

Relatório do Projeto Mobile

Realizado por:

- Bruno Gouveia Nº 20240181
- Paulo Alberto Nº 20240873
- Ngweza Van-Dúnem Nº 20240075
- Mateus Reis Nº20241799

1. Identificação

- **Universidade:** Universidade Europeia
- **Faculdade:** IADE
- **Nome do Projeto:** FazTudo - Plataforma de Serviços
- **Repositório GitHub:** [FazTudo](#)

2. Breve Descrição da App e do Problema

O FazTudo é uma aplicação móvel de serviços, desenvolvida para a plataforma Android (Kotlin/Java), que funciona como uma ponte digital entre clientes e prestadores de serviços independentes. O principal problema que visa resolver é a dificuldade que muitos prestadores de serviços independentes têm em promover o seu trabalho e alcançar potenciais clientes.

A solução proposta permite:

- **Clientes:** Contratar serviços de forma rápida, pesquisar prestadores por categoria e localização, e efetuar pagamentos.
- **Prestadores (Vendedores):** Criar um perfil para oferecer os seus serviços e gerir pedidos.

3. Descrição dos Objetivos e da Motivação

Objetivos:

- Criar uma aplicação intuitiva para clientes e prestadores de serviços.
- Garantir processos de pagamento seguros e transparentes (objetivo futuro).
- Oferecer funcionalidades rápidas de pesquisa por proximidade e avaliação de serviços.

Contexto e Motivação:

A motivação reside na criação de uma plataforma que facilite a transação entre quem procura um serviço e quem o oferece, simplificando a experiência e focando-se num mercado local.

4. Identificação do Público-Alvo

- **Clientes:** Pessoas que procuram serviços rápidos, especialmente em tarefas domésticas, reparos ou serviços pessoais.
- **Prestadores de Serviços (Vendedores):** Profissionais independentes ou *freelancers* que procuram um canal de promoção e gestão de clientes.

5. Pesquisa de Mercado sobre Aplicações Semelhantes

Foram analisadas plataformas de serviços digitais já estabelecidas, como Fiverr (foco em *freelancing*) e Oscar App (foco em tarefas domésticas e reparos).

7. Descrição da Solução Implementada

i. Descrição Genérica da Solução Implementada

A solução consiste numa aplicação móvel Android (Frontend) comunicando com um Backend através de uma API RESTful, que utiliza a Base de Dados relacional FazTudoDB (MySQL) para persistência de dados. A autenticação é baseada em JWT (JSON Web Tokens).

ii. Enquadramento nas Diversas Unidades Curriculares

- **Base de dados:** Ajuda na implementação da base de dados e o seu correto funcionamento.
- **Competências Comunicacionais:** Ajuda tanto como fazer uma apresentação em termos de acessório (PowerPoint) na sua estrutura visual e no conteúdo da mesma.
- **Matemática Discreta:** Apesar da sua à primeira vista não ajudar em termos do desenvolvimento frontend e backend, têm a sua importância através da ajuda e como funcionam as bases de dados pois são através de regras da matemática discreta (Teorema de conjuntos e Operações Lógicas).

- **Programação de Dispositivos Móveis:** Ajuda no desenvolvimento do frontend como na implementação de features.
- **Programação Orientada a Objetos:** Ajuda no desenvolvimento do backend como na implementação de features.
- **Projeto de Desenvolvimento Móvel:** Ajuda e feedback do progresso da app.

iii. Arquitetura da Solução

Arquitetura Client-Server de três camadas:

- Apresentação (Frontend): Aplicação Android (Kotlin/Java).
- Lógica de Negócio (Backend/API): API RESTful (Java/Spring Boot)
- Dados (Base de Dados): MySQL(FazTudoDB).

iv. Tecnologias Utilizadas

- Frontend: Android (Kotlin/Java).
- Backend/API: Java/Spring Boot .
- Base de Dados: MySQL.
- Design: Figma.
- Controlo de Versões: Git/GitHub.

v. Versão Atualizada do Dicionário de Dados

Tabela	Descrição
localizacao	Armazena morada, cidade e código postal dos utilizadores e vendedores.
estado	Define o estado de um serviço (ativo, pendente, concluído).
categoria	Classificação dos serviços disponíveis.
user	Representa os utilizadores (clientes) da aplicação.
vendedor	Representa os prestadores de serviço, podendo ter ligação 1:1 a um utilizador.
vendedor_categoria	Liga vendedores a categorias de serviço específicas.
servico	Serviços oferecidos pelos vendedores a utilizadores.
avaliacao	Avaliações e comentários realizados pelos utilizadores.

Tabela: avaliacao

Campo	Tipo	Chave	Descrição
id_avaliacao	INT	PK	Identificador
id_user	INT	FK	FK → user
id_vendedor_categoria	INT	FK	FK → vendedor_categoria
nota	TINYINT		Pontuação (1 a 5)
comentario	TEXT		Comentário textual
data_avaliacao	DATETIME		Data e hora da avaliação

Tabela: servico

Campo	Tipo	Chave	Descrição
id_servico	INT	PK	Identificador
titulo	VARCHAR(150)		Título do serviço
preco	DECIMAL(10,2)		Preço
data_publicacao	DATETIME		Data de publicação
id_vendedor_categoria	INT	FK	FK → vendedor_categoria
id_user	INT	FK	FK → user
id_estado	INT	FK	FK → estado

Tabela: vendedor_categoria

Campo	Tipo	Chave	Descrição
id_vendedor_categoria	INT	PK	Identificador
id_categoria	INT	FK	FK → categoria
id_vendedor	INT	FK	FK → vendedor
descricao	VARCHAR(255)		Descrição dos serviços
data_registro	DATETIME		Data de registo

Tabela: vendedor

Campo	Tipo	Chave	Descrição
id_vendedor	INT	PK	Identificador do vendedor
nome	VARCHAR(100)		Nome completo
email	VARCHAR(150)	UNIQUE	Email
telemovel	VARCHAR(20)		Contacto
data_registro	DATETIME		Data de registo
id_localizacao	INT	FK	FK → localizacao
id_user	INT	FK/UNIQUE	Ligaçao 1:1 opcional com user

Tabela: user

Campo	Tipo	Chave	Descrição
id_user	INT	PK	Identificador do utilizador
nome	VARCHAR(100)		Nome completo
email	VARCHAR(150)	UNIQUE	Email
palavra_passe	VARCHAR(255)		Palavra-passe cifrada
telemovel	VARCHAR(20)		Contacto
data_registro	DATETIME		Data de registo
id_localizacao	INT	FK	FK → localizacao

Tabela: categoria

Campo	Tipo	Chave	Descrição
id_categoria	INT	PK	Identificador
nome	VARCHAR(100)		Nome da categoria

Tabela: estado

Campo	Tipo	Chave	Descrição
id_estado	INT	PK	Identificador
nome	VARCHAR(50)		Nome do estado

Tabela: localizacao

Campo	Tipo	Chave	Descrição
id_localizacao	INT	PK	Identificador único
cidade	VARCHAR(80)		Cidade
morada	VARCHAR(150)		Morada completa
codigo_postal	VARCHAR(15)		Código postal
latitude	DOUBLE		
longitude	DOUBLE		

vi. Versão Atualizada do Guia de Dados

Tabela	Exemplo de Dados
localizacao	(1, "Lisboa", "Rua Augusta, 100", "1000-000", 1100-053, 38.7103, -9.1367)
estado	(1, "Ativo"), (2, "Concluído"), (3, "Cancelado")
categoria	(1, "Canalização"), (2, "Limpeza"), (3, "Pintura")
user	(1, "Maria Silva", " maria@email.com ", "****", "912345678", "2025-01-01", 1)
vendedor	(1, "João Santos", " joao@email.com ", "913456789", "2025-01-03", 1, 1)
vendedor_categoria	(1, 1, 1, "Serviços de canalização residencial", "2025-01-04")
servico	(1, "Reparação de canos", 50.00, "2025-02-01", 1, 1, 1)
avaliacao	(1, 1, 1, 5, "Excelente serviço!", "2025-02-05")

vii. Link para uma versão atualizada da documentação REST e Manual do Utilizador

Ambos estão no repositório do GITHUB, [FazTudo](#)

9. Autoavaliação da Implementação do Projeto

Apesar de o backend possuir funcionalidades completas para gestão de serviços, a implementação no frontend apresenta limitações significativas que afetam a experiência de utilizadores e vendedores, sendo o problema principal apenas o

utilizador (user) conseguir cancelar pedidos através da interface, os vendedores não têm acesso equivalente a esta funcionalidade.

Existe uma aba para visualizar serviços do utilizador, também existe uma aba para visualizar serviços do vendedor. Os pedidos efetuados não aparecem do lado do vendedor, o que impossibilita:

- Visualização de novos pedidos
- Aceitação de pedidos
- Cancelamento de pedidos pelo vendedor
- Gestão adequada do fluxo de trabalho

Funcionalidades Não Implementadas:

Embora o backend suporte estas operações, o frontend não implementa totalmente:

- Avaliação de serviços
- Interface incompleta ou ausente
- Aceitação de pedidos pelo vendedor
- Cancelamento pelo vendedor

Para Utilizadores:

- Não recebem feedback visual adequado sobre o estado dos pedidos
- Não conseguem avaliar serviços mesmo após conclusão

Para Vendedores:

- Não visualizam pedidos recebidos
- Não podem aceitar/rejeitar pedidos
- Não conseguem gerir o seu pipeline de trabalho
- Impossibilitados de cancelar pedidos quando necessário

Outra função que queríamos implementar, porém nem no backend nem no frontend seria o pagamento através da app de forma segura e um sistema de chat entre o user e o vendedor.

10. Bibliografia

- Fiverr: <https://www.fiverr.com>
- Oscar App: <https://oscarapp.io>

