

TASA

Theater Auto Silence App

48297 João Marques

48305 Gonçalo Ribeiro

Orientador: Artur Ferreira

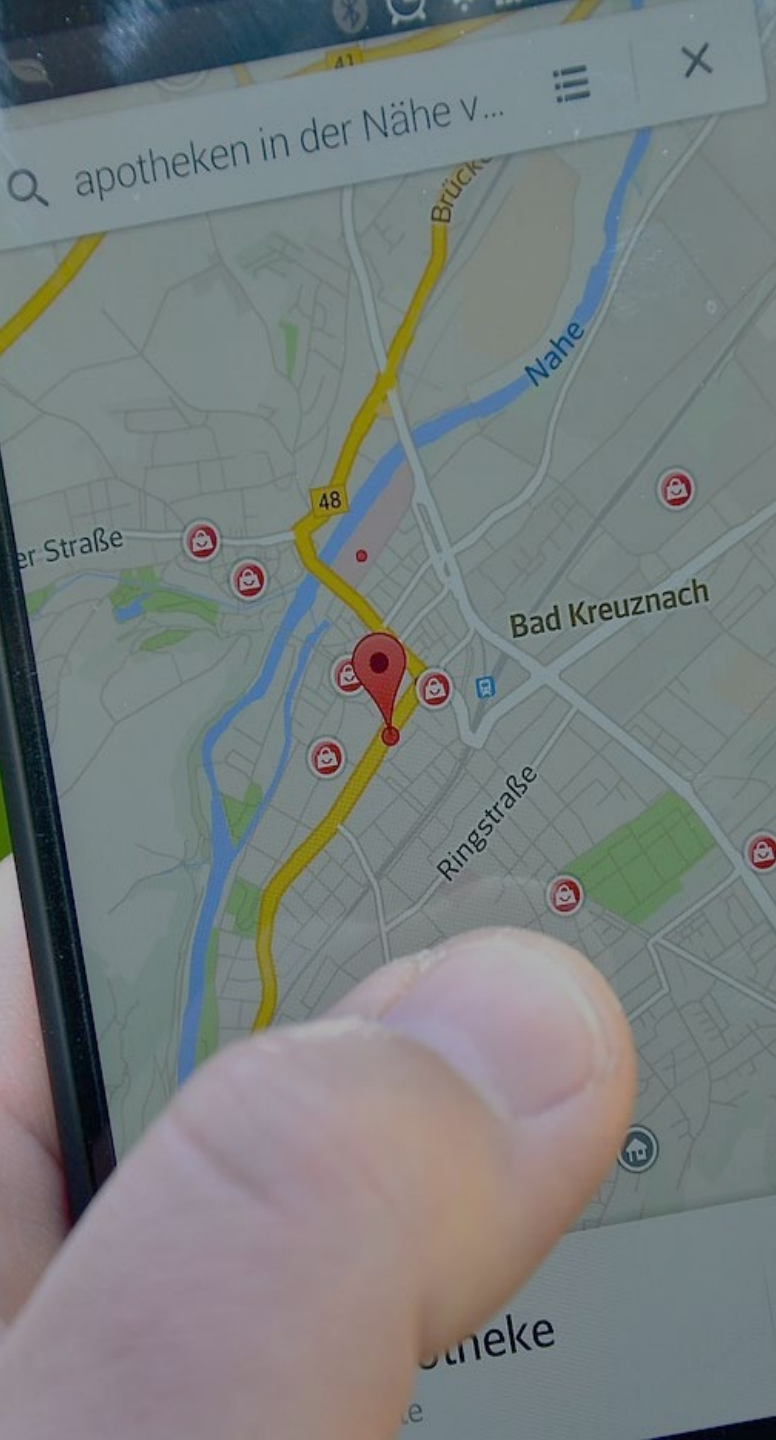


Introdução ao Projeto TASA



Introdução ao Projeto TASA

- Desenvolvimento de uma aplicação móvel que silencia o telemóvel automaticamente.
- Silenciamento baseado na localização geográfica ou em eventos agendados.
- Solução para evitar incómodos causados pelo esquecimento de colocar o telemóvel em silêncio.
- Foco em ambientes onde o silêncio é exigido (ex.: aulas, reuniões, eventos).



Modos de Funcionamento da Aplicação



Modo de Localização

O silenciamento será acionado automaticamente quando o utilizador se aproximar de locais previamente definidos.

Exemplos: salas de espetáculos, bibliotecas e estabelecimentos de ensino.

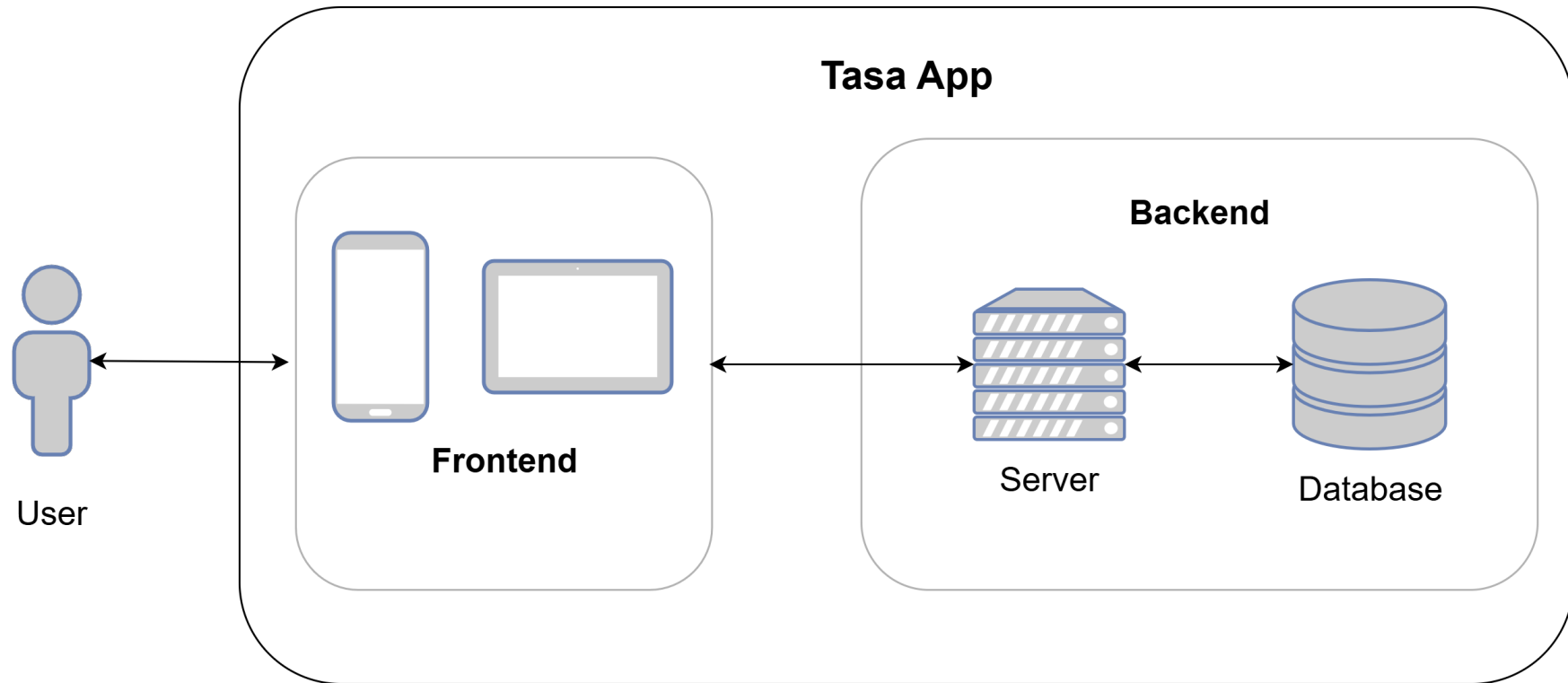


Modo de Atividade

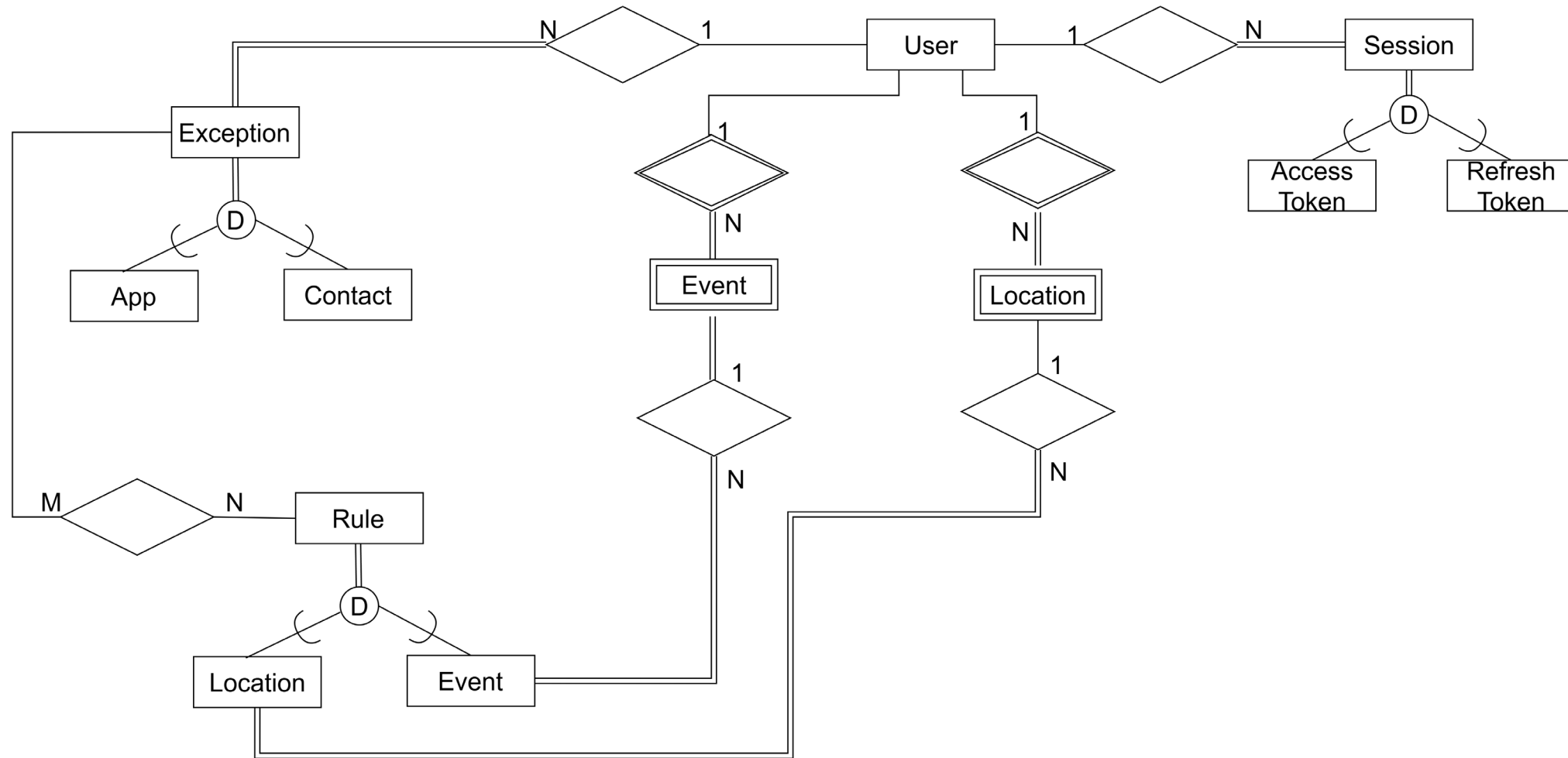
O dispositivo será silenciado em função da participação do utilizador em eventos específicos.

Exemplos: conferências, reuniões e exposições cinematográficas.

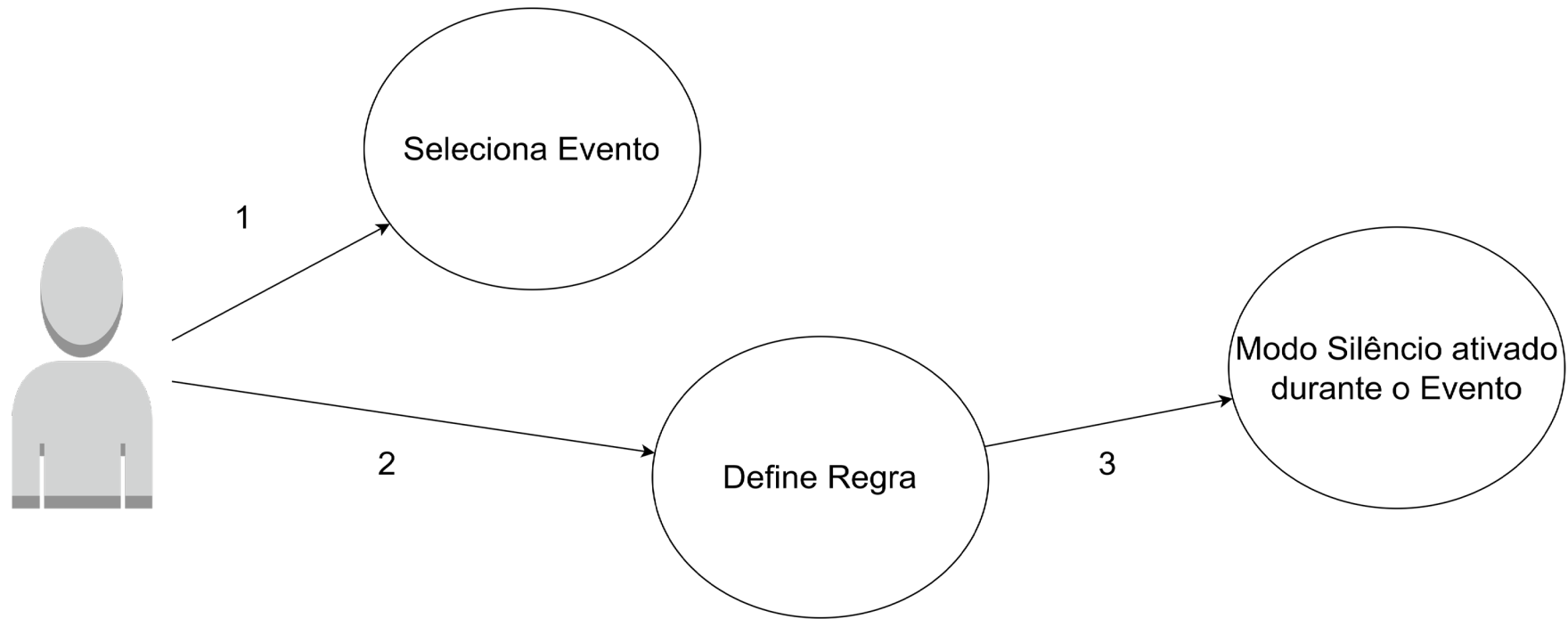
Solução



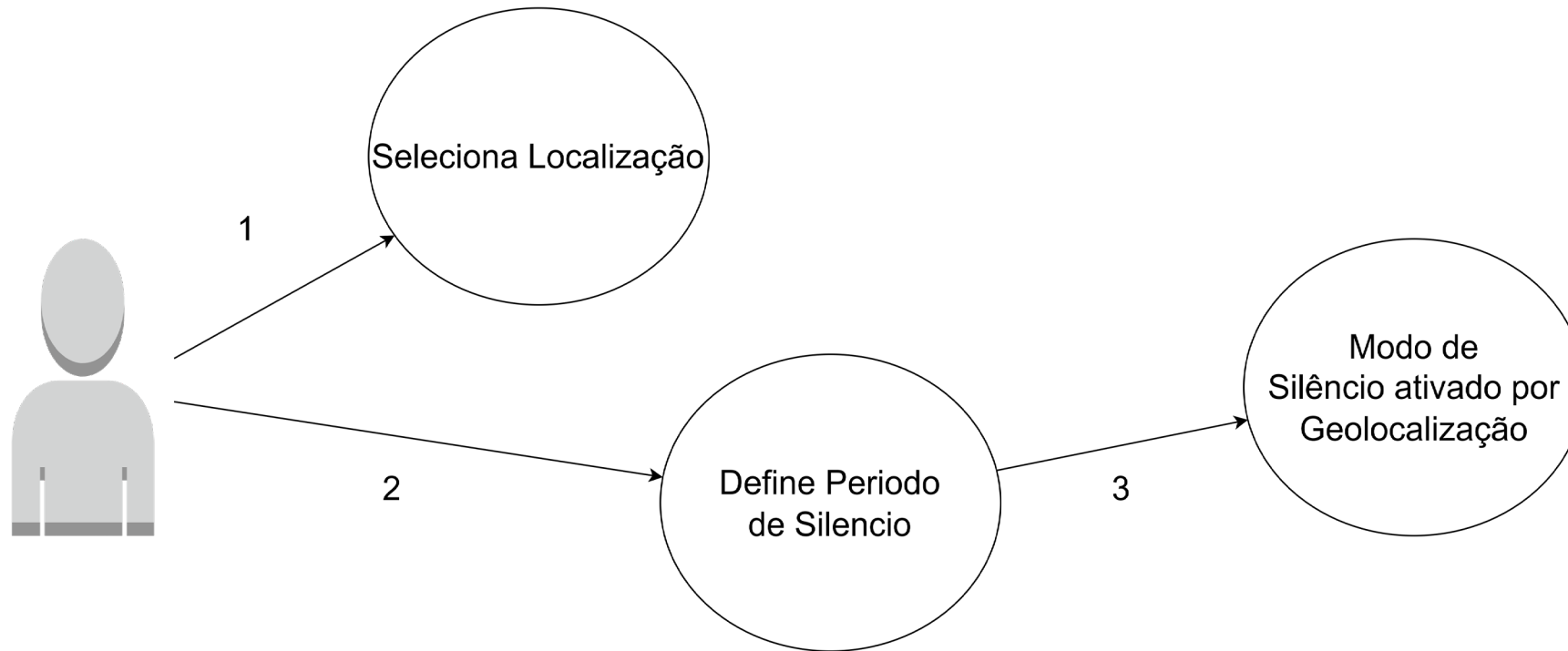
Modelo de dados



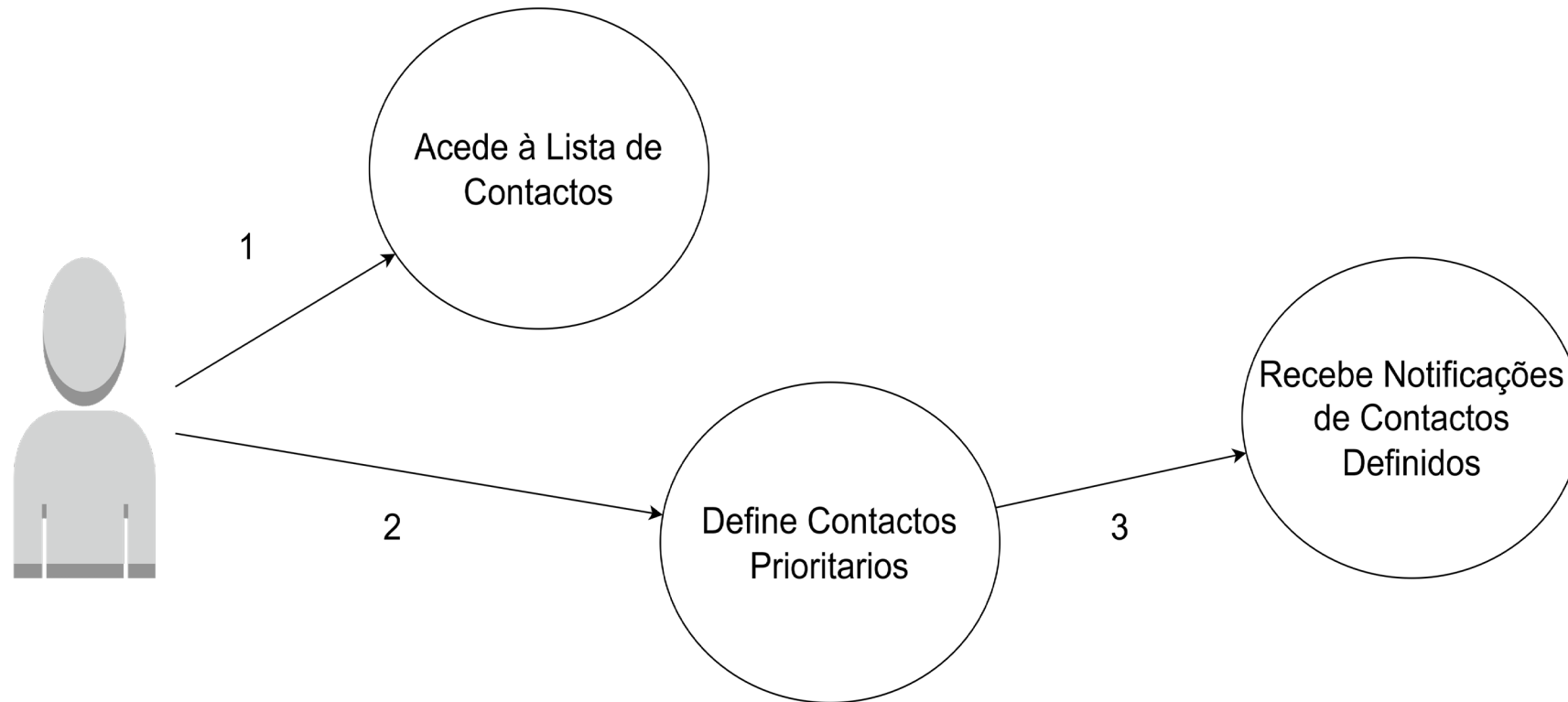
Caso de utilização 1



Caso de utilização 2



Caso de utilização 3



Requisitos Obrigatórios

1

Compatibilidade e Usabilidade

A aplicação deverá ser intuitiva e de fácil utilização.

Deverá funcionar em diferentes dispositivos, nomeadamente smartphones e tablets.

2

Silenciamento Automático

O dispositivo deverá ser silenciado automaticamente em áreas geográficas previamente definidas. O silenciamento deverá ser ativado em eventos pré-agendados.

3

Precisão de Localização

A aplicação deve obter a localização do utilizador com a maior precisão possível. Serão utilizadas ferramentas como Wi-Fi e Bluetooth além do GPS.

4

Eficiência Energética

O consumo de energia deve ser minimizado.

5

Integração com Agenda

A aplicação deverá integrar-se com a agenda do telemóvel.

Sincronizando eventos e compromissos que possam requerer o silenciamento automático.

6

Personalização das Notificações

O utilizador deverá poder seleccionar quais notificações (chamadas e mensagens) serão apresentadas durante o estado de silenciamento.

A exibição das notificações será baseada na prioridade definida pelo utilizador.

7

Acesso feito via API

Será disponibilizada uma API para o acesso e armazenamento de dados.

8

Base de Dados

Deverá existir uma base de dados para armazenar as regras que acionam o silenciamento automático.

Tecnologias - *Backend*



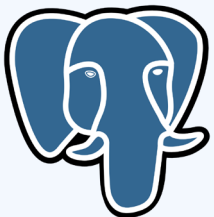
Kotlin

A linguagem Kotlin foi escolhida por ser moderna, fortemente tipificada e concisa. A familiaridade prévia do grupo com Kotlin facilitou a adoção, tornando desnecessária qualquer aprendizagem adicional.



Spring MVC

Optou-se pela utilização do framework Spring MVC devido à sua ampla adoção no desenvolvimento de APIs REST, à vasta disponibilidade de documentação e recursos de apoio. A experiência prévia do grupo com esta tecnologia contribuiu para acelerar o processo de desenvolvimento.



PostgreSQL

A escolha do PostgreSQL foi baseada na sua fiabilidade, no suporte a estruturas de dados complexas e no facto de ser uma solução open-source amplamente adotada. O grupo já tinha experiência com PostgreSQL, o que facilitou a sua integração no projeto.

Tecnologias - *Frontend*



Kotlin

A escolha da linguagem Kotlin para o *frontend* visa garantir a consistência tecnológica com o *backend*, simplificando a manutenção e o desenvolvimento da aplicação. Além disso, tendo em conta a experiência do grupo com programação Android no contexto académico, bem como o facto de a aplicação ter como alvo dispositivos Android, a escolha revelou-se particularmente adequada.



Jetpack Compose

O Jetpack Compose foi seleccionado por permitir a construção declarativa de interfaces de utilizador modernas e de elevado desempenho na plataforma Android. O domínio prévio da *framework* pela equipa eliminou a curva de aprendizagem e permitiu concentrar esforços na optimização da interface.



Ktor Client

O cliente HTTP Ktor foi escolhido pela sua integração nativa com Kotlin. O domínio desta biblioteca pela equipa agilizou a implementação das chamadas à API.

Ponto da situação



Março

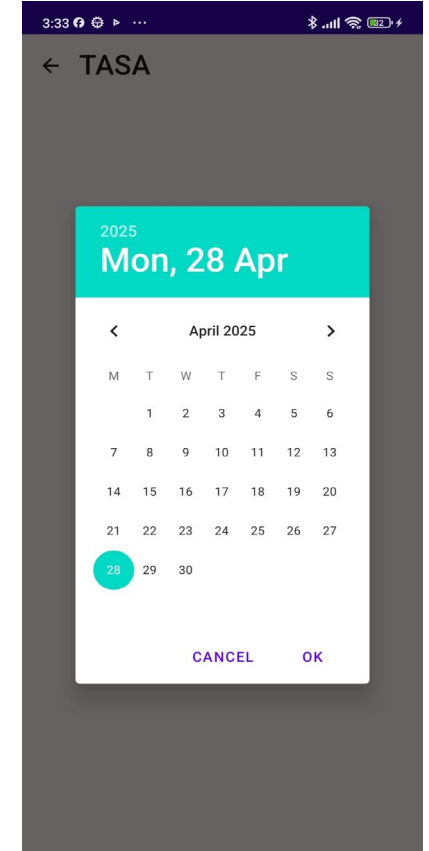
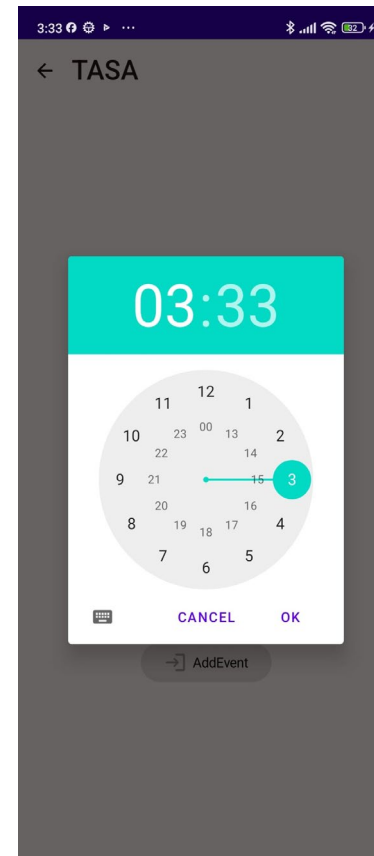
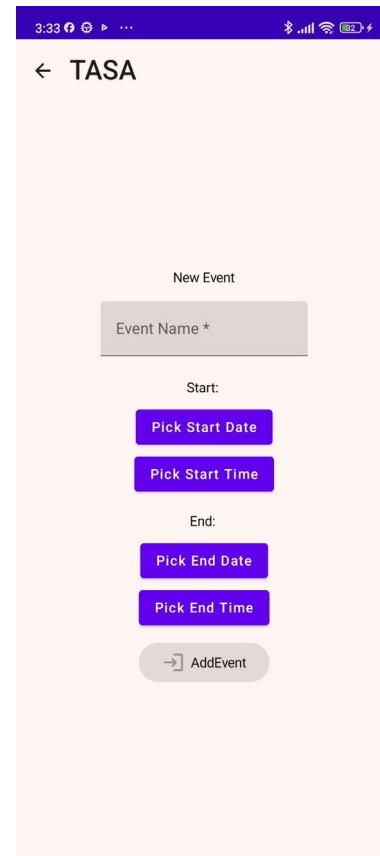
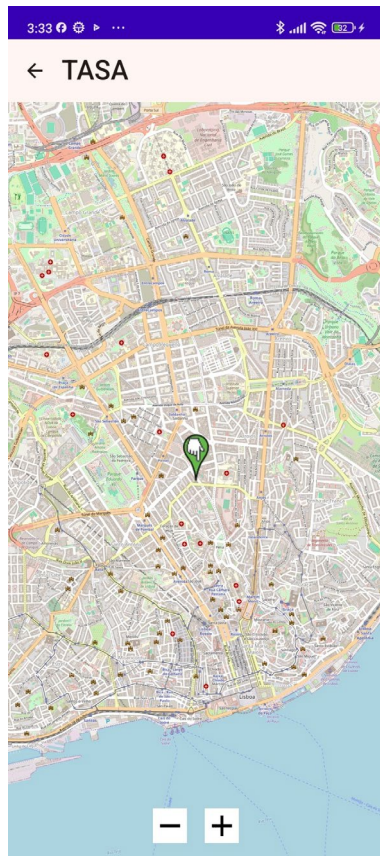
- Casos de utilização.
- Desenho da arquitetura da solução.
- Conceção do modelo de dados.
- Implementação do modelo de dados.
- Conceção da arquitetura do *backend*
- Implementação do *backend*
- Desenho e Implementação dos ecrãs da aplicação.



Abril

- Implementação do *backend* e testes.
- Desenho e Implementação dos ecrãs da aplicação.
- Preparar a Apresentação de Progresso.

Ecrãs



Próximos passos



Maio

- Implementar algoritmos/métodos de localização exata.
- Continuar a implementação do *frontend*.
- Melhorar o *backend*.
- Preparar a versão beta.



Junho

- Terminar o *frontend*.
- Testar o *frontend*.
- Entrega da versão beta.

Questões?

