Instituto Politécnico de Viseu

Escola Superior de Tecnologia e Gestão de Viseu

Departamento de Informática



**Relatório de Projeto**

**Absence Helper**

**Realizado por:**

Nelson Andrade, nelson.andrade98mail.com, nº 16601

**Orientadores:**

ESTGV: Filipe Sá

Bizdirect: Nelson Andrade

Viseu, 2020

Instituto Politécnico de Viseu

Escola Superior de Tecnologia e Gestão de Viseu

Departamento de Informática

Relatório de Projeto

Curso de Licenciatura em

Engenharia Informática

**Absence Helper**

Realizado em:

Bizdirect

De 18 de fevereiro de 2019 a 8 de junho de 2019

Por:

Nelson Elias de Andrade

Ano Letivo 2019/2020

Orientadores:

ESTGV: Filipe Sá

Bizdirect: David Mota

Viseu, 2020

**Agradecimentos**

A realização deste relatório só foi possível com o apoio de várias pessoas que sem as quais a sua realização não seria a mesma.

Quero começar por agradecer ao meu orientador, David Mota, pois o resultado deste projeto deve-se e não só a toda a sua orientação, opiniões e críticas. Ao David digo um grande obrigado.

Agradecer igualmente ao meu outro orientador, o professor Filipe Sá, pelo apoio que este demonstrou mesmo quando o assunto se tratava de algo fora das suas especialidades.

À *Bizdirect* quero agradecer pela comodidade e disponibilidade ao longo destes 3 meses.

Aos meus colegas de curso¸ por todos os momentos vividos ao longo destes 4 anos, e agradecer também por me terem ajudado a ultrapassar imensas vezes as minhas dificuldades académicas.

Por último, mas não menos importante, agradeço à minha família por me ter apoiado sempre, sem estes, o meu caracter e ambição não eram os mesmos e sem estes o meu percurso académico não teria sido possível.

A todos, eu dedico o meu trabalho.

**Resumo**

Nos dias de hoje a evolução tecnológica é muito rápida e muito presente na nossa sociedade.

No âmbito escolar essa evolução é constante. O sistema que organiza as faltas e presenças dos alunos face às aulas que estes frequentam não acompanhou esta evolução, pois ainda se usa papéis de presenças e assinaturas comprometendo a organização destes e em pior caso acabando mesmo por perdê-los.

Face à identificação a este problema, surgiu este projeto, oferecendo uma solução digital e eficaz perante este problema.

Neste projeto, criar-se-á uma plataforma onde os alunos irão assinalar a sua presença através de um *QR code* de uma app.

O professor conseguirá consultar essas presenças e várias outras estatísticas relativas dos alunos.

Esta plataforma terá um sistema de notificações para os alunos, como para os professores, onde estes serão alertados para aulas prestes a começar ou quando estes faltam à mesma.

**Abstract**

Nowadays, the technology evolution is very fast and has an enormous presence in our society.

In a school environment this evolution is continuous. However, the system that files and organizes the number of presences of a student within a discipline didn´t follow that evolution, that system still uses the old presence paper with signatures compromising the organization of those same papers and with the risk of losing those papers.

This project came up as a solution to this problem, offering a digital and an efficient solution.

In this project, it will be created a platform where students can with a QR code from an app make known their presence in the classroom.

The professor within the discipline that he teaches in the same platform, can see the presences of which student and some other statistics of the students within a period.

This platform will have a notification system where students and teachers will be notified about some critical situation as missing classes or missing a class that is about to begin.

**Índice**

[**1.** **Introdução** 10](#_Toc39939639)

[**1.1** **Enquadramento do Projeto** 10](#_Toc39939640)

[**1.2** **Objetivos** 11](#_Toc39939641)

[**1.3** **Estrutura e conteúdo do relatório** 15](#_Toc39939642)

[**2** **Estado de Arte** 17](#_Toc39939643)

[**2.1** **A Nível de Desenvolvimento** 17](#_Toc39939644)

[**2.2** **A Nível do Produto Final** 19](#_Toc39939645)

[**3** **Ferramentas e Tecnologias** 21](#_Toc39939646)

[**3.1** **Power Apps** 21](#_Toc39939647)

[**3.2** **Common Data Service** 22](#_Toc39939648)

[**3.3** **Power Automate** 22](#_Toc39939649)

[**4** **Análise do Projeto** 23](#_Toc39939650)

[**4.1** **Arquitetura da Aplicação** 23](#_Toc39939651)

[**4.2** **Casos de Uso** 27](#_Toc39939652)

[**5** **Desenvolvimento do Projeto** 28](#_Toc39939653)

[**5.1** **Metodologias e gestão do projeto** 28](#_Toc39939654)

[**5.2** **Base de Dados** 29](#_Toc39939655)

[**5.3** **Estrutura e Interface da Aplicação** 38](#_Toc39939656)

[**6** **Setup Experimental e Testes** 48](#_Toc39939657)

[**6.1** **Setup Experimental** 48](#_Toc39939658)

[**6.2** **Testes** 48](#_Toc39939659)

[**7** **Conclusões e Trabalho Futuro** 51](#_Toc39939660)

[**7.1** **Conclusões** 51](#_Toc39939661)

[**8** **Referências** 52](#_Toc39939662)

[**9** **Bibliografia** 53](#_Toc39939663)

[**10** **Anexos** 54](#_Toc39939664)

[**10.1** **Flows** 54](#_Toc39939665)

[**10.2** **Casos de Uso** 58](#_Toc39939666)

[**10.3** **Manual de utilização** 58](#_Toc39939667)

**Índice de Figuras**

[Figura 1 - Esquema da arquitetura da aplicação 23](file:///D:\stuff\projeto\relatório\Relatório.docx#_Toc39937390)

[Figura 2 - Ambiente de Desenvolvimento 24](#_Toc39937391)

[Figura 3 - Ambiente de Base de Dados 25](#_Toc39937392)

[Figura 4 - Ambiente do Power Automate 26](#_Toc39937393)

[Figura 5 - Conceptual Data Model 29](file:///D:\stuff\projeto\relatório\Relatório.docx#_Toc39937394)

[Figura 6 - Tabela User 31](#_Toc39937395)

[Figura 7 - Tabela Disciplina 32](#_Toc39937396)

[Figura 8 - Tabela Week 33](#_Toc39937397)

[Figura 9 - User\_disciplina 34](#_Toc39937398)

[Figura 10 - Tabela Aulas 35](#_Toc39937399)

[Figura 11 - Tabela Faltas 36](#_Toc39937400)

[Figura 12 - Tabela Notificacao 38](#_Toc39937401)

[Figura 13 - Dummy screen 39](#_Toc39937402)

[Figura 14 – Começar uma aula 40](#_Toc39937403)

[Figura 15 - Analisar qr code 41](#_Toc39937404)

[Figura 16 - Disciplinas associadas (lado do professor e lado do aluno) 42](#_Toc39937405)

[Figura 17 - Inscrições de disciplinas 43](file:///D:\stuff\projeto\relatório\Relatório.docx#_Toc39937406)

[Figura 18 - Perfil da disciplina (print do lado esquerdo com scroll para baixo) 43](file:///D:\stuff\projeto\relatório\Relatório.docx#_Toc39937407)

**Índice de Tabelas**

[Tabela 1 - Tabela de Atividades 14](#_Toc39939668)

[Tabela 2 - Tabela de Comparação 19](#_Toc39939669)

[Tabela 3 - Heurísticas de Nielsen 49](#_Toc39939670)

**Lista de Acrónimos e Abreviaturas**

**API**: Application Programming Interface

**REST**: Representational State Transfer

**JSON:** JavaScript Object Notation

**SDK:** Software Development Kit

**SQL:** Structured Query Language

**UWP:** Universal Windows Platform

**WPF:** Windows Platform Foundation

**XAML:** eXtensible Application Markup Language

**GUI:** Graphical User Interface

**URL:** Uniform Resource Locator

**EI**: Engenharia Informática

**GPS**: Global Positioning System

**CDS**: Common Data Service

**ESTGV**: Escola Superior de Tecnologia e Gestão de Viseu

**IPV**: Instituto Politécnico de Viseu

# **Introdução**

No âmbito da conclusão do terceiro (3º) ano do curso de Engenharia Informática (EI) na Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Viseu foi proposta a realização de um projeto de estágio, sendo me atribuído o projeto “Absence Helper”, proposto pela empresa *Bizdirect*.

## **Enquadramento do Projeto**

### **Empresa**

A *Bizdirect* é uma empresa de consultoria nas áreas das Tecnologias de Informação do Grupo *SONAE* cujo objetivo é contribuir para a transformação dos processos de negócio dos seus clientes através de soluções e serviços diferenciadores ajustados às necessidades dos clientes, permitindo maximizar a eficiência e a performance das suas equipas e das suas infraestruturas tecnológicas. A empresa conta com mais de oitenta colaboradores, qualificados e certificados, formando uma equipa experiente a trabalhar em modelo de *Costumer* *Experience* apoiando o cliente na sua jornada através de serviços de mapeamento, desenho e gestão de projetos. O projeto foi realizado no Centro de Competências da *Bizdirect*, localizado no campus do Instituto Politécnico de Viseu.

(*Bizdirect*, 2020)

### **Curso**

O curso de EI privilegia um modelo de ensino predominantemente prático, contemplando a aplicação dos conhecimentos adquiridos em aulas práticas laboratoriais e um projeto/estágio curricular.

O principal objetivo do curso é formar especialistas qualificados para o exercício competente de uma atividade profissional, capazes de aplicar adequadamente as Tecnologias da Informação e Comunicação (TIC), conjugando fatores de índole técnica, económica, organizacional e social.

Visa também preparar os futuros profissionais para que tenham a capacidade de responder à constante evolução das TIC e das metodologias de desenvolvimento, bem como aos desafios colocados no ambiente empresarial fortemente competitivo e globalizado.

Outro objetivo importante é formar profissionais que, para além das suas aptidões técnicas, adquiram uma forte capacidade de relacionamento humano, de modo a possibilitar uma fácil comunicação dos licenciados em ambientes de trabalho em rede, colaborativos e em mudança contínua.

(“Escola Superior de Tecnologia e Gestão de Viseu,” n.d.)

### **Projeto**

O projeto desenvolvido ao longo do semestre tem como objetivo principal o desenvolvimento de uma aplicação que permita que os alunos da Escola Superior de Tecnologia e Gestão de Viseu tenham uma forma de registar a sua presença perante uma aula digitalmente.

Ao longo do tempo, o professor terá acesso a um histórico sobre as presenças ou faltas dos alunos que frequentarem o turno que este leciona.

## **Objetivos**

Na secção 1.1.3 foram discutidos com ambos os orientadores da empresa e do departamento os objetivos específicos para o projeto, sendo estes:

• **Possibilidade de o utilizador consultar as aulas que terá hoje** - para tal foi criada uma secção na aplicação onde os utilizadores poderiam consultar as aulas que teriam no dia, podendo ver o nome da disciplina e a que horas começa a aula. Poderá clicar na aula (só quando esta começar) e se for um professor, aparecerá um *qr code* com os dados da aula. Caso seja um aluno, este terá um botão para ler *qr codes*. Se este ler o *qr code* respetivo de uma aula, se estiver no IPV e estiver registado nesta disciplina, conseguirá registar a sua presença na aula.

• **Possibilidade de o utilizador consultar as disciplinas que leciona/frequenta** - para tal foi criada uma secção na aplicação onde os utilizadores poderiam adicionar, disciplinas que o utilizador frequenta ou leciona caso seja um professor.

Ao clicar numa disciplina terá acesso ao perfil da disciplina onde estão várias informações da mesma, como: turno em que está inscrito, quem leciona o turno e o respetivo gabinete e o seu horário de funcionamento e em que dias é que é lecionado o turno.

Se o utilizador for um professor, neste perfil é possível:

-Ver um gráfico que contém todas as aulas lecionadas até ao momento com a quantidade de faltas da mesma. Ao clicar numa aula, consegue ver quem faltou.

-Ver quem está inscrito neste turno e o seu registo de faltas.

Se o utilizador for um aluno, este terá 3 *toggles*: um para receber notificações de faltas (este é notificado sempre que falta a uma aula respetiva ao turno da disciplina a que está inscrito), outro para receber uma notificação sempre que uma aula está prestes a começar (30 minutos de antecedência) e outro que notifica o aluno caso este tenha faltado à aula.

• **Possibilidade de o utilizador registar as disciplinas que leciona/frequenta –** Ao pensar neste projeto foram pensados 2 cenários:

-Para cada utilizador, este já estaria inscrito nas disciplinas de acordo com a base de dados do IPV, ou seja, haveria sincronismo.

-Pensar nesta app como um auxiliar em que não há necessidade de pré-popular a base de dados para cada aluno/professor. Dar a possibilidade estes se inscreverem às disciplinas livremente.

Foi escolhida a segunda opção, pois daria menos carga de trabalho a quem fosse pré-popular a base de dados e ao mesmo tempo deu-me mais conhecimento na plataforma onde trabalhei.

Para além dos objetivos referidos em texto na secção anterior, foi elaborada uma tabela de planeamento dos vários estados da aplicação:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Calendário de Atividades | | | | | | |
| Nº Tarefa | **Tarefa** | **Data Início** | | **Data Fim** | **Duração** | **Dependência** |
| 1 | Contextualização do problema | 11/02/2020 | 12/02/2020 | | 2 dias | - |
| 2 | Estado da Arte | 13/02/2020 | 21/02/2020 | | 8 dias | - |
| 3 | Levantamento dos Requisitos | 13/02/2020 | 14/02/2020 | | 1 dia | Tarefa 1 e 2 |
| 4 | Validação dos Requisitos | 15/02/2020 | 16/02/2020 | | 1 dia | Tarefa 3 |
| 5 | Elaboração do Protótipo | 22/02/2020 | 29/02/2020 | | 7 dias | Tarefa 4 |
| 6 | Validação do Protótipo | 1/03/2020 | 1/03/2020 | | <1 dia | Tarefa 5 |
| 7 | Formação e Treino | 2/03/2020 | 10/03/2020 | | 8 dias | - |
| 8 | Implementação Front End - Login | 11/03/2020 | 12/03/2020 | | 1 dia | Tarefa 6 |
| 9 | Implementação Front End - Aulas | 12/03/2020 | 14/03/2020 | | 2 dias | Tarefa 6 |
| 10 | Implementação Front End - Disciplinas | 14/03/2020 | 15/03/2020 | | 1 dia | Tarefa 6 |
| 11 | Implementação Front End - Inscrições | 15/03/2020 | 16/03/2020 | | 1 dia | Tarefa 6 |
| 12 | Implementação Front End - Perfil | 16/03/2020 | 17/03/2020 | | 1 dia | Tarefa 6 |
| 13 | Implementação Front End - Notificações | 17/03/2020 | 18/03/2020 | | 1 dia | Tarefa 6 |
| 14 | Implementação Front End – Qr Code Professor | 18/03/2020 | 19/03/2020 | | 1 dia | Tarefa 6 |
| 15 | Implementação Front End – Qr Code Aluno | 19/03/2020 | 19/03/2020 | | <1 dia | Tarefa 6 |
| 16 | Implementação Front End – Perfil Disciplina Professor | 19/03/2020 | 20/03/2020 | | 1 dia | Tarefa 6 |
| 17 | Implementação Front End – Perfil Disciplina Aluno | 20/03/2020 | 20/03/2020 | | <1 dia | Tarefa 6 |
| 18 | Implementação Front End –Gráfico Faltas | 20/03/2020 | 21/03/2020 | | 1 dia | Tarefa 6 |
| 19 | Implementação Front End – Alunos registados | 21/03/2020 | 21/03/2020 | | <1 dia | Tarefa 6 |
| 20 | Implementação Back End - Login | 21/03/2020 | 22/03/2020 | | 1 dia | Tarefa 8 |
| 21 | Implementação Back End - Aulas | 22/03/2020 | 26/03/2020 | | 4 dias | Tarefa 9 |
| 22 | Implementação Back End - Disciplinas | 26/03/2020 | 27/03/2020 | | 1 dia | Tarefa 10 |
| 23 | Implementação Back End - Inscrições | 27/03/2020 | 29/03/2020 | | 2 dias | Tarefa 11 |
| 24 | Implementação Back End - Perfil | 29/03/2020 | 30/03/2020 | | 1 dia | Tarefa 12 |
| 25 | Implementação Back End – Qr Code Professor | 30/03/2020 | 02/04/2020 | | 3 dias | Tarefa 14 |
| 26 | Implementação Back End – Qr Code Aluno | 02/03/2020 | 03/04/2020 | | 1 dia | Tarefa 15 |
| 27 | Implementação Back End – Perfil Disciplina Professor | 04/04/2020 | 06/04/2020 | | 2 dias | Tarefa 16 |
| 28 | Implementação Back End – Perfil Disciplina Aluno | 06/04/2020 | 07/04/2020 | | 1 dia | Tarefa 17 |
| 29 | Implementação Back End –Gráfico Faltas | 07/04/2020 | 09/04/2020 | | 2 dias | Tarefa 18 |
| 30 | Implementação Back End – Alunos registados | 09/04/2020 | 10/04/2020 | | 1 dia | Tarefa 19 |
| 31 | Implementação Back End - Notificações | 11/04/2020 | 13/04/2020 | | 2 dias | Tarefa 13 |
| 32 | Flow Notificação Antecedência Aula | 14/04/2020 | 20/04/2020 | | 6 dias | - |
| 33 | Flow Notificação Começo Aula | 21/04/2020 | 23/04/2020 | | 2 dias | - |
| 34 | Flow Notificação Falta | 24/04/2020 | 25/04/2020 | | 2 dias | - |
| 35 | Implementação de edição de gabinete do professor | 26/04/2020 | 27/04/2020 | | 1 dia | - |
| 36 | Testes | 29/04/2020 | 29/05/2020 | |  | - |
| 37 | Escrita relatório | 16/03/2020 | 29/05/2020 | | 74 dias | - |
| 38 | Revisão relatório | 29/05/2020 | 02/06/2020 | | 5 dias | - |

Tabela 1 - Tabela de Atividades

## **Estrutura e conteúdo do relatório**

Este relatório é composto por vários capítulos sendo estes:

**Capítulo 1 – Introdução**

Neste capítulo é realizada, tal como o nome indica, uma introdução ao projeto elaborado, abordando um enquadramento da entidade acolhedora, do curso de Engenharia Informática e do projeto em questão. São abordados também os objetivos do projeto de modo a fornecer uma contextualização da ideia geral do mesmo.

**Capítulo 2 – Estado de Arte**

Neste capítulo foi feito um estudo bastante detalhado através da análise de algumas das aplicações que estão atualmente em funcionamento. Foi consultada alguma literatura, como artigos científicos para tentar perceber qual era o estado atual deste tipo de aplicações.

**Capítulo 3 – Ferramentas e Tecnologias Utilizadas**

Neste capítulo são enumeradas as diferentes ferramentas e tecnologias utilizadas na realização do projeto bem como feita uma pequena descrição das mesmas e o motivo da sua utilização.

**Capítulo 4 – Análise do Projeto**

Neste capítulo é feita uma análise à estrutura do projeto, falando na arquitetura da aplicação, enumerando e explicando os vários casos de uso da aplicação e por fim analisando o modelo de base de dados adotado para o projeto.

**Capítulo 5 – Desenvolvimento do Projeto**

Neste capítulo será feita uma abordagem à aplicação elaborada durante este projeto, explicando como funciona a sua utilização e elucidando acerca de alguns dos seus aspetos mais importantes.

**Capítulo 6 – Setup Experimental e Testes**

Neste capítulo será descrita a fase de Setup Experimental e Testes. Este é um processo que tem o objetivo de verificar se o projeto desenvolvido executa as funcionalidades para as quais foi destinado, em que são validados tantos os requisitos funcionais, como os não funcionais.

**Capítulo 7 – Conclusões e Trabalho Futuro**

Neste capítulo serão mostradas as conclusões referentes à elaboração deste projeto falando também sobre um possível trabalho futuro relevante ao mesmo.

# **Estado de Arte**

No presente capítulo foi feito um estudo bastante detalhado através da análise de algumas das aplicações que estão atualmente em funcionamento. Foi consultada alguma literatura, como artigos científicos para tentar perceber qual era o estado atual deste tipo de aplicações. Esta análise permitiu verificar o estado em que se encontram as aplicações, onde desde logo, foi possível concluir que muitas tinham uma intenção diferente.

## **A Nível de Desenvolvimento**

### **Android Studio**

O Android Studio é o ambiente de desenvolvimento integrado (IDE, na sigla em inglês) oficial para o desenvolvimento de apps para Android e é baseado no IntelliJ IDEA. Além do editor de código e das ferramentas de desenvolvedor avançadas do IntelliJ, o Android Studio oferece ainda mais recursos para aumentar sua produtividade na compilação de apps Android, como um sistema de compilação flexível baseado em *Gradle*, um emulador rápido com inúmeros recursos, um ambiente unificado que possibilita o desenvolvimento para todos os dispositivos Android, *Frameworks* e ferramentas de teste cheios de possibilidades, compatibilidade com C++ e NDK e muito mais. ([https://developer.android.com/studio/intro? hl=pt-br](https://developer.android.com/studio/intro?%20hl=pt-br))

### **Betty Blocks**

Betty Blocks é uma solução de desenvolvimento de aplicações baseada na nuvem que apresenta uma interface sem código e *drag-and-drop* para o desenvolvimento de aplicações de negócios.

O design da interface permite que os utilizadores configurem o texto, cabeçalhos, formulários e tabelas. O Betty Blocks apresenta modelagem visual, que permite aos utilizadores criar fluxos de trabalho de aplicativos por meio de uma interface visual. Os desenvolvedores também podem partilhar processos de aplicações com outros desenvolvedores do Betty Blocks, permitindo que outros implementem os mesmos processos nas suas próprias aplicações.

Aplicações desenvolvidas com Betty Blocks podem ser utilizados em dispositivos móveis, tablets ou desktops, e todos as aplicações têm uma interface de *back office* padrão. Para além disso, o Betty Blocks está em conformidade com os padrões de segurança ISO 27001 e oferece suporte à integração com vários serviços Web de terceiros que incluem Slack, Oracle, Dropbox, Salesforce e muito mais. [(https://www.softwareadvice.com/appdevelopment/betty-blocks-profile/)]((https:/www.softwareadvice.com/appdevelopment/betty-blocks-profile/))

### **OutSystems**

OutSystems é uma plataforma de entrega rápida de aplicações *low-code* que acelera a entrega de aplicações móveis e *Web*. Os desenvolvedores usam um ambiente de desenvolvimento integrado que cobre todo o ciclo de vida do desenvolvimento: desenvolvimento, garantia de qualidade, implementação, monitorização e gestão. (<https://www.outsystems.com/evaluation-guide/developing-with-outsystems/>)

Abaixo está uma tabela que compara estas 4 tecnologias em que o grau de comparação é subjetivo (tem como base a minha experiência pessoal e várias consultas de tutoriais e documentação das mesmas, sendo essa comparação através de um nível entre 0 e 10).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tabela de Comparação – A Nível de Desenvolvimento (0-10) | | | | |
| Funcionalidade | **Android Studio** | **Betty Blocks** | **OutSystems** | **PowerApps** |
| Low Code | 0 | 8 | 10 | 10 |
| Listar Dados de Entidades | 3 | 7 | 8 | 9 |
| Date/Time Pickers de fácil utilização | 4 | 7 | 8 | 10 |
| Facilidade de inserção de registos | 3 | 7 | 8 | 8 |
| Base de Dados Local | 6 | 7 | 7 | 10 |
| Procurar Items Específicos (Entidade) | 5 | 6 | 9 | 10 |
| User Friendly | 3 | 7 | 9 | 9 |
| Nível de Segurança | 4 | 6 | 10 | 10 |
| Verificação automática e autónoma de dados em tempo real | 4 | 7 | 9 | 9 |
| Preço Acessível | 8 | 8 | 4 | 9 |

Tabela 2 - Tabela de Comparação

Foi utilizada a Power Apps pois, após uma extensa pesquisa e discussão com a empresa, e tendo em consideração a tabela de comparação acima foi decidido por ambas as partes que esta plataforma seria a ideal para este projeto.

## **A Nível do Produto Final**

### **Attendance!**

Esta aplicação tem o intuito de ajudar o professor no decorrer das suas aulas. Tem como funcionalidades: registo de presenças de alunos na aula, guardar informações de cada aluno, guardar notas associadas a uma disciplina.( <https://apps.apple.com/us/app/attendance/id596455627>)

### **Additio App**

Esta aplicação é um auxiliar só para professores. Estes dentro da sua conta, conseguem criar várias turmas (*groups*) em que nessa turma o mesmo professor regista os vários alunos e cria numa tabela os vários parâmetros de avaliação.

Esta aplicação tem 2 outras secções, uma para sumários e a outra para o registo de presenças.

Na secção de sumários, o professor mete resumos das aulas que ocorreram ou que estão para ocorrer. Esta secção serve mais para este se guiar.

Na secção de registo de presenças, o professor consegue selecionar cada aluno e atribui-lhe um valor de “presente”, “faltou”, “atraso”, “falta justificada” ou “expulso da sala”. (<https://www.additioapp.com/en/>)

### **TeacherKit**

Esta aplicação, tem características parecidas com as aplicações referidas acima, tal como um registo de presenças perante uma disciplina, no entanto é mais fácil pesquisar por um determinado aluno do que Additio App e no Attendance!. É também possível criar tabelas de avaliação e criar várias disciplinas no lado do professor, e só este é que pode inscrever os alunos na determinada disciplina. A única distinção é esta: é possível criar uma tabela separada da avaliação de notas que serve para avaliar diferentes parâmetros de comportamento.

É preciso referir, que a interface desta aplicação é mais apelativa que as 2 acima referidas(<https://www.teacherkit.net/>).

### **Distinção**

As 3 aplicações acima referidas são bons auxiliares para gerir as presenças de uma turma ou gerir a sua avaliação, mas em relação ao Absence Helper, estas carecem nestes 4 aspetos:

O professor tem de criar sempre o registo da aula e inserir os seus respetivos dados, no entanto, no Absence Helper esse registo de aula é feito simplesmente ao clicar num botão.

O registo de presenças é todo do lado do professor, ou seja, o professor ao registar a presença dos alunos numa aula será sobrecarregado com o número de alunos/registos, enquanto que no Absence Helper, esse registo é feito pelo próprio aluno, verificando se este frequenta a disciplina e a sua localização.

No Absence Helper existem 2 funcionalidades que não existem nas acima referidas:

-Aulas do dia, que mostra as aulas que o professor ou o aluno tem durante o dia e a sua respetiva hora.

-Sistema de Notificações automático – criação de notificações automáticas que avisa o aluno quando este está prestes a ter uma aula, quando esta começa, quando o aluno falta a à mesma ou quando o professor muda a sala de aula da disciplina. O aluno só receberá estas notificações se assim o permitir.

# **Ferramentas e Tecnologias**

Neste capítulo irão ser faladas as tecnologias mais importantes que contribuíram para a realização do projeto. Estas tecnologias estendem-se desde a plataforma utilizada da Microsoft, a linguagem de programação, e até outros serviços da Microsoft.

## **Power Apps**

O Power Apps é um conjunto de aplicações, serviços e plataformas de dados que fornece um ambiente de programação rápida de aplicações para que seja possível criar aplicações personalizadas para qualquer necessidade empresarial. Com o Power Apps, é possível criar rapidamente aplicações empresariais personalizadas que se liguem aos dados empresariais armazenados quer estejam na plataforma de dados subjacente ou em várias origens de dados online e no local (SharePoint, Excel, Office 365, Dynamics 365, SQL Server, entre outras).

As aplicações criadas com o Power Apps disponibilizam funcionalidades avançadas de fluxo de trabalho e de lógica de negócio para transformar processos de negócio manuais em processos digitais e automatizados. Além disso, as aplicações criadas com o Power Apps têm um design interessante e podem ser executadas de forma totalmente integrada no browser ou em dispositivos móveis (telemóvel ou tablet). O Power Apps "democratiza" a experiência de criação de uma aplicação empresarial personalizada ao permitir que os utilizadores criem aplicações personalizadas com inúmeras funcionalidades.

O Power Apps também disponibiliza uma plataforma extensível que permite que os programadores profissionais interajam com os dados, apliquem lógica, criem conectores personalizados e integrem com dados externos através de programação. (<https://docs.microsoft.com/pt-pt/powerapps/powerapps-overview>)

## **Common Data Service**

O Common Data Service permite armazenar e gerir com segurança os dados usados pelas aplicações de negócios. Os dados no Common Data Service são armazenados num conjunto de entidades. Uma entidade é um conjunto de registos usados para armazenar dados, semelhante à maneira como uma tabela armazena dados numa base de dados. O Common Data Service inclui um conjunto básico de entidades padrão que abrangem cenários típicos, mas podemos também criar entidades personalizadas específicas da nossa organização e preenchê-las com dados. Os desenvolvedores podem usar os Power Apps para criar aplicações avançadas usando esses dados. Foi utilizado o Common Data Service pois permitiu que a aplicação fosse desenvolvida com recurso a esta funcionalidade sem necessitar de uma base de dados exterior. (<https://docs.microsoft.com/en-us/powerapps/maker/common-data-service/data-platform-intro>)

## **Power Automate**

O Power Automate é um serviço que o ajuda a criar fluxos de trabalho automatizados entre as suas aplicações e serviços favoritos para sincronizar ficheiros, obter notificações, recolher dados e muito mais. O Power Automate é um dos pilares do Power Platform. Fornece uma plataforma de código simples para automatização dos processos e dos fluxos de trabalho. (<https://docs.microsoft.com/pt-pt/power-automate/getting-started>)

# **Análise do Projeto**

Neste capítulo irá ser realizada uma análise aprofundada à estrutura do projeto, falando brevemente sobre a arquitetura da aplicação, explicando os casos de uso e efetuando uma análise do modelo de base de dados do projeto.

## **Arquitetura da Aplicação**

Neste ponto irá ser feita uma apresentação de um simples modelo da arquitetura do sistema utilizado neste projeto, esquematizado na Figura 1:

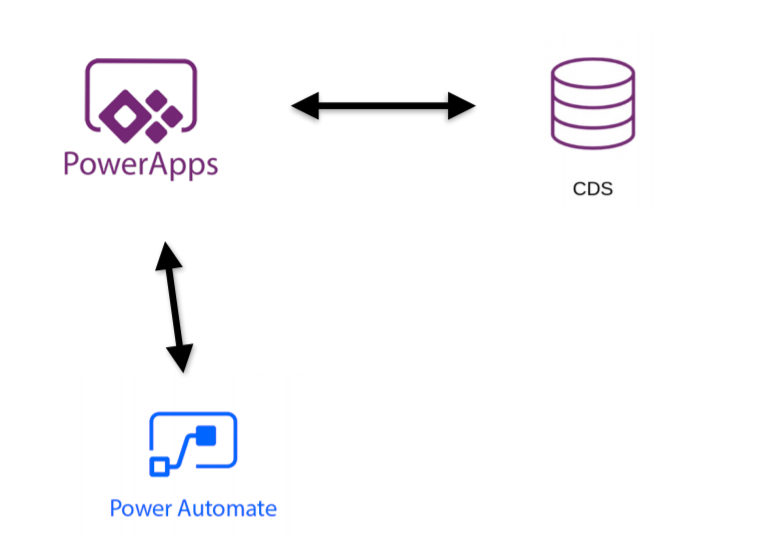


Figura - Esquema da arquitetura da aplicação

A aplicação é elaborada na plataforma Power Apps e para o seu funcionamento vai comunicar com o serviço Power Automate, pertencente aos serviços *cloud* do Dynamics365. Para os dados não serem armazenados localmente a aplicação vai também comunicar com uma base de dados no CDS (Common Data Service) onde vai armazenar e ler dados cruciais para o funcionamento da mesma. Esta base de dados faz parte, também, dos serviços *cloud* do Dynamics365.

### **Arquitetura da plataforma Power Apps**

A plataforma Power Apps, onde foi desenvolvido este projeto, está dividida em dois ambientes principais: Desenvolvimento da aplicação e Base de dados (neste caso a base dados é o CDS).

Na Figura 2 abaixo é exibido o ambiente de desenvolvimento da aplicação. Neste é evidente a presença do *Front End* no centro da figura onde a aplicação é construída. Do lado esquerdo apresentam-se os vários ecrãs da aplicação e os componentes de cada ecrã e do lado oposto são apresentadas as propriedades de cada componente do presente ecrã selecionado. As operações lógicas e qualquer função necessária ao bom funcionamento da aplicação estão presentes no topo da página e são específicas para cada componente no ecrã.

De notar que todo este ambiente se apresenta no browser.

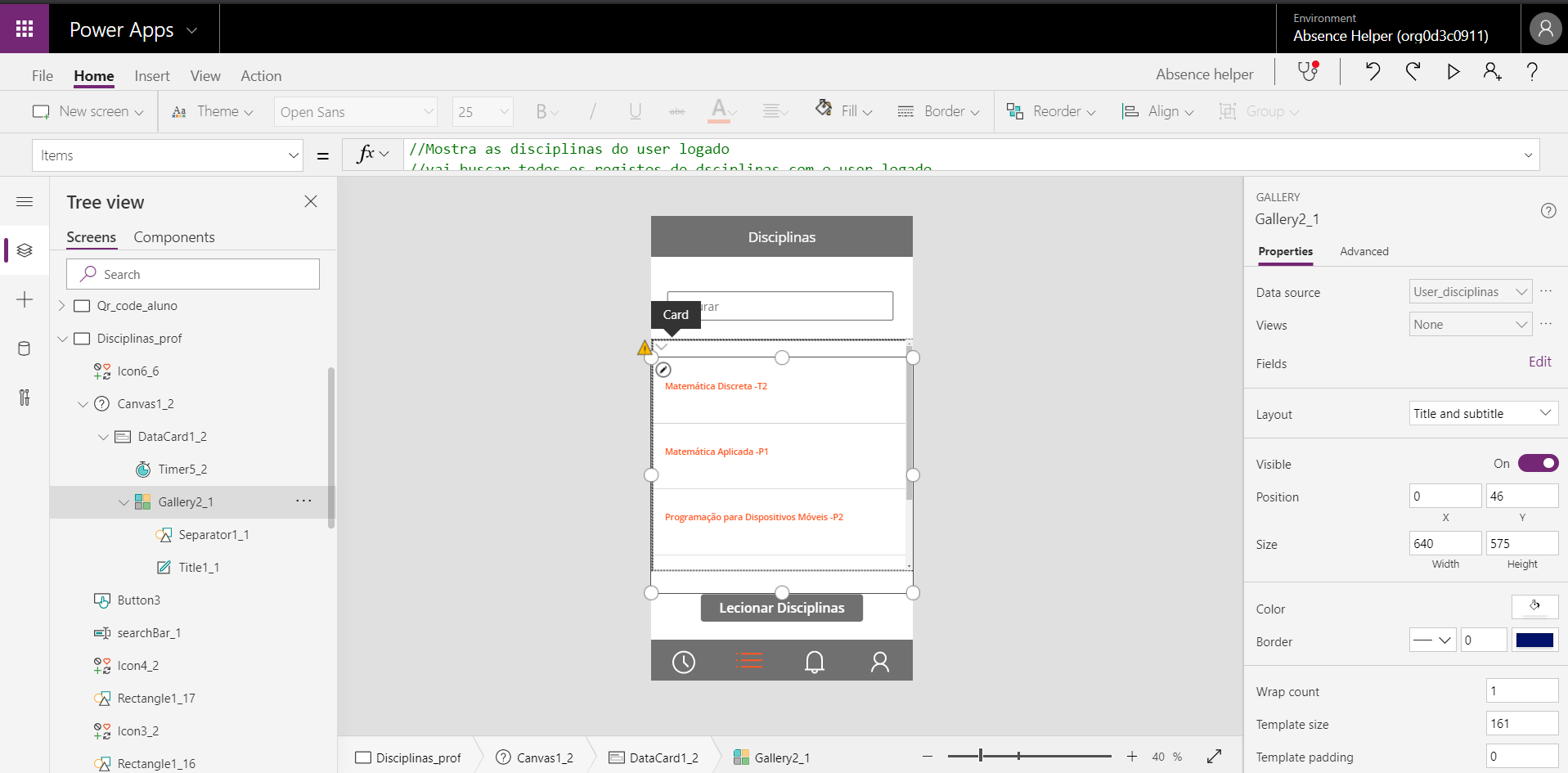


Figura - Ambiente de Desenvolvimento

Tal como o ambiente de desenvolvimento, o ambiente de Base de dados apresenta-se também no browser de forma bastante intuitiva.

Para ser possível a ligação entre a aplicação e a base de dados, o Power Apps disponibiliza aquilo que se designa de “Solução”. Uma Solução é um repositório de todos os componentes para o bom funcionamento da aplicação. Isto inclui a aplicação em si tal como toda a base de dados necessária e até quaisquer outros serviços que sejam fundamentais.

Analisando a Figura 3 abaixo é possível verificar a existência de uma solução com o nome “Absence Helper” onde estão presentes todos os componentes para o funcionamento da aplicação. Note-se que na terceira coluna é exibido o tipo de componente.

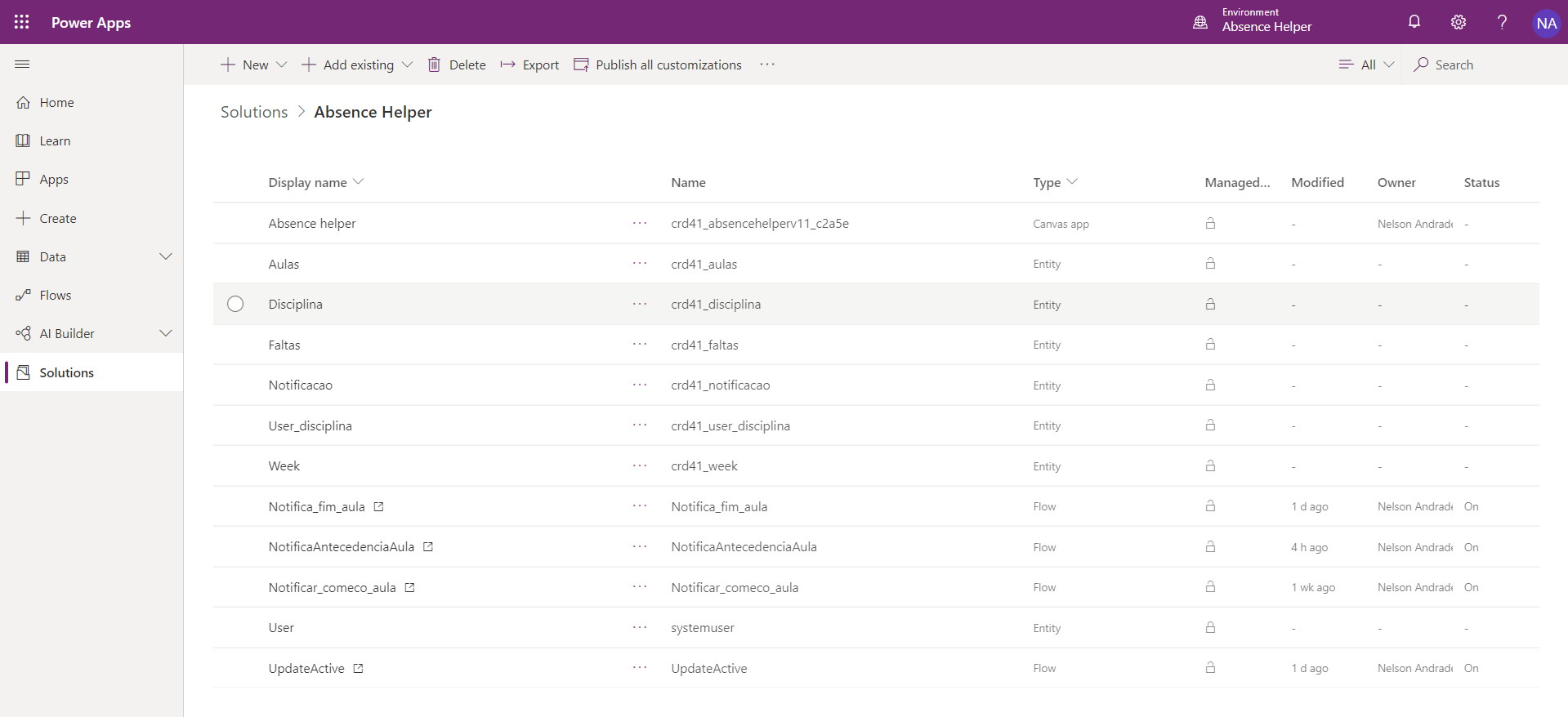


Figura - Ambiente de Base de Dados

Presente na figura anterior é possível verificar, também, a existência de todas as entidades necessárias para a aplicação. No ponto 5.2. deste relatório as entidades serão discutidas na integra e explicadas o seu funcionamento bem como os campos necessários para cada uma.

Como já referido no ponto 3.3. deste relatório, foi utilizado também o Power Automate. Em conformidade aos dois ambientes já descritos, o Power Automate é também uma plataforma na web pertencente aos serviços do Dynamics365 que permite criar fluxos de trabalho automatizados para minimizar a programação necessária para certas funcionalidades.

Na Figura abaixo mostra-se o ambiente do Power Automate. Do lado esquerdo são acessíveis certas funcionalidades como criar novos fluxos e adicionar novos dados. No centro é visível uma estrutura em forma de árvore. Este é o fluxo. Aqui é possível aceder a dados vindos da aplicação com a primeira ação presente no fluxo (“Power Apps”). Após isto é possível guardar qualquer dado vindo da aplicação em variáveis com o intuito de qualquer ação necessária ao funcionamento da aplicação.

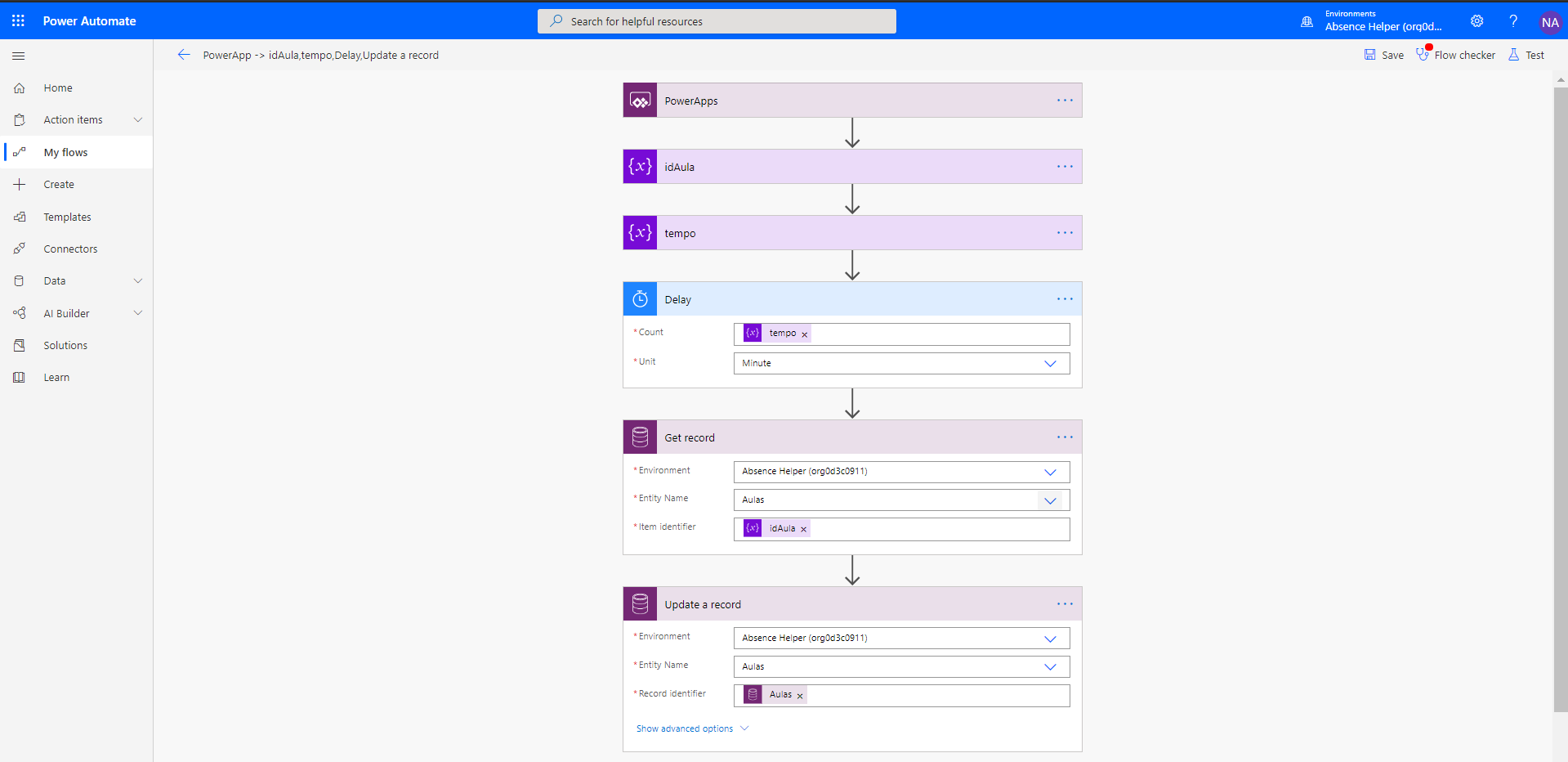


Figura - Ambiente do Power Automate

## **Casos de Uso**

No contexto deste projeto podemos considerar dois atores distintos: Aluno e Professor. De seguida irão ser apresentados os casos de uso para os diferentes tipos de utilizador.

Consulta de aulas do dia atual

Iniciar uma Aula

Consulta de Notificações

Consulta de Perfil

Consulta de disciplinas

Associar-se a uma disciplina

Desassociar-se a uma disciplina

Consultar presenças dos alunos de um turno

Editar dados relativos ao gabinete

Consultar faltas de alunos de uma aula

Eliminar falta de um aluno de uma aula

Registar presença

# **Desenvolvimento do Projeto**

No presente capítulo serão descritas as várias etapas e decisões tomadas durante o desenvolvimento do projeto, bem como a apresentação deste e das diversas funcionalidades implementadas.

## **Metodologias e gestão do projeto**

Ao longo do desenvolvimento do projeto foram realizadas várias reuniões tanto com o orientador da ESTGV bem como com o orientador da entidade acolhedora. Nestas reuniões foram discutidos assuntos como: enquadramento do aluno no projeto, pontos principais sobre o desenvolvimento do trabalho, dificuldades encontradas e discussão de passos para ultrapassar tais dificuldades.

Estas reuniões fazem parte da metodologia ágil designada Scrum, no qual o projeto é dividido em ciclos temporais, denominados de Sprints, e dentro de cada um, um conjunto de atividades a serem realizadas.

As funcionalidades que vão ser implementadas no projeto são mantidas numa lista, e a plataforma Trello é utilizada neste contexto. Nesta plataforma estão presentes um histórico dos itens dessa lista e tem como objetivo identificar as tarefas concluídas, bem como dar a conhecer os impedimentos e priorizar o trabalho ainda por realizar. (“Scrum - what it is, how it works, and why it’s awesome,” n.d.)

A cada reunião, o aluno apresenta aos orientadores as funcionalidades já implementadas e também são planeadas as próximas funcionalidades a serem implementadas.

Apesar de toda a situação de pandemia vivida durante o processo de desenvolvimento de estágio/projeto devido à Covid-19 foi possível manter contacto com os dois orientadores. E com a ajuda do Trello foi possível, também, dar continuidade a todo o processo de acompanhamento do projeto marcando reuniões via Zoom e/ou via Microsoft Teams.

## **Base de Dados**

Para a criação da base de dados da aplicação foi necessário primeiramente construir um modelo de dados que explique as características do funcionamento e comportamento da aplicação.

### **Modelo Relacional de Dados**

O Modelo Relacional de Dados representa as tabelas e relacionamentos entre estas de forma esquemática para a obtenção de uma representação da Base de Dados existente.

Para obtenção do Modelo Relacional de Dados da Base foi utilizada a ferramenta “”. Na Figura 5 é apresentado o Modelo Relacional de Dados da Base de Dados do Projeto.

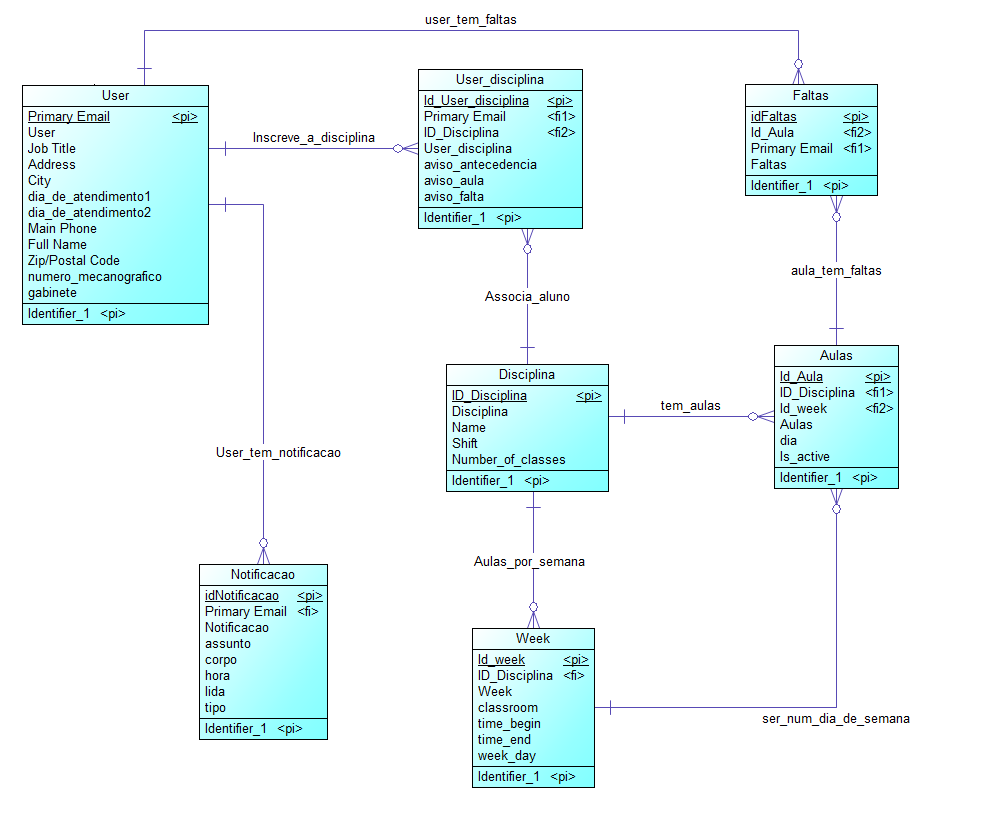


Figura - Conceptual Data Model

### **Dicionário de Dados**

A base de dados da aplicação possui sete tabelas e nesta secção cada uma vai ser analisada, bem como os seus atributos (*fields*) juntamente com uma figura que apresenta todos os atributos da tabela e o tipo de dados associado ao atributo.

**Tabela “User”**

Esta tabela da base de dados é utilizada para registar os dados dos utilizadores da aplicação. A sua chave primária é o campo “Primary Email”. Esta entidade já vem com atributos já definidos pois é uma entidade já existente no Power Apps, no entanto, foi preciso criar alguns atributos para os utilizadores do Absence Helper (sendo que estes todos não são obrigatórios):

Para o professor:

* gabinete – formato Text, em que é guardado o valor do gabinete do professor.
* dia\_de\_atendimento1 – formato Text, em que é guardado um dos 2 dias de atendimento do professor.
* dia\_de\_atendimento2 – formato Text, em que é guardado um dos 2 dias de atendimento do professor.

Para o aluno:

* numero\_mecanografico – formato Wholenumber, em que é guardado o valor do número mecanográfico do aluno.

Dos atributos já existentes da tabela User, os usados são:

* Full Name – guarda o nome do utilizador em formato Text
* Adress – guarda a morada do utilizador em formato Text
* Main Phone – guarda o telemóvel do utilizador em formato Text
* Postal Code/Zip Code – guarda o código postal do utilizador em formato Text
* City – guarda a cidade do utilizador em formato Text
* Job Title – guarda a profissão (aluno ou professor) do utilizador em formato Text

Por fim, o campo gerado automaticamente pela plataforma Power Apps, “User” que representa um id único para cada utilizador e é do formato id.

Em termos de relacionamentos esta tabela relaciona-se com as tabelas:

* “Notificacao” – que guarda notificações, em que cada registo tem o “Primary Email” a identificar para quem é a notificação.
* “User\_disciplina” – que guarda as inscrições de um utilizadores com uma disciplina, sendo que em cada registo é possível ver o “Primary Email” de quem se registou e o “Id\_Disciplina” que é o id da disciplina a qual o utilizador se registou.
* “Faltas” – que guarda todas as faltas feitas pelos alunos, em que cada registo guarda o “IdAula” que representa o id da aula a que o aluno faltou e o “Primary Email” que identifica que aluno é que faltou à dita aula.

Na figura seguinte apresenta-se a tabela analisada.

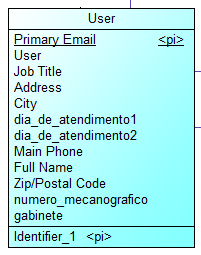


Figura - Tabela User

**Tabela “Disciplina”**

A tabela Disciplina guarda todos os turnos de todas as disciplinas.

Esta tabela está pré-preenchida com os turnos de todas as disciplinas do 2º semestre do 1º, 2º e 3º ano de EI.

A sua chave primária é “Id\_Disciplina” que representa o Id da disciplina e está no formato Wholenumber.

Os atributos desta tabela são os seguintes:

* “Name” – Noma da Disciplina, em formato Text.
* “Shift” – Turno da disciplina, em formato Text.
* “Number\_of\_classes” – Número de aulas da disciplina até ao fim do semestre, Este número foi calculado de acordo com o número de aulas por semana que um turno tem até ao fim do semestre desde que a tabela foi criada. Se o número de aulas por semana for 1 esta terá 16 aulas, se forem 2 aulas por semana, o número será 32 aulas. Em formato Wholenumber.

Por fim, o campo gerado automaticamente pela plataforma Power Apps, “Disciplina” que representa um id único para cada utilizador e é do formato id.

Em termos de relacionamentos esta tabela relaciona-se com as tabelas:

* “User\_disciplina” – como já foi referido, é nesta tabela que são guardadas as inscrições de utilizadores a disciplinas e para tal, é necessário o id da disciplina.
* “Week” – Esta tabela guarda todas as aulas que uma disciplina tem por semana e para identificar a que disciplina, estas aulas pertencem, é usado o “Id\_Disciplina”.
* “Aulas” – Esta tabela guarda todas as aulas realizadas até ao momento. Em cada registo é possível ver o atributo “Id\_Disciplina” para identificar a que disciplina esta aula pertence.

Na figura seguinte apresenta-se a tabela analisada.

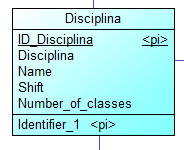


Figura - Tabela Disciplina

**Tabela “Week”**

A tabela Week guarda todas as aulas por semana de todas as disciplinas.

A sua chave primária é “Id\_Week” que representa o Id da aula por semana e está no formato Wholenumber.

Os atributos desta tabela são os seguintes:

* “Id\_Disciplina” – que representa o id da disciplina a que esta aula por semana está associada.
* “classroom” – Sala em que essa aula é lecionada, em formato Text
* “time\_begin” – Hora em que começa a aula, em formato Date.
* “Time\_end” – Hora em que acaba a aula, em formato Date.
* “Week\_day” – Dia da semana em que é lecionada a aula, em formato Text

Por fim, o campo gerado automaticamente pela plataforma Power Apps, “Week” que representa um id único para cada utilizador e é do formato id.

Em termos de relacionamentos esta tabela relaciona-se com as tabelas:

* “Disciplina” – (já referida acima)
* “Aulas” – Quando uma aula é lecionada, esta é registada e para tal, é necessário guardar no registo, o tempo de início e fim da aula que são acedidos na tabela Week pelo “Id\_week”.

Na figura seguinte apresenta-se a tabela analisada.

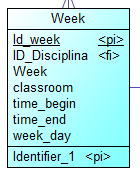


Figura - Tabela Week

**Tabela “User\_disciplina”**

A tabela User\_Disciplinas guarda todas as inscrições entre alunos e disciplinas.

A sua chave primária é “Id\_User\_disciplina” que representa o Id da dita inscrição e está no formato Wholenumber.

Os atributos desta tabela são os seguintes:

* “Id\_Disciplina” – que representa o id da disciplina da inscrição.
* “Primary Email” – que é o id do utilizador da inscrição.
* “aviso\_antecedencia” – atributo, em formato Boolean que serve para verificar se o utilizador deseja receber avisos de aulas prestes a começar.
* “aviso\_aula” – atributo, em formato Boolean que serve para verificar se o utilizador deseja receber avisos de aulas que começaram.
* “aviso\_falta” – atributo, em formato Boolean que serve para verificar se o utilizador deseja receber avisos de aulas a que faltou.

Por fim, o campo gerado automaticamente pela plataforma Power Apps, “User\_disciplina” que representa um id único para cada utilizador e é do formato id.

Em termos de relacionamentos esta tabela relaciona-se com as tabelas “Disciplina” e “User”.

Na figura seguinte apresenta-se a tabela analisada.

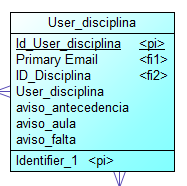


Figura - User\_disciplina

**Tabela “Aulas”**

A tabela Aulas guarda todas as aulas lecionadas até ao momento.

A sua chave primária é “Id\_Aula” que representa o Id da Aula e está no formato Wholenumber.

Os atributos desta tabela são os seguintes:

* “Id\_Disciplina” – que representa o id da disciplina da aula lecionada.
* “Primary Email” – que é o id do utilizador da inscrição.
* “Id\_week” – de onde se obtém a hora de ínicio e fim da aula e o dia da semana da aula.
* Dia – que representa o dia em que a aula foi lecionada em formato Date
* Is\_active – valor Boolean que indica se a aula ainda está ser lecionada ou se já acabou.

Por fim, o campo gerado automaticamente pela plataforma Power Apps, “Aulas” que representa um id único para cada utilizador e é do formato id.

Em termos de relacionamentos esta tabela relaciona-se com as tabelas “Disciplina” e “Week”.

Na figura seguinte apresenta-se a tabela analisada.

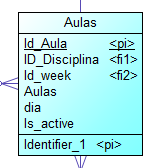


Figura - Tabela Aulas

**Tabela “Faltas”**

A tabela Faltas guarda todas as aulas faltas dos alunos

A sua chave primária é “IdFaltas” que representa o Id da Falta e está no formato Wholenumber.

Os atributos desta tabela são os seguintes:

* “Id\_Aula” – que representa o id da disciplina da aula lecionada onde a falta foi feita e está em formato Wholenumber .
* “Primary Email” – que é o id do utilizador a que pertence a falta e está em formato Text.

Por fim, o campo gerado automaticamente pela plataforma Power Apps, “Faltas” que representa um id único para cada utilizador e é do formato id.

Em termos de relacionamentos esta tabela relaciona-se com as tabelas “Aulas” e “User”.

Na figura seguinte apresenta-se a tabela analisada.

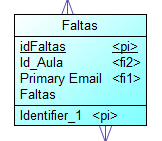


Figura - Tabela Faltas

**Tabela “Notificacao”**

A tabela Notificacao guarda todas as Notificações dos utilizadores

A sua chave primária é “IdNotificacao” que representa o Id da Notificação e está no formato Wholenumber.

Os atributos desta tabela são os seguintes:

* “Primary Email” – que é o id do utilizador a que pertence a notificação e está em formato Text.
* Assunto – que diz do que a notificação se trata e está em formato Text.
* Corpo – que é o conteúdo da notificação e está em formato Text.
* Hora – que é a hora em que a notificação foi formada e está em formato Text.
* Tipo – que representa o tipo de notificação e está em formato Text. Os tipos que existem são:
  + “Falta” – notifica quando o utilizador faltou a uma aula.
  + “Aula” – notifica quando uma aula acabou de começar (o qr code de presença está pronto a ser analisado).
  + “Sala” – notifica quando o professor muda a sala de uma disciplina a que o aluno está inscrito.
* Lida – este atributo é um Boolean e representa se o utilizador já leu esta notificação.

Por fim, o campo gerado automaticamente pela plataforma Power Apps, “Notificacao” que representa um id único para cada utilizador e é do formato id.

Em termos de relacionamentos esta tabela relaciona-se com a tabela “User”.

Na figura seguinte apresenta-se a tabela analisada.

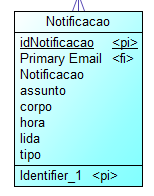


Figura - Tabela Notificacao

## **Estrutura e Interface da Aplicação**

A secção atual tem como objetivo demonstrar a implementação das funcionalidades desenvolvidas para a aplicação de acordo com os requisitos pedidos. Aqui vão ser, também, apresentadas algumas interfaces gráficas das funcionalidades implementadas, onde vai ser explicado de forma sucinta e clara o objetivo de que cada componente.

### **Dummy**

Logo após o login efetuado na aplicação Power Apps com as credenciais do Office365 disponibilizadas pela entidade escolar e a abertura da aplicação Absence Helper, o utilizador é levado à página Dummy. Esta página é como o nome indica, um *screen dummy* em que é aqui que a app verifica que tipo de utilizador é o utilizador que acabou de fazer login. Se este for um professor, este é redirecionado para a página inicial do professor, caso contrário, este é redirecionado para a página inicial do aluno.

Esta distinção é feita no email do utilizador, em que este segue a mesma estrutura que os emails do IPV, em que os emails dos alunos têm o domínio “alunos” (exemplo: [estgv16601@alunos.estgv.ipv.pt](mailto:estgv16601@alunos.estgv.ipv.pt)).

Se o email do utilizador que fez login conter a palavra “alunos” é porque é um aluno, caso contrário é um professor.



Figura - Dummy screen

### **Aulas**

Depois da verificação do tipo de utilizador este é redirecionado para a página inicial do Absence Helper. Esta é a secção de aulas, onde é possível ver as aulas que o utilizador está a ter ou vai ter nesse dia. Se for um professor, serão as aulas que este irá lecionar, caso contrário serão as aulas das disciplinas que o aluno está a frequentar.

Cada aula que é apresentada neste screen, tem o nome da disciplina, a sala de aula e a hora em que começa. Caso esta aula comece, a sua hora transforma-se para a palavra “Live” que significa que a aula “está aberta” (“a porta da sala de aula abriu-se”), podendo o professor e o aluno entrar nessa aula.

Quando uma aula acaba, esta desparece.

Enquanto uma aula não começar, a “porta da sala de aula” está fechada e o utilizador não consegue entrar.

Se faltar 30 minutos para a aula começar, os alunos e o professor são notificados com esse aviso.

Quando o utilizador consegue entrar na aula se este for:

* Professor, este ao clicar em “Começar aula” será gerado um qr code, com informações relativas à aula (id da aula e id da disciplina) e os alunos inscritos nessa disciplina serão notificados de modo a que saibam que já podem analisar o qr code de presença. Ao gerar esse qr code, é criado um registo desta aula na base de dados e quando esta acabar o atributo “Is\_active” muda o seu valor para 0. É também inserido na base de dados, para cada aluno desta disciplina, uma falta desta aula que poderá ser retirada quando o aluno analisar o qr code.

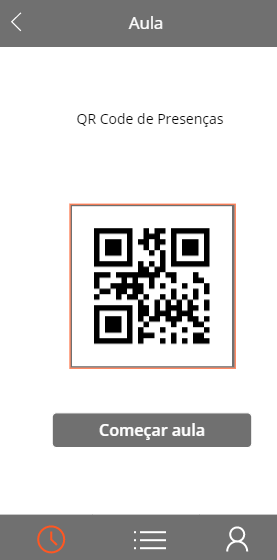
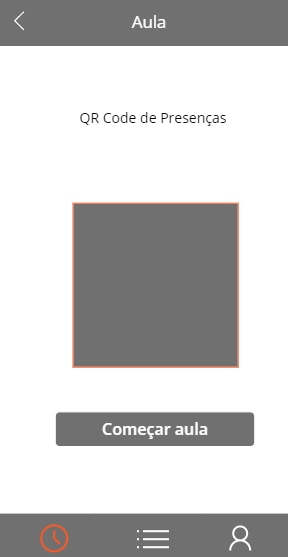
 

Figura – Começar uma aula

* Aluno, este ao clicar em “Analisar Qr code” é lhe aberto o scanner de qr codes, e ao analisar o qr code da aula, são verificadas as seguintes condições:
  + A sua localização: Verifica se este está dentro de uma certa área do IPV (por causa do Covid-19, neste momento a aplicação verifica se o aluno está dentro da minha vizinhança)
  + Se a aula ainda não terminou: Vai ser verificado se a aula ainda está a ser lecionada, verificando se o atributo da aula “Is\_active” está a 1.
  + Se a sua falta desta aula existe: Se existir, elimina a dita falta, caso contrário devolve uma mensagem de erro.

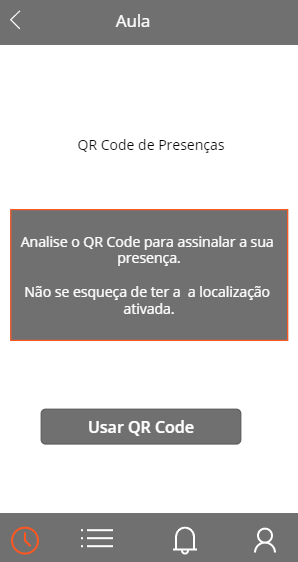


Figura - Analisar qr code

Existem certas ações neste screen que são feitas automaticamente através do Power Automate, que podem ser consultados nos anexos, tais como:

A aula quando acaba, fica inativada através da mudança do valor boolean “Is\_active” – figura 17

O aviso de começo de aula 30 minutos antes – figura 18

Avisar o aluno que começou a aula – figura 19

Avisar o aluno que faltou à aula – figura 20

### **Disciplinas**

Este screen mostra todas as disciplinas a que o utilizador que fez login está associado.

Caso seja um professor, são as disciplinas que este leciona, caso seja um aluno são as disciplinas que o aluno está inscrito.

Este *screen* tem também um *searchBar* que server para procurar uma disciplina em concreto mais rapidamente e um botão caso o utilizador se queira inscrever a outras disciplinas.

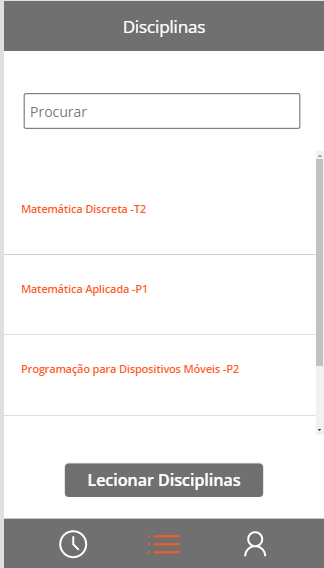


Figura - Disciplinas associadas (lado do professor e lado do aluno)

Ao clicar em “Lecionar Disciplinas” o professor será apresentado com todas as disciplinas existentes que não estejam a ser lecionadas por outro professor.

Ao clicar numa dessas disciplinas, é lhe perguntado se quer lecionar essa disciplina.

Ao clicar em “Lecionar” este é direcionado para as suas disciplinas onde consegue encontrar a sua nova disciplina.

No lado do aluno, o processo é o mesmo, só que com menos restrições, em que lhe são apresentadas todas as disciplinas em que não está inscrito.

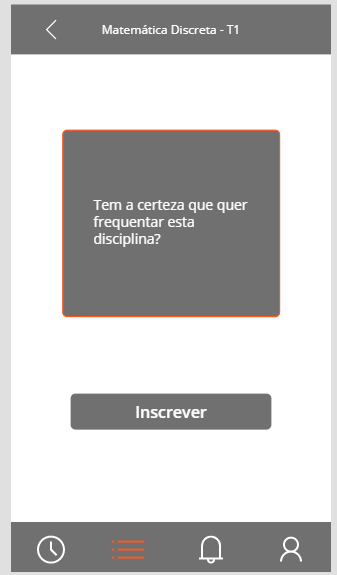
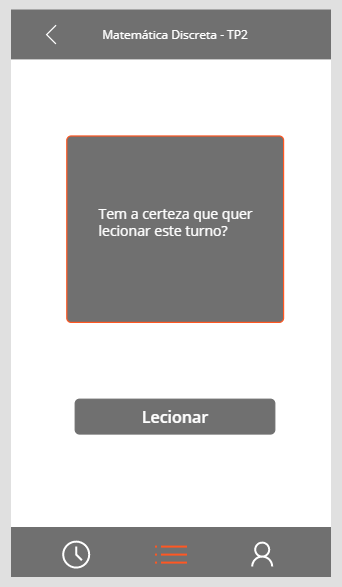


Figura - Inscrições de disciplinas

Nas disciplinas inscritas, o utilizador ao clicar numa delas, acede ao perfil da dita disciplina e às suas informações.

No caso do Professor o Perfil é assim:

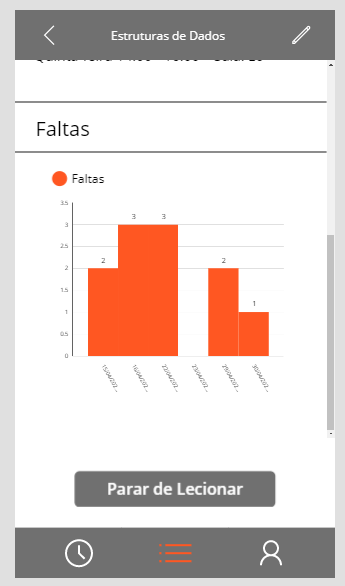
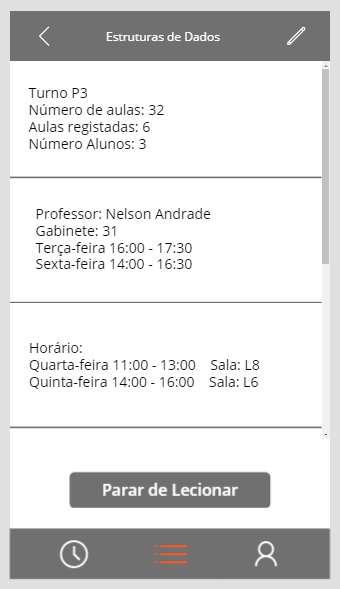


Figura - Perfil da disciplina (print do lado esquerdo com scroll para baixo)

As informações são as seguintes: turno, número de aulas para dar até ao final do semestre, aulas dadas até ao momento, alunos inscritos neste turno, professor que leciona este turno, gabinete do professor e horário de atendimento do mesmo, horário de aulas e as suas salas e um gráfico com o número de faltas por aulas lecionadas.

Ao clicar no número de alunos inscritos no turno, consegue-se ver os alunos inscritos no turno (nome e email) e ao clicar em cada um é possível ver o seu registo de faltas no turno.

Ao clicar no gráfico este é expandido e é possível consultar quem faltou a uma certa aula e remover alguma falta de um aluno.

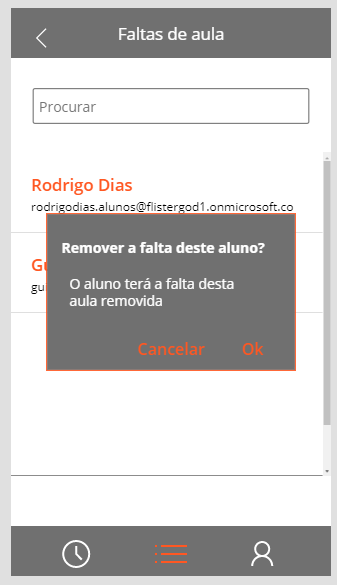
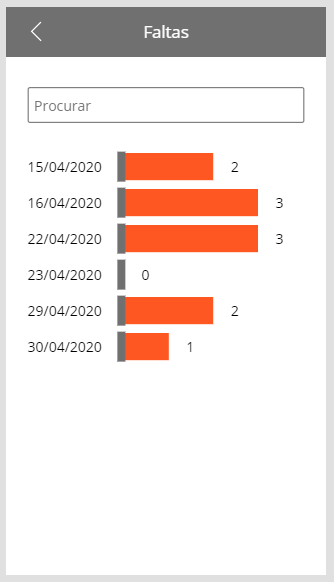
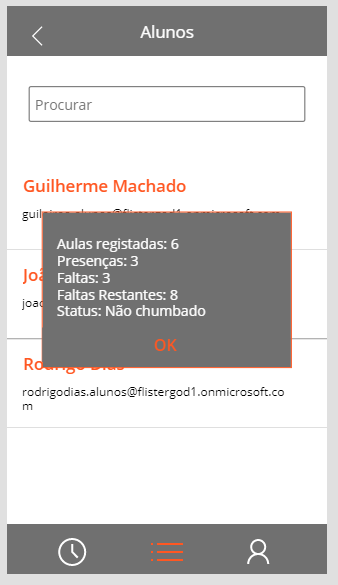


Figura 21 – Remover a falta de um aluno numa aula

Figura 20 – Aulas lecionadas de um turno

Figura 19 - Registo de faltas de um aluno inscrito no turno

O Professor consegue editar alguns dados da disciplina, como a sala de aula, gabinete ou horário de atendimento. Ao guardar as alterações feitas ao perfil da disciplina, os alunos inscritos na mesma, são notificados sobre essas alterações.

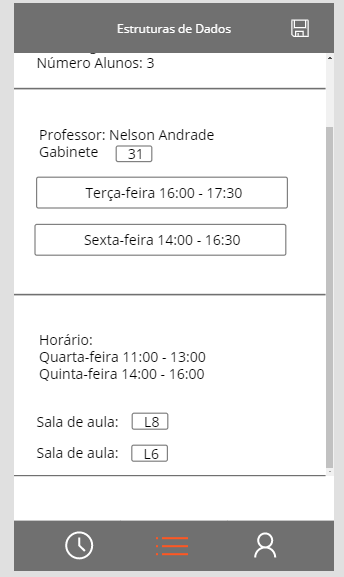
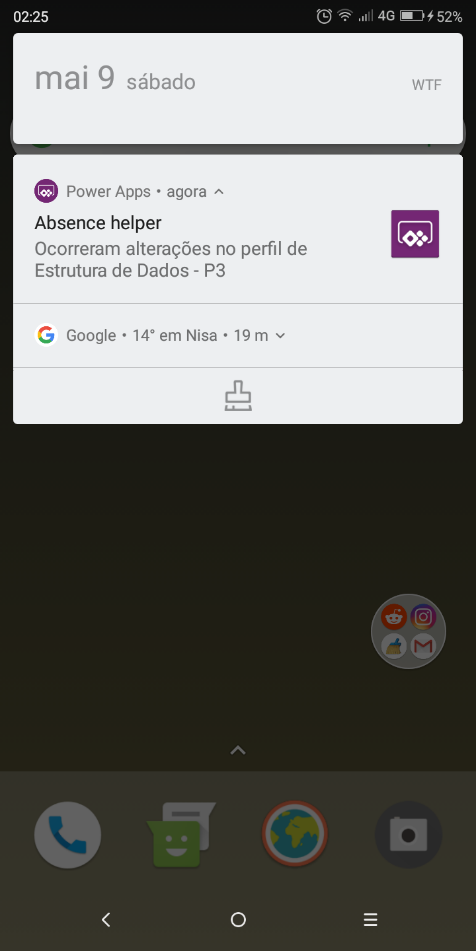


Figura 22 - Alteração de dados de perfil da disciplina

No lado do Aluno, o perfil é parecido, mas com informações centradas no próprio aluno.

Aqui o aluno pode desativar as diferentes notificações que pode receber da disciplina.

A única que este não pode desativar, é caso o professor edite o perfil da disciplina, pois a sua ocorrência é menor.

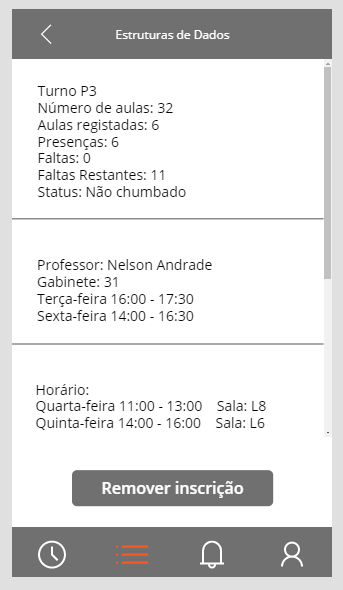
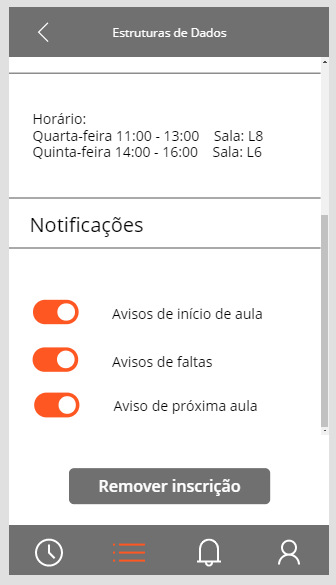


Figura 23 – Perfil da disciplina vista pelo aluno

### **Notificações**

Neste screen são apresentadas as notificações pop-up que o aluno alguma vez recebeu.

Podemos pensar neste screen como um histórico de notificações.

Estas são ordenadas da mais recente para a menos recente e são de cor branca caso o aluno já tenha clicado nestas.

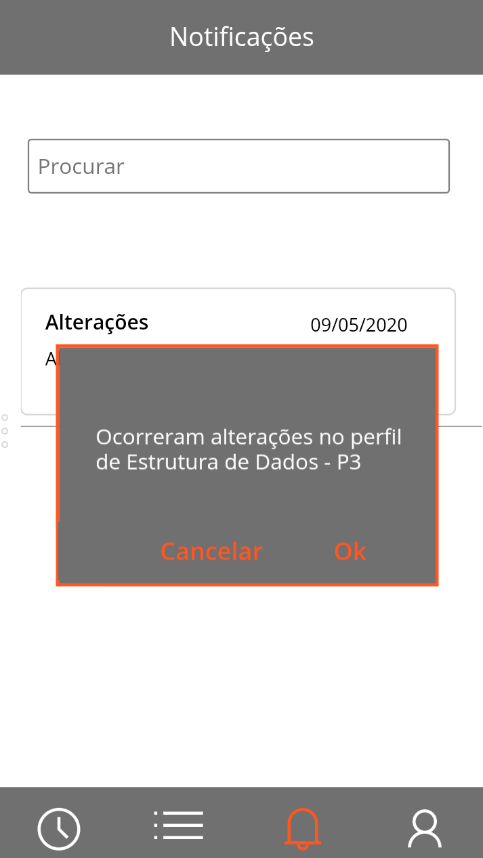
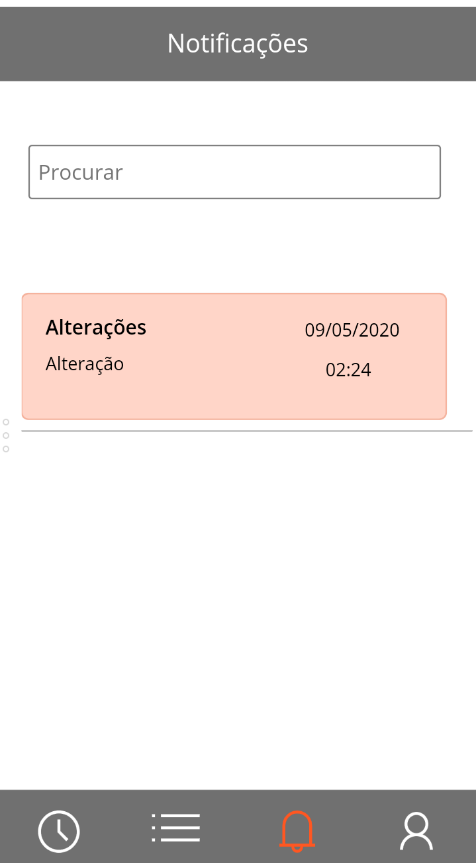
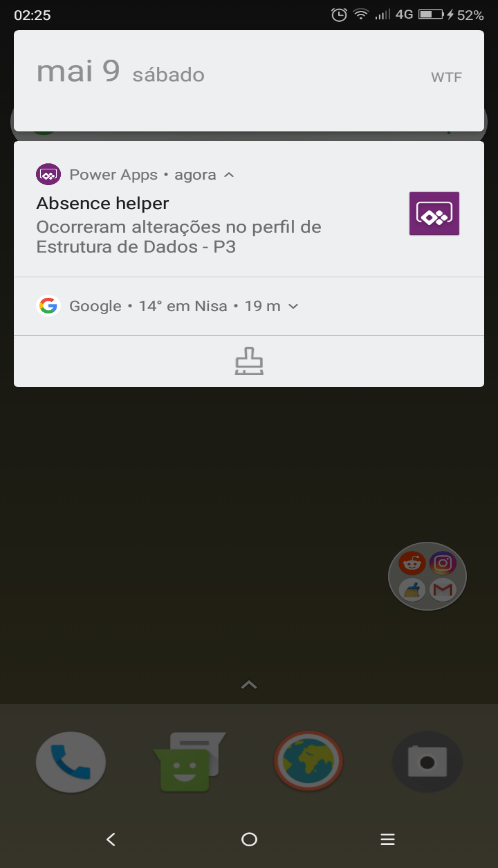


Figura 24 – Registo de notificações

### **Perfil**

O Perfil é um *screen* onde o utilizador pode consultar alguma das suas informações.

A imagem do utilizador não aparece, pois é uma funcionalidade para contas Premium.

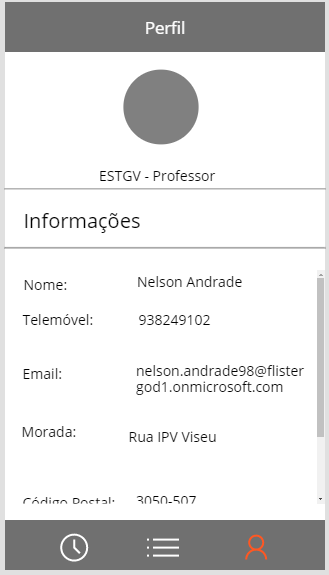


Figura 25 - Perfil do utilizador

# **Setup Experimental e Testes**

Neste capítulo será descrita a fase de Setup Experimental e Testes, uma das mais importantes no desenvolvimento de software. Este é um processo que tem o objetivo de verificar se o projeto desenvolvido executa as funcionalidades para as quais foi destinado, em que são validados tantos os requisitos funcionais, como os não funcionais

## **Setup Experimental**

Tendo em consideração que esta aplicação irá ser utilizada num ambiente escolar e por dispositivos móveis foram utilizados os seguintes dispositivos para o Setup Experimental: um simulador na plataforma PowerApps que corre no browser Google Chrome que está presente num Asus 2,6 GHz Intel Core i7-9750H com 6 núcleos e 12 GB RAM, um Lenovo K5 Play 32GB ROM e 3GB RAM, e Huawei Y5 2018 2GB RAM e 16 GB ROM.

## **Testes**

De entre os muitos testes possíveis, foram tidos em conta os mais relevantes, de acordo com as necessidades do projeto. No final, foram escolhidas as abordagens que correspondem aos subtítulos deste subcapítulo.

### **Testes de Desempenho**

Para testar a aplicação foi utilizada uma ferramenta em versão beta do PowerApps que se designa por Test Studio. Aqui é possível simular várias

### **Testes de Usabilidade**

Os testes de usabilidade, sendo realizados por utilizadores reais, fornecem informações muito importantes acerca do funcionamento do produto final. O atributo de qualidade Usabilidade tem uma prioridade alta e como tal, no decorrer do desenvolvimento da aplicação foram seguidas as 10 Heurísticas de Nielsen. Uma avaliação heurística é um método informal, onde várias pessoas ou avaliadores, são questionados acerca de vários aspetos no design de uma interface.

De forma a satisfazer as 10 Heurísticas apresentadas de seguida foram tidos em conta as seguintes soluções:

Tabela - Heurísticas de Nielsen

|  |  |  |
| --- | --- | --- |
| Heurísticas de Nielsen | | |
| Número | **Heurística** | **Solução Adotada** |
| 1 | Visibilidade do estado do sistema | Na aplicação, a barra inferior, ou navbar, apresenta a laranja o ícone da secção onde o utilizador se encontra. |
| 2 | Correspondência entre o sistema e o mundo real | Foram escolhidos botões intuitivos, e apresentada a informação de forma lógica e natural. Os submenus estão devidamente identificados com texto sugestivo |
| 3 | Controlo do utilizador e liberdade | Para fornecer ao utilizador mecanismos de escape em caso de erro, este poderá sempre utilizar a opção “Retroceder” e “Avançar”. |
| 4 | Consistência e padrões | Todos os menus e botões são utilizados na aplicação da mesma forma, para que o utilizador quando já tenha algum conhecimento olhe e intuitivamente perceba de imediato a sua ação. |
| 5 | Prevenção de erros | Mensagens de erro ou de informação são mostradas ao utilizador assim que este realize uma operação que envolva outros processos. Assim, oferecendo feedback ao utilizador faz com que este esteja constantemente informado acerca das ações que realiza. |
| 6 | Reconhecimento em vez de lembrar | Para minimizar a quantidade de informação a ser memorizada pelo utilizador, a aplicação em cada ecrã fornece botões com skeuomorfismo para serem facilmente reconhecimento |
| 7 | Flexibilidade e eficiência de uso | Utilizadores experientes podem tirar partido de “aceleradores” para poder realizar as suas ações de forma mais rápida. Um exemplo pode ser a pesquisa de uma disciplina a que este está inscrito em específico em vez da visualização de todos as disciplinas a que este está inscrito |
| 8 | Design estético e minimalista | O design escolhido é uniforme, ou seja, a maior parte das diferentes secções seguem o mesmo padrão, sendo este um padrão de listas |
| 9 | Reconhecimento e recuperação de erros | As mensagens de erro são fornecidas em plain text, para que não haja equívocos por parte do utilizador |
| 10 | Ajuda e documentação | Todas as secções mostram só o necessário e é bastante fácil retirar destas a informação que o utilizador necessita. No entanto existe também um manual de utilizador |

# **Conclusões e Trabalho Futuro**

Este capítulo visa a explicar, em pormenor, toda a parte do projeto que

## **Conclusões**

As indicações apresentadas pretendem contribuir para uma melhor compreensão sobre o funcionamento das nossas Aplicações para efetuar este nosso Projeto.

Pretende-se com este Relatório que o leitor fique com uma noção básica de todos os temas abordados no mesmo.

Relativamente aos protótipos feitos, deram imensa ajuda e facilitaram o desenvolvimento e codificação.

Assim sendo, como trabalho futuro sugere-se o seguinte conjunto de iniciativas:

* Melhorar a localização do utilizador incluindo à localização do gps, determinar se o utilizador está ligado a uma certa rede (wifi da sala de aula).
* Implementar a criação de aulas não planeadas, ou seja, o professor consegue agendar aulas especiais (exemplo: para compensação de feriados) com horários, sala e dia fora do normal. Ao criar essa aula especial, o aluno poderia receber uma notificação sobre a criação dessa aula e este receberia as notificações normais sobre uma aula, mas relativas a esta aula especial.

Resta-nos, pois, esperar que este documento seja útil e contribua positivamente para o conhecimento do leitor.

# **Referências**

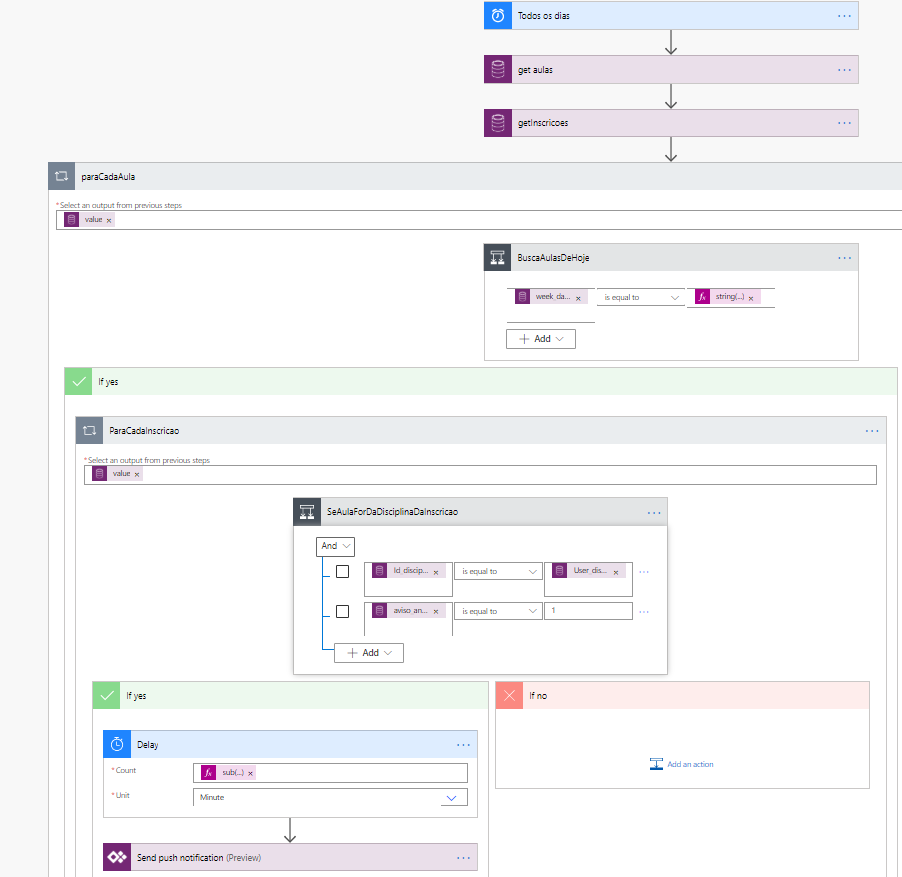
# **Bibliografia**

# **Anexos**

## **Flows**



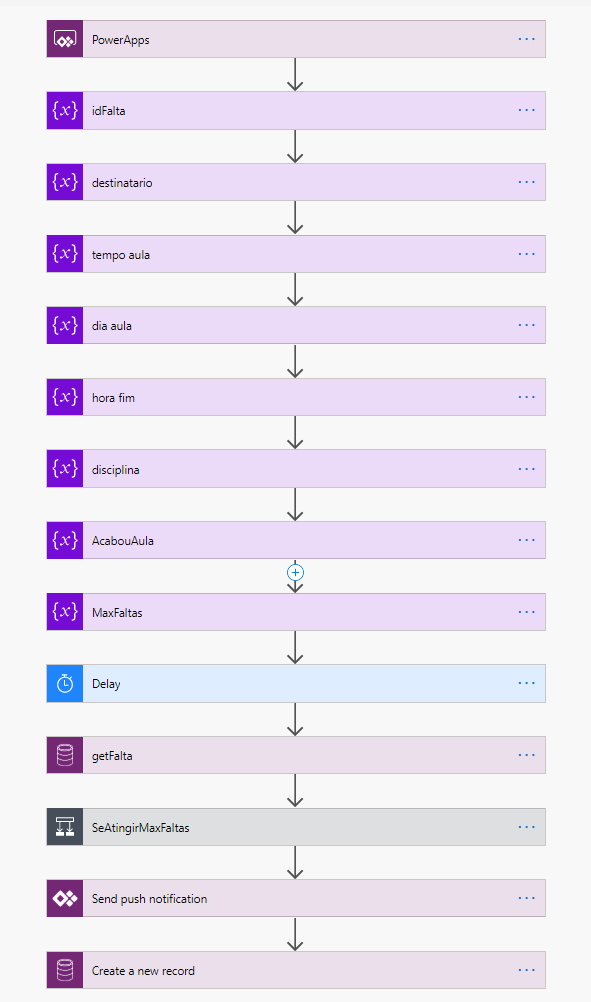
A aula quando acaba, fica inativada através da mudança do valor boolean “Is\_active”



O aviso de começo de aula 30 minutos antes



Avisar o aluno que começou a aula



Avisar o aluno que faltou à aula

## **Casos de Uso**

## **Manual de utilização**