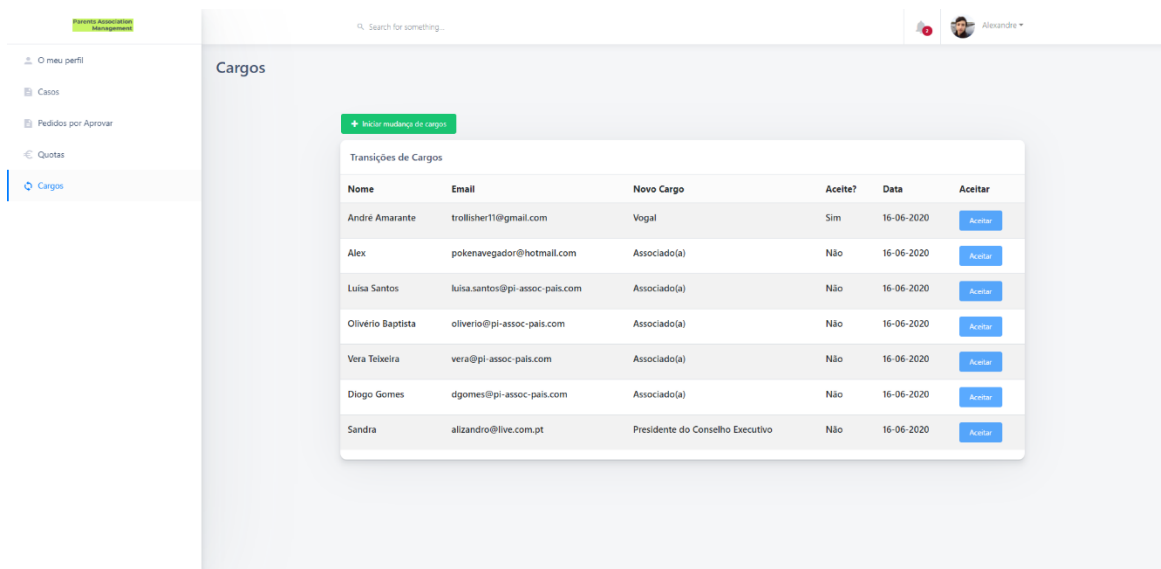


Figura 36 - Página de cargos

Entradas na tabela de cargos contém um botão para aceitação do cargo por parte do utilizador associado. Apenas este pode aceitar uma transição de cargo referente a si. A aceitação dum novo cargo poderá levar ao *logout* automático no caso em que o utilizador era administrador (membro dos órgãos sociais) e deixa de o ser, ou vice-versa. Esta mudança de permissões leva à mudança das páginas apresentadas na interface de utilizador, pelo que o *logout* automático assegura que nenhum membro da associação terá em momento algum acesso a permissões que já não lhe dizem respeito.

Notificações

As notificações são utilizadas como uma forma de subscrição a eventos que ocorram em certos casos, comentários ou mudanças de cargos. Desta forma os utilizadores podem sempre receber notificações caso recebam uma resposta num caso que estão a seguir, ou uma mudança de cargos em que estão envolvidos.

Foi desenvolvido um protótipo funcional, mas não chegou a ser implementado na versão final do projeto por restrições de tempo. Este foi desenvolvido usando o Firebase Cloud Messaging [17], que é uma solução de envio de mensagens entre plataformas, com ajuda de tutoriais [18].

Envio de emails

De forma a simplificar a autenticação de utilizadores, foi utilizada a autenticação por email através de um link gerado. Para isto, foi necessário utilizar um módulo para Node.js que permitisse

realizar Cloud Functions que enviassem emails para os utilizadores. Para o efeito, foi utilizado o módulo **Nodemailer** [19] que permite o envio de forma simplificada de emails. Para a sua configuração, apenas foi necessário especificar o serviço de email a utilizar (“gmail”, no nosso caso) e configurar as credenciais do email remetente destas mensagens que, no nosso caso, foi um email da Google criado para o projeto.

Esta ferramenta não é apenas utilizada para emails com efeito de autenticação, mas também para emails com propósitos de notificação relativos a transições de cargos e registo na plataforma.

Exportação de dados

Um dos requisitos mencionados previamente foi a exportação de dados. Esta foi realizada através do serviço de Cloud Functions da Google Firebase. Para desenvolver esta funcionalidade foi necessária a utilização de duas dependências extra. Estas são o **json2csv** [20] e o **pdfKit** [21].

A nossa plataforma suporta a exportação de dados em dois formatos. Os pais/encarregados de educação e os seus educandos podem ser exportados em formato de CSV devido à utilização de folhas de cálculo para listagem de membros e educandos posterior ao nosso projeto. Os casos existentes na plataforma poderão ser todos exportados da base de dados em formato PDF. Isto porque as informações contidas dentro de um caso não são facilmente legíveis num documento semelhante a uma folha de cálculo.

Para as exportações de dados num formato de CSV foi utilizada a dependência **json2csv**. Esta dependência, como o nome sugere, permite a conversão de objetos com o formato JSON para um formato de CSV onde as colunas correspondem aos vários atributos de cada objeto JSON. Esta dependência foi utilizada devido à sua simplicidade de instalação e de utilização. Para além do mais, é uma dependência rápida e leve.

As funções cujo objetivo é a exportação de dados através de um ficheiro PDF utilizam a dependência **pdfKit**, que permite a geração de ficheiros PDF complexos com múltiplas páginas. Esta dependência também foi utilizada porque a sua instalação e utilização é simples e intuitiva. Como a exportação de casos por PDF não era um requisito, inicialmente, e não constitui nenhuma das funcionalidades chave do nosso projeto, o PDF gerado na exportação de casos é um ficheiro muito simples com a informação correspondente aos documentos da base de dados relativos aos casos, pelo que não foi investido muito tempo na apresentação desses documentos.

Além da exportação efetuada pelas Cloud Functions acima descrita, no *frontend* foi ainda utilizada a biblioteca **JSZip** [22] . Esta biblioteca foi usada para empacotar num *zip* os ficheiros CSV com as informações dos alunos e dos educandos recebidos do *backend* no momento de exportação destes dados através da interface, para uma melhor organização para o utilizador. Esta biblioteca foi escolhida devido ao seu suporte e facilidade de utilização, já que abstrai a criação de ficheiros *zip* através da chamada a métodos simples que permitem ir adicionando os ficheiros que queremos compactar e, finalmente, gerar o ficheiro *zip* propriamente dito. Para terminar, bastou apenas utilizar também a biblioteca **FileSaver.js** [23] para desencadear a transferência deste ficheiro para o dispositivo do utilizador, também apenas através de uma chamada a um método disponibilizado pela biblioteca.

Testes e resultados

Nesta secção, iremos apresentar os vários resultados que produzimos nas várias etapas do desenvolvimento do projeto, bem como alguns testes e validações a que procedemos para nos certificarmos que a aplicação estava a ir ao encontro dos requisitos.

Protótipos

Protótipo Estático

Nas etapas iniciais, optámos por criar um protótipo estático de algumas interfaces. Estas foram muito úteis para poder validar junto do orientador do projeto a nossa interpretação da plataforma e de possíveis fluxo de interação, permitindo-nos obter uma ideia inicial das principais funcionalidades e usabilidade da plataforma.

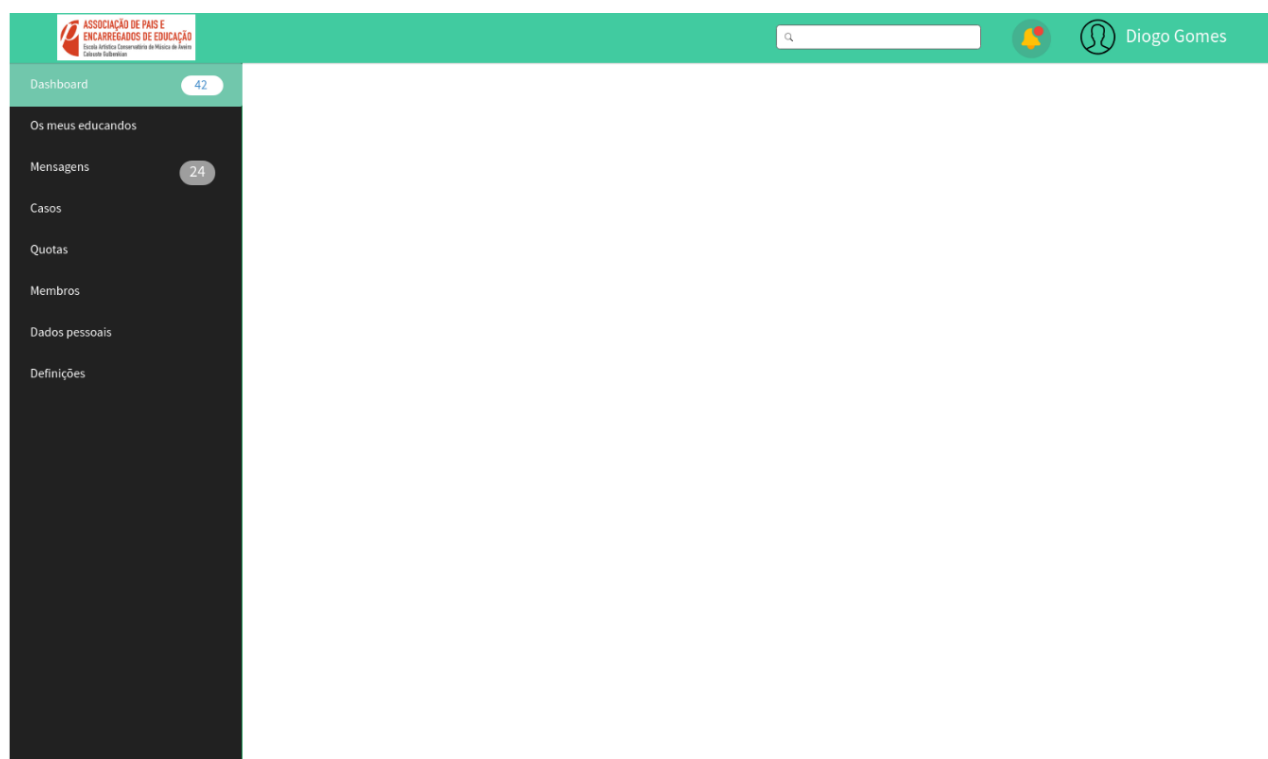


Figura 37 - Protótipo da dashboard

Foi criada uma página inicial, que viria a ser algo semelhante a uma *dashboard* com todas as funcionalidades futuras da plataforma, que o utilizador veria assim que entrasse na sua conta.

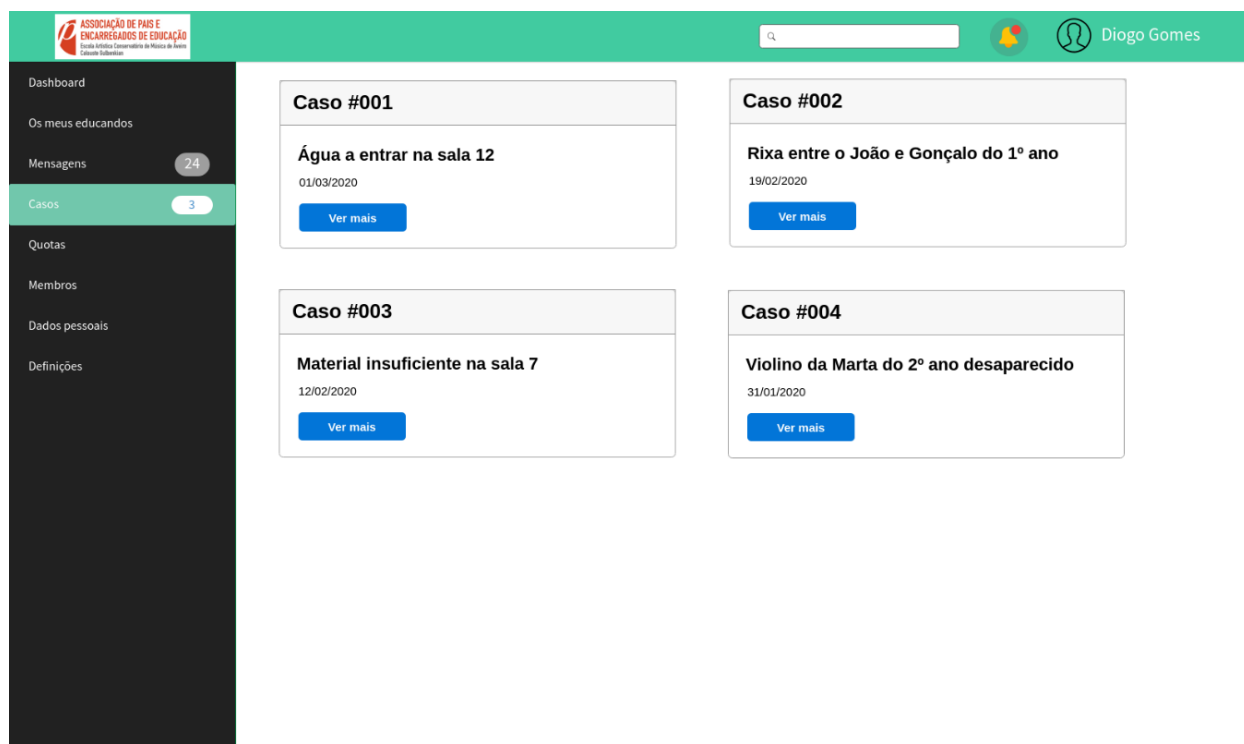


Figura 38 - Protótipo da página de casos

Um dos pontos fulcrais dos requisitos é a gestão de casos, de forma a haver comunicação rápida e facilitada entre os membros da associação. Desta forma, pareceu-nos adequado modelar as interfaces e funcionalidade relacionadas com este tópico, através desta página estática para os casos da plataforma.

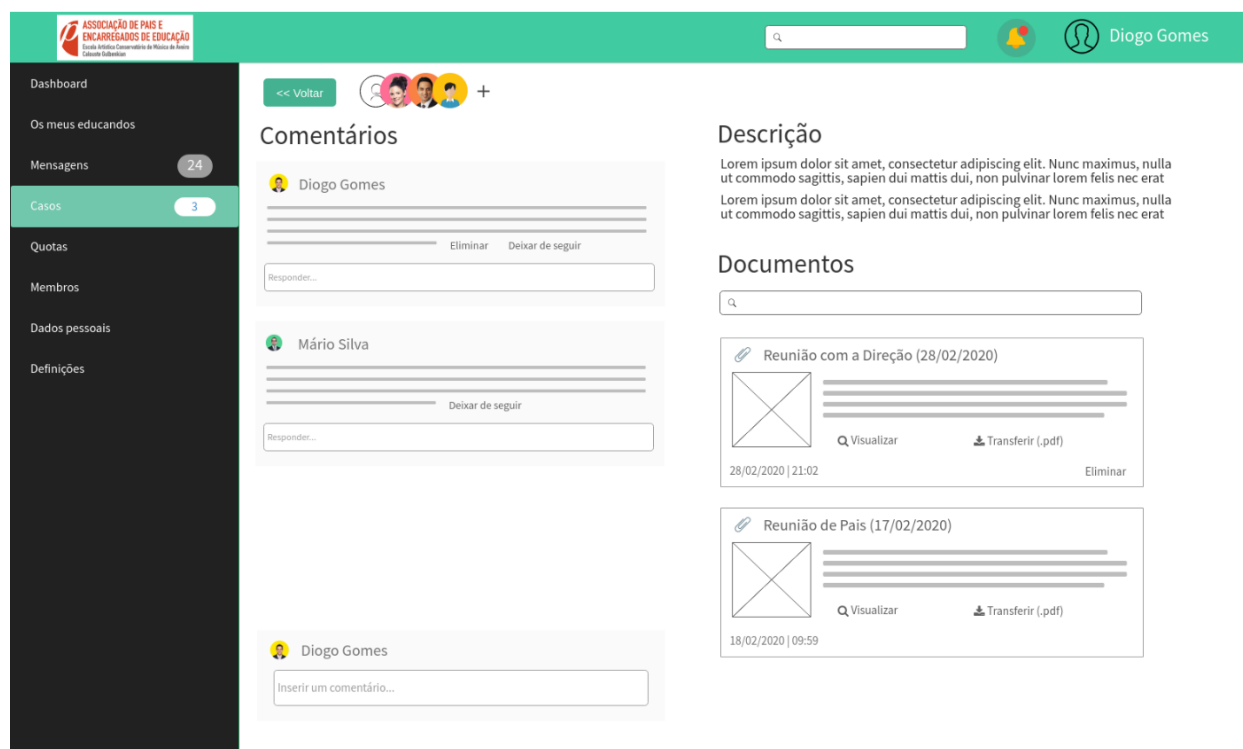


Figura 39 - Protótipo da página de um caso

Por fim, além da página de visualização geral de casos, modelámos ainda a página de um caso, onde podemos ver a sua descrição, comentários e documentos associados.

Analisando estes modelos estáticos, podemos ver que a aplicação final acabou por ser bastante semelhante ao protótipo inicial. Assim, esta abordagem tornou-se muito benéfica, pois acelerou o desenvolvimento das interfaces atuais, já que permitiu à equipa ter uma ideia da disposição dos elementos e do seu tipo de interação aquando do desenvolvimento do *frontend*.

Protótipo Funcional

Além da validação da interface, queríamos ter a certeza que as tecnologias que elegemos para desenvolver o projeto poderiam ser facilmente integradas e que os vários módulos comunicavam sem problemas entre si. Tendo isto em conta, optámos também por desenvolver um protótipo funcional para demonstrar a interação entre alguns módulos.

As seguintes imagens demonstram três páginas diferentes do protótipo.

Configuração

Nome da associacao

Nome da associação

Logótipo

Forneça uma imagem num dos seguintes formatos: .png, .jpeg, .svg

Nenhum ficheiro selecionado

Email

Email da associação

IBAN

IBAN da associação

Dados da Associação

Forneça uma folha de cálculo (.csv, .xls, etc.)

Nenhum ficheiro selecionado

Figura 40 - Protótipo da página de instalação

Entrar

Email

exemplo@email.pt

Palavra-passe

Palavra-passe

☐ Lembrar-me

Figura 41 - Protótipo da página de login

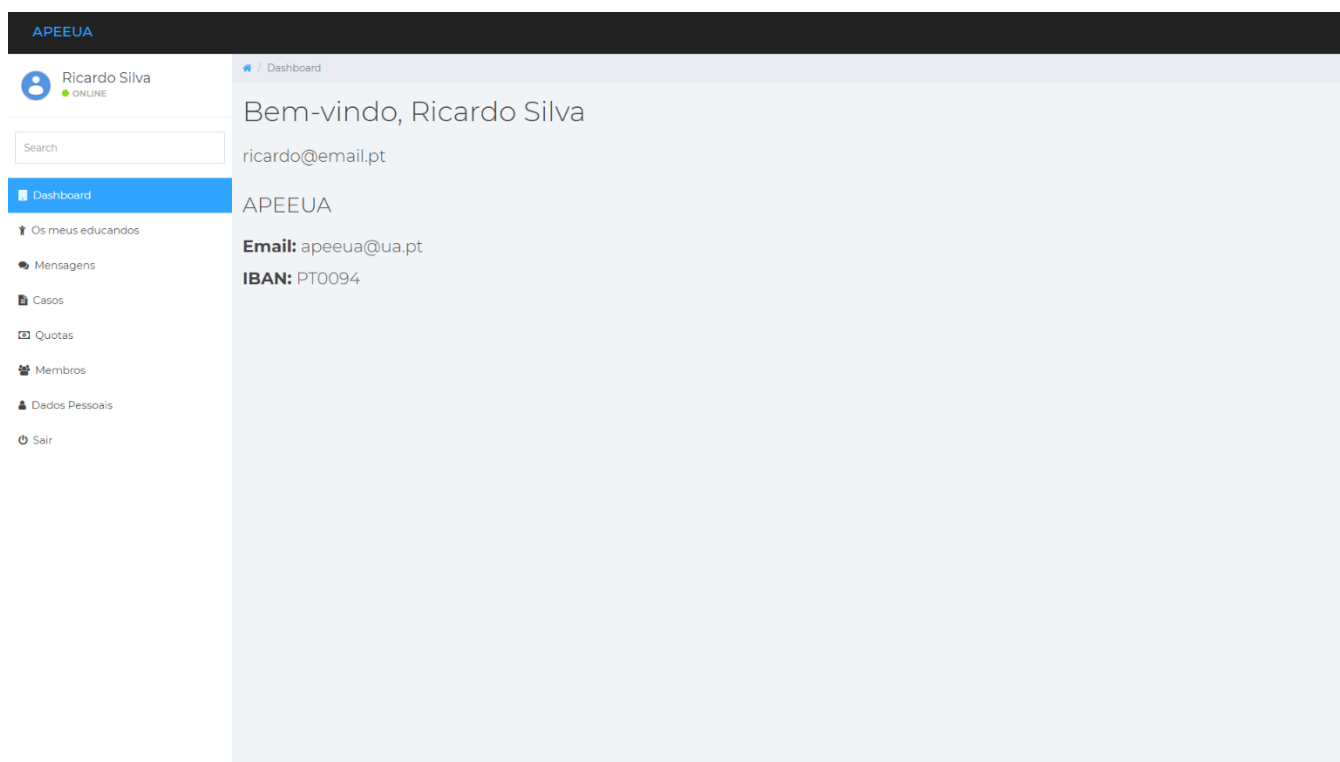


Figura 42 - Protótipo da página inicial

Na primeira figura é apresentado o formulário de instalação/configuração da aplicação. Após ser submetido, será substituído pela página de login, que é representada pela segunda figura.

A última figura exibe a página que é vista assim que o utilizador entra na plataforma, onde é possível ver os seus dados através da ligação à base de dados.

Com este protótipo demonstramos a comunicação entre a interface, as Cloud Functions e a base de dados (Cloud Firestore) e a alteração da página inicial (última figura) dependendo dos dados introduzidos no formulário de configuração da aplicação. Vale mencionar também que, no momento da submissão do formulário, é criado automaticamente, meramente para efeitos demonstrativos, um utilizador com email e palavra-passe pré-definidos, uma vez que a autenticação não foi o foco deste protótipo.

Testes de usabilidade

Tendo em conta que o orientador do projeto pertence a uma associação de pais, foram inicialmente planeados testes de usabilidade com utilizadores reais. Estes testes iriam consistir em diversos membros de uma associação existente utilizarem a plataforma para poderem dar feedback em relação a funcionalidades existentes ou funcionalidades que deveriam existir, de modo a procedermos às melhorias e correções necessárias.

Contudo, estes testes acabaram por não ser realizados, principalmente por restrições de tempo. Mesmo assim, organizámos, no decorrer de todo o desenvolvimento, sessões regulares via videoconferência com o orientador do projeto, também ele um potencial utilizador, em que lhe fomos apresentando o estado das funcionalidades da plataforma, permitindo-nos obter feedback fiável e muito útil para melhorar vários aspetos da aplicação.

Testes automáticos

Relativamente a testes automáticos ao código da aplicação propriamente dito, a equipa não se focou muito neste aspeto durante o decorrer do projeto, pelo que foi desenvolvido apenas um pequeno conjunto de testes unitários para verificar o comportamento de algumas funções.

Os testes desenvolvidos testaram principalmente funções relacionadas com a configuração inicial. Estas incluem funções que participam nos processos de validação dos ficheiros fornecidos pelo utilizador nessa fase inicial, como na validação dos ficheiros JSON de parâmetros adicionais e de cargos, bem como na dos ficheiros CSV de membros e educandos, de forma a garantirmos que a nossa implementação faz uma leitura correta desses ficheiros e reconhece os seus conteúdos e contextos para que o possa analisar e validar.

Para realizar estes testes, foi utilizada a biblioteca **Jest** [24] . A escolha desta opção deveu-se essencialmente devido ao suporte pela comunidade em geral e pela documentação disponível, bem como da sua facilidade de utilização e desempenho, já que o Jest não necessita de nenhuma configuração e corre os testes nos seus próprios processos individualmente. Além disso, a sua API também é bastante rica, fornecendo vários métodos, por exemplo, para definir valores esperados e compará-los com os valores realmente devolvidos pelas funções durante o teste, garantir que uma certa função lança exceções ou ainda definir *callbacks*, úteis para testar funções assíncronas.

Conclusão

Este projeto trata de criar uma plataforma web com o objetivo de auxiliar na gestão de uma associação de pais. Isto inclui a gestão de membros, órgãos sociais, casos, cotas e cargos. As associações de pais, geralmente, não possuem um local de reunião designado ou uma plataforma dedicada, na qual possam organizar os seus materiais ou gerir a equipa, nem uma quantidade grande de fundos monetários para investir em soluções do género, pelo que a comunicação entre os vários membros é muitas vezes limitada e a resolução de problemas dificultada.

Este produto, completamente gratuito, responde a essa necessidade de comunicação permitindo uma centralização de informações relacionadas com a associação que, por sua vez, melhora a comunicação entre os seus membros que, de outra forma, seria feita através de uma multitude de serviços de comunicação e redes sociais.

Resultados

O desenvolvimento deste projeto resultou numa plataforma web que cumpre grande parte dos requisitos especificados no início da sua conceção. Esta foi criada de forma a que utilizasse serviços de hospedagem gratuitos de forma a tornar a sua instalação gratuita. O seu design foi realizado de forma a tornar a sua usabilidade o mais intuitiva possível, tendo sido a facilidade de utilização da interface um ponto de grande foco do desenvolvimento. Além disso, o produto final possui as seguintes funcionalidades propostas no início do projeto:

- Criação de casos de forma a facilitar a sua divulgação e discussão;
- Adição de anexos/ficheiros a casos;
- Gestão do pagamento das cotas dos membros;
- Importação de encarregados de educação e educandos;
- Exportação de dados, incluindo casos e encarregados de educação e respetivos educandos;
- Diferenciação de permissões entre utilizadores com cargos na associação;
- Transferência e atribuição de cargos aos membros da associação por parte dos órgãos sociais;
- Flexibilidade de configuração de atributos relativamente às entidades de encarregado de educação e educando de forma a que a plataforma seja adaptável a diferentes associações de diferentes tipos de estabelecimentos de ensino;
- Realização de um manual de instruções que descreve o processo de instalação da plataforma.

Embora não tenha sido possível completar todos os requisitos estabelecidos nas fases iniciais deste projeto, ainda consideramos que este originou uma plataforma web com valor, devido ao número de requisitos que cumpre com sucesso. Desta forma, consideramos que, globalmente, este projeto atingiu o seu objetivo.

Benefícios e limitações da solução

A solução realizada possui vários benefícios, sendo o principal que cumpre vários dos requisitos funcionais estabelecidos no início do projeto. Além disso, também possui uma interface intuitiva e simples de utilizar. Também houve uma simplificação de vários processos da aplicação como os de registo e autenticação de forma a reduzir a sua complexidade para que os utilizadores não desistam da utilização da plataforma. A aplicação web foi adaptada a dispositivos móveis, pelo que a usabilidade não deverá ser afetada pelo dispositivo utilizado. O mecanismo de importação de dados permite que uma associação possa instalar a solução mesmo quando o ano letivo já está em progresso e, por sua vez, o mecanismo de exportação de dados permite que a plataforma possa ser desinstalada a qualquer altura, permitindo que os dados de utilização da plataforma possam ser guardados externamente.

No entanto, também existem vários fatores que conferem algumas limitações inerentes à nossa solução. As tecnologias utilizadas, escolhidas devido a serem gratuitas para utilização, possuem limitações de utilização. Estas limitações foram desprezadas devido à dimensão relativamente baixa de dados conferida pelo domínio do nosso projeto, mas em utilização real e regular da plataforma o impacto desta limitação é desconhecido, pelo que pode ser maior que o esperado. A utilização dos serviços de *cloud* da Google, também podem representar algumas falhas de performance, no sentido em que algumas operações podem demorar mais tempo dependendo da utilização coletiva da plataforma. A utilização dos serviços *cloud* referidos também confere uma forte dependência na disponibilidade desses mesmos serviços, pelo que falhas de manutenção da Google estarão fora do nosso controlo e poderão refletir-se no funcionamento deste produto.

A nossa solução também possui algumas limitações relacionadas com desenvolvimento. A funcionalidade relacionada com a transição de cargos entre membros não foi completa, pelo que o seu funcionamento ainda não é executado corretamente. Além disso, o fluxo de registo da plataforma ainda necessita de ser revisto na íntegra de forma a garantir que não existem “pontas soltas” que possam ameaçar a integridade dos dados da aplicação.

Trabalho futuro

Trabalho futuro relacionado com este projeto deverá incluir a revisão de alguns mecanismos, o acabamento de uma funcionalidade inacabada e a realização de testes.

Os fluxos relacionados com registo e o pagamento de cotas deverão ser revistos de novo de forma a garantir a consistência total do sistema e eliminar quaisquer “pontas soltas” que possam ter escapado. A funcionalidade relacionada com a transição de cargos também deverá ser melhorada para completar os requisitos funcionais da plataforma. E, por fim, deverão ser realizados testes funcionais e testes de usabilidade com utilizadores reais que, devido a restrições de tempo, não puderam ser realizados antes do final da unidade curricular para o qual este projeto foi realizado. O mesmo se aplica aos testes automáticos, que receberão mais foco para garantir que a plataforma apresenta os comportamentos esperados e cobrir o maior número de situações quanto possível.

Referências

- [1] CONFAP, “Porquê e como constituir uma Associação de Pais,” CONFAP, [Online]. Available: <http://www.confap.pt/conteudo/como-construir-associacao-pais/porque-como-construir-uma-associacao-de-pais>. [Acedido em 20 06 2020].
- [2] Pipedrive, 2020. [Online]. Available: https://www.pipedrive.com/pt?utm_content=JivoChat&utm_medium=copy_text&utm_source=partners_program. [Acedido em 26 06 2020].
- [3] meets, 2020. [Online]. Available: https://meets.com.br/?utm_source=jivo&utm_medium=jivoblog&utm_campaign=jivo%20blog&utm_term=jivo&utm_content=jivo. [Acedido em 20 06 2020].
- [4] freshsales, 2020. [Online]. Available: https://www.freshworks.com/br/freshsales-crm/?gspk=bHVjYXNjYXJ2YWxobzEwMjA&gsxid=HqfGtYsoKw6B&utm_campaign=Growsumo&utm_medium=lucascarvalho1020-Growsumo&utm_source=Growsumo&utm_tactic_id=3564352. [Acedido em 20 06 2020].
- [5] Google, “Firebase,” [Online]. Available: <https://firebase.google.com/?hl=pt>. [Acedido em 27 6 2020].
- [6] Google, “Firebase Hosting,” [Online]. Available: <https://firebase.google.com/docs/hosting>. [Acedido em 27 6 2020].
- [7] Google, “Cloud Functions para Firebase,” 16 06 2020. [Online]. Available: <https://firebase.google.com/docs/functions>. [Acedido em 20 06 2020].
- [8] M. Roberts, “Serverless Architectures,” 22 05 2020. [Online]. Available: <https://martinfowler.com/articles/serverless.html#unpacking-faas>. [Acedido em 20 06 2020].
- [9] Google, “Cloud Firestore | Firebase,” [Online]. Available: <https://firebase.google.com/docs/firestore>. [Acedido em 27 6 2020].
- [10] S. Bagsariya, “Why choose Firebase? | Features & Benefits of Firebase,” Kody Technolab Pvt Ltd, [Online]. Available: <https://kodytechnolab.com/features-benefits-firebase>. [Acedido em 27 6 2020].

- [1] Google, “Cloud Storage | Firebase,” [Online]. Available:
1] <https://firebase.google.com/docs/storage>. [Acedido em 27 6 2020].
- [1] Google, “Firebase Authentication,” [Online]. Available: <https://firebase.google.com/docs/auth>.
2] [Acedido em 27 6 2020].
- [1] Google, “Authenticate Using Facebook Login with JavaScript | Firebase,” [Online]. Available:
3] <https://firebase.google.com/docs/auth/web/facebook-login>. [Acedido em 27 6 2020].
- [1] Facebook, “React - A JavaScript library for building user interfaces,” [Online]. Available:
4] <https://reactjs.org/>. [Acedido em 27 6 2020].
- [1] Automattic, “What Is Gravatar?,” [Online]. Available: <https://pt.gravatar.com/support/what-is-gravatar/>. [Acedido em 20 06 2020].
- [1] Automattic, “Gravatar - Globally Recognized Avatars (Creating the Hash),” [Online]. Available:
6] <https://pt.gravatar.com/site/implement/hash/>. [Acedido em 27 6 2020].
- [1] Google, “Firebase Cloud Messaging,” 20 06 2020. [Online]. Available:
7] <https://firebase.google.com/docs/cloud-messaging>. [Acedido em 20 06 2020].
- [1] I. Aderinokun, “ireade/simply-notify at start,” 20 06 2020. [Online]. Available:
8] <https://github.com/ireade/simply-notify/tree/start>. [Acedido em 20 06 2020].
- [1] Nodemailer, “Nodemailer,” [Online]. Available: <https://nodemailer.com/about/>. [Acedido em
9] 26 06 2020].
- [2] M. Zeiss, “Github repository: zemirco / json2csv,” 06 06 2020. [Online]. Available:
0] <https://github.com/zemirco/json2csv>. [Acedido em 20 06 2020].
- [2] PDFKit, “PDFKit,” [Online]. Available: <https://pdfkit.org/>. [Acedido em 20 06 2020].
1]
- [2] JSZip, “JSZip,” [Online]. Available: <https://stuk.github.io/jszip/>. [Acedido em 27 6 2020].
2]
- [2] E. Grey, “eligrey/FileSaver.js: An HTML5 saveAs() FileSaver implementation,” GitHub, [Online].
3] Available: <https://github.com/eligrey/FileSaver.js/>. [Acedido em 27 6 2020].

- [2] Facebook, “Jest · Delightful JavaScript Testing,” [Online]. Available: <https://jestjs.io/>.
- 4] [Acedido em 27 6 2020].

Apêndices

Apêndice A: Complementos do projeto

- Website do produto: <https://alexandrejflopes.github.io/pi-assoc-pais/>
- Repositório: <https://github.com/alexandrejflopes/pi-assoc-pais/>
- Trello: <https://trello.com/b/mBgK41g9/assoc-pais>
- Vídeo promocional: <https://www.youtube.com/watch?v=bipmQ-HEFT8>
- Demonstração em vídeo: <https://youtu.be/SAwCtn6ajjA>