



INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

LICENCIATURA EM ENGENHARIA INFORMÁTICA E COMPUTADORES

KeepMyPlanet: Sistema Kotlin Multiplatform para Limpeza Ambiental

Diogo Ribeiro

Telefone: +351 911 889 669

E-mail: a47207@alunos.isel.pt

Rafael Pegacho

Telefone: +351 913 464 971

E-mail: a49423@alunos.isel.pt

Orientador

Eng. Pedro Pereira

E-mail: pedro.pereira@isel.pt

10 de março de 2025

1 Introdução

1.1 Contextualização

Portugal e o mundo enfrentam um desafio ambiental significativo no que diz respeito à gestão de resíduos e poluição. De acordo com a Agência Portuguesa do Ambiente, no ano de 2023, a produção de resíduos urbanos em Portugal foi de 5,3 milhões de toneladas, registando um aumento de 0,28% em relação ao ano anterior [1]. Ainda mais alarmante é o facto de que apenas 32% destes resíduos são preparados para reutilização e reciclagem, valor consideravelmente inferior à meta europeia de 55% estabelecida para 2025 [1].

A nível mundial, são geradas anualmente cerca de 2,24 mil milhões de toneladas de resíduos urbanos com projeções de aumento em 73%, estimando-se um total de 3,88 mil milhões até 2050 [2]. A poluição residual não afeta apenas as áreas onde são depositadas, mas também contamina espaços naturais, urbanos e zonas costeiras. Nas zonas costeiras, aproximadamente 80% do lixo marinho é composto por plásticos [3].

Apesar da crescente consciencialização ambiental e do interesse da população em participar em ações de limpeza, existe uma lacuna significativa nas ferramentas disponíveis para identificar zonas poluídas e coordenar esforços de limpeza de forma eficaz. Atualmente, a organização de iniciativas de limpeza ambiental enfrenta desafios como a dispersão de informação, a dificuldade de coordenação entre voluntários e a falta de mecanismos para monitorizar o progresso das ações realizadas.

1.2 Solução

É neste contexto que surge o ***KeepMyPlanet***, um sistema multiplataforma desenvolvido com *Kotlin Multiplatform (KMP)* [4] que permite identificar e mapear zonas poluídas, bem como organizar e participar em eventos comunitários de limpeza. O sistema proporciona uma interface interativa onde os voluntários podem assinalar áreas como poluídas, partilhar fotografias e descrições das condições encontradas, e criar ou aderir a iniciativas de limpeza organizadas para essas zonas. ***KeepMyPlanet*** é o mote de convergência para a ação comunitária e ambiental, abordando um problema real cada vez mais relevante e emergente.

2 Análise

O sistema será concebido com um conjunto de funcionalidades destinadas a facilitar a identificação, monitorização e coordenação das ações dos utilizadores voluntários, assegurando privacidade de dados e uma boa acessibilidade.

2.1 Requisitos e Funcionalidades

2.1.1 Requisitos Funcionais

Gestão de Utilizadores e Permissões - O sistema deve disponibilizar funcionalidades de registo e autenticação de utilizadores. Deve também ser capaz de diferenciar as permissões e funcionalidades de acordo com os diferentes perfis de utilizador, como *Guest*, *User*, *Organizer* e *Admin*.

Identificação e Sinalização de Zonas Poluídas - Deve ser possível que os utilizadores registados sinalizem áreas poluídas num mapa interativo, com a possibilidade de anexar fotografias e descrições detalhadas sobre o estado da área identificada.

Gestão de Eventos - O sistema deve permitir que utilizadores registados possam criar, editar e excluir eventos de limpeza. Cada evento está associado a uma zona previamente assinalada no mapa e corresponde a uma data e hora onde os seus participantes se unem para a limpeza dessa mesma zona. Deve ainda possibilitar a inscrição dos utilizadores nos eventos e a confirmação da sua participação. Para facilitar o controlo da presença, será implementado um sistema de validação, por exemplo, através da leitura de código QR, fornecido pelos organizadores, e lido pelos voluntários. O sistema também deverá disponibilizar estatísticas sobre o impacto de cada evento.

Comunicação e Colaboração - Será implementado um sistema de mensagens, associado a cada evento, para permitir a comunicação entre os participantes do mesmo. Além disso, os organizadores terão a capacidade de enviar atualizações e notificações para os utilizadores inscritos nos seus eventos, garantindo uma comunicação eficaz e atualizada.

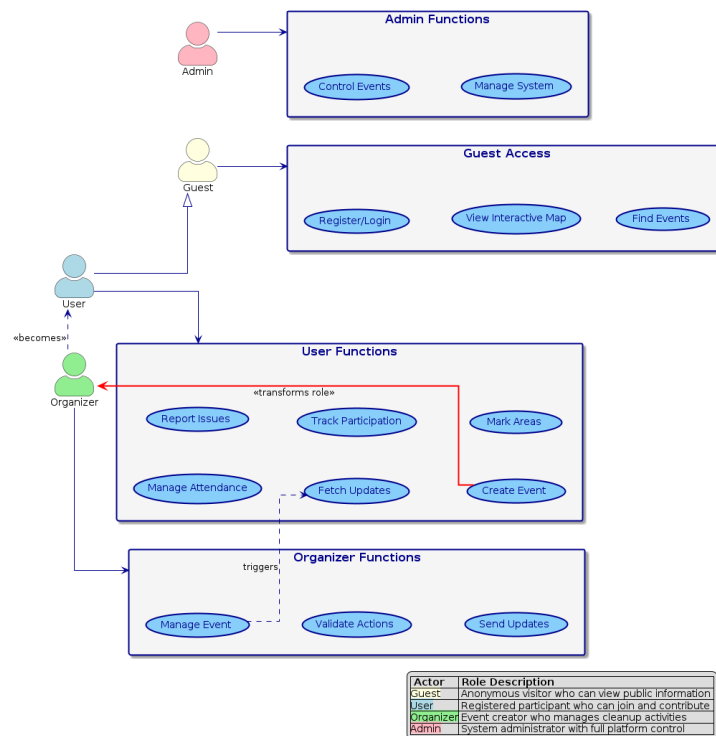


Figura 1: Diagrama *Use Case* do sistema

2.1.2 Requisitos Não Funcionais

Maximizar o código partilhado usando KMP - O sistema deve ser desenvolvido aproveitando ao máximo das capacidades do **Kotlin Multiplatform**, visando a partilha extensiva de código entre as diferentes plataformas suportadas. Esta abordagem é crucial para garantir consistência funcional, reduzir significativamente o tempo de desenvolvimento e facilitar a manutenção do sistema a longo prazo. A lógica da aplicação, gestão de estado, comunicação com APIs e persistência de dados devem ser implementadas na camada comum, permitindo que apenas os componentes específicos de UI de cada plataforma sejam desenvolvidos separadamente, promovendo assim uma maior eficiência de recursos na experiência do utilizador em todos os dispositivos.

Desempenho e Escalabilidade - O sistema deve ser capaz de suportar um grande número de utilizadores simultâneos sem comprometer a experiência de utilização. As operações críticas,

como a sinalização de zonas no mapa e a inscrição em eventos, não podem ser demoradas, garantindo que o sistema se mantenha eficiente mesmo em cenários de elevada afluência.

Segurança e Privacidade - A autenticação deve ser implementada de forma segura, garantindo que o acesso às funcionalidades exclusivas dos utilizadores registados seja protegido. Além disso, a proteção de dados dos utilizadores deve ser assegurada em conformidade com o Regulamento Geral sobre a Proteção de Dados, para garantir a privacidade e segurança das informações pessoais.

Acessibilidade - A interface do sistema deve ser intuitiva e fácil de utilizar, garantindo que diferentes perfis de utilizadores possam adotar a plataforma com facilidade. O sistema também deve ser compatível com diferentes dispositivos.

2.2 Tecnologia

O sistema será implementado utilizando **Kotlin Multiplatform (KMP)** [4], uma tecnologia que possibilita a partilha de código entre várias plataformas, com foco na redução da duplicação de código facilitando a manutenção do sistema e a escrita de código. A arquitetura do sistema será baseada no padrão **Model-View-Viewmodel (MVVM)**, permitindo uma separação clara entre a interface do utilizador e a lógica do sistema, promovendo a escalabilidade e a manutenção do código, para além de melhorar a testabilidade. A integração com uma API externa de mapas representa um dos maiores desafios do projeto. Esta API será responsável por permitir a sinalização de zonas poluídas no mapa interativo, sendo essencial que a comunicação entre a aplicação e o serviço externo seja eficiente e fiável. Para efetuar essa integração, será usada a **framework Ktor** [5], especializada na realização de pedidos HTTP. Será responsável por garantir que a comunicação com a API é efetuada de forma assíncrona, otimizada e segura, garantindo uma correta gestão de dados e erros. Esta abordagem assegura uma interação fluida com o serviço de mapas, fundamental para o desempenho global da aplicação.

3 Planeamento

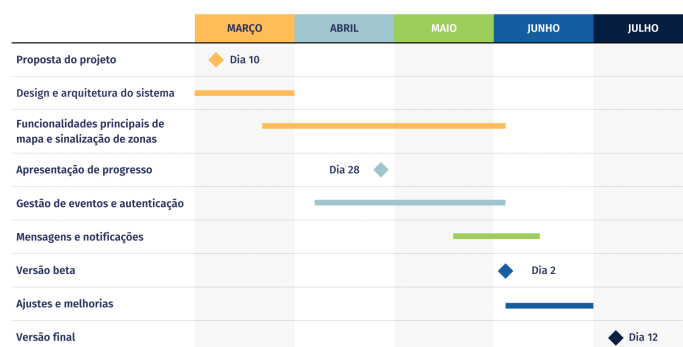


Figura 2: Planeamento do projeto

4 Referências

- [1] Agência Portuguesa do Ambiente (APA). (2023). Relatório Anual de Resíduos Urbanos 2023. Disponível em: https://apambiente.pt/sites/default/files/_Residuos/Producao_Gest%C3%A3o_Residuos/Dados%20RU/2023/raru_2023.pdf Acedido em 10/03/2025.

- [2] World Bank Group. (2022). Solid Waste Management. Disponível em: <https://www.worldbank.org/en/topic/urbandevelopment/brief/solid-waste-management> Acedido em 10/03/2025.
- [3] International Union for Conservation of Nature (IUCN). (2024). Plastic Pollution. Disponível em: <https://iucn.org/resources/issues-brief/plastic-pollution> Acedido em 10/03/2025.
- [4] Kotlin. (n.d.). Introduction to Kotlin Multiplatform. Disponível em: <https://kotlinlang.org/docs/multiplatform-intro.html> Acedido em 10/03/2025.
- [5] Ktor. (n.d.). Welcome to Ktor. Disponível em: <https://ktor.io/docs/welcome.html> Acedido em 10/03/2025.