

**Mestrado em Engenharia Informática**

PROJETO UNIDADE CURRICULAR PROGRAMAÇÃO DE DISPOSITIVOS MÓVEIS

**Trabalho Prático**

Ricardo Abreu Gomes

Barcelos, julho, de 2019

Índice

[1 Resumo 3](#_Toc14747332)

[2 Introdução 3](#_Toc14747333)

[3 Arquitetura da solução 4](#_Toc14747334)

[4 Modelo de dados 5](#_Toc14747335)

[5 Desenvolvimento da aplicação 6](#_Toc14747336)

[1 Conclusão 7](#_Toc14747337)

# Resumo

# Este documento aborda o desenvolvimento de uma aplicação capaz de ajudar um professor a planear as suas aulas, cursos e afins. Em que inicialmente foi realizada uma pequena introdução ao problema e são definidos os objetivos da aplicação, de seguida foi apresentada a arquitetura utilizada para resolver o problema, o modelo de dados utilizado bem como alguns detalhes de desenvolvimento e por fim foi realizada uma conclusão.

# Introdução

Um bom professor planea as suas aulas e os seus cursos antecipadamente, normalmente ferramentas como o Excel são utilizadas para esse efeito, contudo por vezes alguns detalhes escapam o que é característico nesta ferramenta.

Este trabalho tem como objetivo a realização de uma aplicação móvel capaz de ajudar professores do ensino superior no planeamento das suas aulas e cursos. Isto possibilita a criação de aulas, planear o seu conteúdo bem como decidir momentos de avaliação e preparar um curso mais realista para os estudantes.

# Arquitetura da solução

Para a realização deste trabalho foi utilizado o Android Studio juntamente com uma base de dados o Realm para o armazenamento dos dados da aplicação, uma vez que esta base de dados ainda não tinha sido explorada para além das aulas e pareceu ser interessante pois para além disso esta base de dados é não relacional e foi a nossa primeira experiência com uma base de dados deste tipo.



Figura 1 –Arquitetura

# Modelo de dados

Apesar do Realm ser uma base não relacional é foi possível estabelecer relações compondo várias classes. Assim após compreendido o problema chegamos ao seguinte esquema de dados (Ver Figura.2).

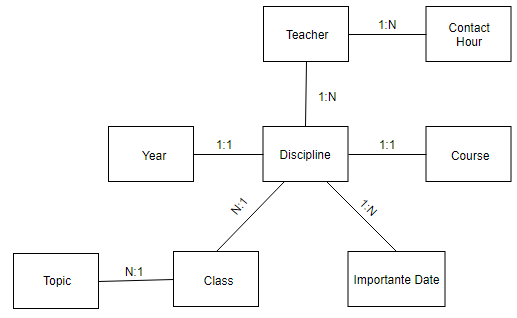


Figura 2 –Modelo de dados para o problema

Em que cada professor tem um horário de horas em que pode ser contactado, este contacto é representado por um dia da semana a hora de início e de fim bem como o tipo de contacto realizado que neste caso pode ser físico ou de forma remota através de uma vídeo chamada por exemplo.

Cada professor leciona várias disciplinas, em que cada disciplina é composta por um nome, um acrónimo, um período, um ano, um curso, várias aulas e várias datas importantes.

Cada aula é composta por uma sala, a data em que vai ser realizada, a hora que começa, a sua duração, se é especial ou não por exemplo aulas para a realização de testes e por vários tópicos.

# Desenvolvimento da aplicação

Inicialmente foi desenvolvida a atividade destinada para o registo de cada professor em que após o registo este estava apto para realizar o login na aplicação, os dados do registo são compostos por um nome e uma password em que ambos não podem ser nulos e compreendidos entre 3-8 caracteres, para além disso o nome do utilizador não estar em uso.

De seguida foi realizada uma atividade designada menu em que esta é composta apenas por botões que dão acesso a todas as funcionalidades da aplicação, como enviar emails e realizar operações CRUD nos cursos, disciplinas, aulas, anos, contactos, tópicos e aulas importantes.

Para as operações de criação foi necessário a implementação de *views* capazes de recolher os dados necessários como por exemplo *spinner*, calendar e texto *views*. Neste tipo de operações foram também realizadas várias verificações antes de inserir o objeto na base de dados, como por exemplo verificar se esse objeto já existe, verificar se todos os campos estão corretos e se os obrigatórios estão preenchidos, no caso de inserção de números, verificar se é composto apenas por números.

Seguidamente para as operações de leitura foram utilizados *recycler views* bem como *list viwes*, em que cada uma destas implementava métodos de *onClick* responsável por a seleção do item desejado para realizar outro tipo de operações como remoção ou atualização.

Por fim para o envio de emails foi utilizado um *intent* implícito em que a aplicação recolhia os dados necessários para o preenchimento do email como o destinatário e a mensagem, de seguida o utilizador só precisava de escolher a aplicação para enviar o email.

# Conclusão

Com a realização deste trabalho foi possível consolidar os conteúdos dados nas aulas bem como explorar outras ferramentas do *Android Studio*.