

Aplicações Móveis

BinasJC

Sistema de Apoio ao Ciclismo em Centro Urbano baseado em Arquitectura Híbrida

Coordenação de Engenharia Informática

Departamento de Engenharias e Tecnologias

Instituto Superior Politécnico de Tecnologias e Ciências

Trabalho Prático

11 de Outubro de 2024

1. Introdução

O objectivo é desenvolver uma aplicação móvel distribuída, que visa fornecer um conjunto de funcionalidades aos ciclistas nos centros urbanos que usam o sistema BinasJC. A plataforma alvo é a versão Android ≥ 4.0 . O WiFi Direct será usado para comunicação sem fio.

Este projecto lhe dará a oportunidade de adquirir diversas habilidades na área de Aplicações Móveis, a saber: (a) projectar uma aplicação móvel com base em uma lista de requisitos, (b) implementar uma aplicação Android, (c) gerir redes sem fio móveis com base em WiFi Direct, (d) lidar com replicação de estado e consistência em redes sem fio móveis, (e) desenvolver técnicas adaptativas para melhorar a utilização de recursos, disponibilidade de dados e desempenho, e (f) desenvolver mecanismos de segurança para aplicações móveis.

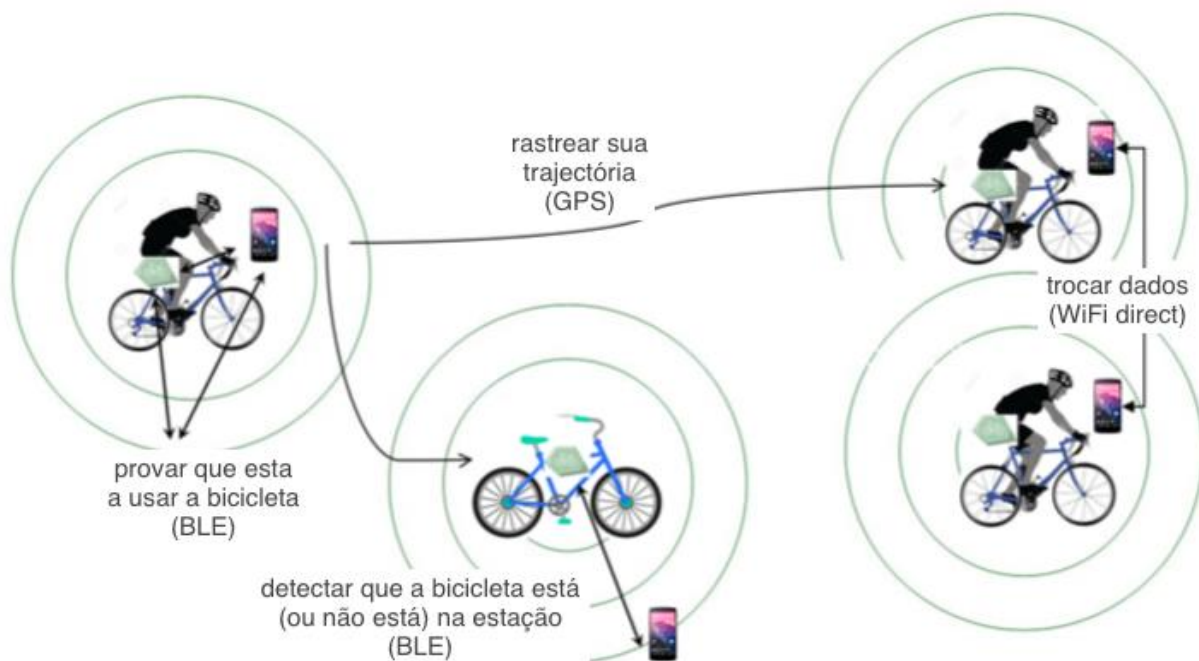


Figura 1: Cenário de uso da aplicação móvel

2. Especificação

2.1. Funcionalidade de linha de base

BinasJC é uma aplicação móvel que permite que os ciclistas ganhem e partilhem pontos enquanto pedalam (veja a Figura 1). Os ciclistas ganham pontos pedalando (mais distância percorrida se traduz em mais pontos ganhos). Além disso, os ciclistas podem i) enviar/receber pontos para/de outros ciclistas e ii) enviar/receber mensagens de texto para/de outros ciclistas. A comunicação entre ciclistas (para partilhar pontos e para trocar mensagens de texto) é feita pelo WiFi Direct.

A aplicação BinasJC é responsável por rastrear a trajectória (usando GPS) durante o ciclismo. O rastreamento começa quando o utilizador se aproxima da bicicleta. Usando GPS, a BinasJC rastreia (i) a distância (e trajectória) percorrida pedalando e (ii) quando o utilizador se afasta da bicicleta. Cada bicicleta tem um adesivo de beacon BLE (Bluetooth Low Energy) anexado, que permite que a BinasJC detecte quando o utilizador está a utilizar, ou não, uma bicicleta (específica).

As bicicletas são retiradas e deixadas em estações de estacionamento. Todos os utilizadores devem se registrar (utilizando o BinasJC) em um servidor central. O servidor sabe sobre todos os utilizadores actualmente registados no sistema, incluindo sua pontuação actual (número total de pontos) e trajectórias. Esses dados (pontuação e trajectórias) são carregados oportunisticamente pela aplicação BinasJC (em execução no dispositivo de cada utilizador) para o servidor central. Além disso, ao entrar em contacto com o servidor central, a aplicação móvel pode ser utilizada para reservar uma bicicleta disponível em uma estação de estacionamento específica.

O servidor sabe quais bicicletas estão disponíveis em cada estação de estacionamento porque a aplicação BinasJC informa o servidor sempre que uma bicicleta chega ou sai da estação. Para detectar esses eventos automaticamente, a aplicação pode utilizar heurísticas com base em leituras de GPS e detecção de beacons BLE. A aplicação em execução no dispositivo de um ciclista pode detectar um evento de retirada de bicicleta comparando a localização GPS actual do dispositivo com as coordenadas GPS da estação de estacionamento onde a bicicleta está estacionada e executar leituras GPS subsequentes para detectar que o dispositivo está a se afastar da estação de estacionamento. Para garantir que o ciclista esteja a pedalar a bicicleta (e não simplesmente indo embora), a aplicação deve ser capaz de detectar constantemente o sinal sem fio do beacon BLE da bicicleta. Para detectar um evento de entrega de bicicleta, o raciocínio é semelhante, excepto que, em vez de detectar que o dispositivo e o beacon estão a partir da estação de estacionamento, a aplicação deve detectar que estão a se aproximar de uma estação de estacionamento e, eventualmente, são imobilizados, e o utilizador se afasta da bicicleta.

Em resumo, precisa suportar as seguintes interações:

- Entre dispositivos móveis (usando WiFi Direct)
 - Enviar e receber pontos

- Enviar e receber mensagens de texto
- Entre dispositivos móveis e o servidor BinasJC
 - Registrar utilizador
 - Enviar nova trajectória
 - Apresentar as trajectórias mais recentes e passadas em um mapa
 - Obter informações do utilizador (incluindo pontuação actual e trajectórias)
 - Obter a lista de estações com bicicletas disponíveis para reservar
 - Reservar a bicicleta em uma estação específica (mostrando sua localização em um mapa)
- Entre dispositivos móveis, beacons BLE e o servidor BinasJC
 - Notificar a retirada da bicicleta
 - Notificar a entrega da bicicleta
- Entre o utilizador e a bicicleta em uso (usando BLE)
 - Detectar qual bicicleta está a ser usada e por qual utilizador

2.2. Funcionalidades avançadas

Os estudantes também podem implementar dois recursos avançados para pontos extras na classificação final: segurança e robustez. Esses recursos não se aplicam a mensagens de texto trocadas entre utilizadores. Observe que, para implementar esses recursos, pode confiar na aplicação, ou seja, assumir que ninguém é capaz de atacar o dispositivo móvel e comprometer a aplicação BinasJC. No entanto, não pode confiar na rede.

Para lidar com a segurança, ao considerar mensagens usadas para trocar pontos de um utilizador para outro, o sistema deve ser resiliente à replicação e temperagem de mensagens.

Por exemplo, se alguém espiona uma mensagem contendo pontos, deve ser impossível:

- reenviar a mensagem várias vezes para o mesmo utilizador ou para utilizadores diferentes, ou
- adulterar os dados na mensagem (por exemplo, aumentar o número de pontos na mensagem).

Em relação à robustez, o sistema deve suportar a reordenação de eventos. Como o servidor central é actualizado oportunisticamente, os utilizadores podem actualizar o servidor central com informações que dependem de eventos que ainda não estão no servidor central.

Por exemplo, imagine que o utilizador A pedala para ganhar pontos e então envia todos os pontos para o utilizador B. Agora, o utilizador B actualiza o servidor central antes que o utilizador A relate que ganhou pontos pedalando. Quando tais situações ocorrem, o sistema deve ser capaz de: i) aceitar os pontos dados ao utilizador B, e ii) actualizar a pontuação do utilizador A. Em outras palavras, o servidor deve sempre saber como os utilizadores ganharam seus pontos.

3. Implementação

A plataforma de destino para BinasJC é a versão Android ≥ 4.0 . A comunicação entre os dispositivos móveis deve ser baseada em WiFi Direct. A linguagem de programação oficial da disciplina para o projecto é Java, mas outras linguagens podem ser permitidas (solicite autorização ao docente via email). Os estudantes podem utilizar APIs do Android livremente. No entanto, bibliotecas de terceiros não são permitidas, a menos que explicitamente aprovadas pelo docente.

Infelizmente, não é possível fornecer dispositivos Android reais para testar o projecto. Portanto, o projecto deve ser testado em um testbed de emulação baseada em software, que compreende o emulador Android nativo e o emulador Termite WiFi Direct. Este testbed permitirá que emule: 1) dispositivos de ciclistas a executar a aplicação móvel, 2) comunicação entre dispositivos, 3) sinalização de beacon, 4) rastreamento por GPS e 5) comunicação com o servidor BinasJC. O testbed pode ser configurado em um único computador. Este computador também pode ser usado para hospedar o servidor BinasJC.

O emulador Android vem nativamente com Android SDK e fornece suporte para GPS e conectividade com a Internet. No entanto, não emula APIs WiFi Direct e Bluetooth.

Termite é um emulador de rede WiFi Direct. Termite pode emular uma rede sem fio de nós virtuais que se movem ao longo do tempo e são capazes de interagir oportunisticamente sempre que estiverem localizados em um alcance próximo. Os nós virtuais consistem em instâncias do emulador Android executando a aplicação de teste. O desenvolvedor é responsável por especificar a topologia e a dinâmica da rede virtual.

No contexto da BinasJC, o Termite deve ser utilizado para emular a comunicação WiFi Direct entre os dispositivos dos ciclistas e a sinalização/detecção de beacons BLE. Como os beacons BLE usam Bluetooth para anunciar sua localização, eles não podem ser simulados directamente pelo Termite. Portanto, para simplificar, os beacons BLE devem ser simulados como nós WiFi Direct normais que são usados exclusivamente para transmitir um identificador de rede para os dispositivos de ciclistas próximos.

Outra alternativa é utilizar dispositivos reais, composto por dois smartphone e um laptop (como foi explicado na aula).

A interface a ser disponibilizada pela aplicação BinasJC deve ser simples e, ao mesmo tempo, completa, ou seja, mostrar de forma clara todas as funcionalidades suportadas.

4. Etapas de Desenvolvimento

Recomendamos que desenvolva o projecto em cinco etapas:

1. **Desenhar a GUI:** estude os requisitos do projecto e desenhe a interface gráfica do utilizador da sua aplicação. Crie um *wireframe* de actividade da aplicação.
2. **Implementar a GUI:** implemente todos os componentes gráficos de sua aplicação incluindo a navegação entre telas. Neste ponto, não se preocupe com a rede. Use dados codificados para

simular a interação com o servidor ou Beacon. O rastreamento por GPS pode ser implementado nesta fase. Certifique-se de desenhar sua aplicação de forma modular.

3. **Comunicação com o servidor:** implemente o servidor central e estenda a aplicação móvel para se comunicar com o servidor.
4. **Comunicação WiFi-Direct:** complete a funcionalidade de linha de base do projeto implementando a comunicação WiFi Direct. Só precisa utilizar o Termite neste momento, se pretende testar no emulador.
5. **Funcionalidades avançadas:** implemente funcionalidades avançadas de segurança e robustez.

5. Avaliação e Marcos

Os trabalhos serão avaliados com base em várias dimensões.

Os pontos mais importantes a avaliar são:

- linhas de base implementadas e recursos avançados
- modularidade da implementação
- qualidade técnica dos algoritmos e protocolos
- decisões de eficiência de recursos.

Também avaliaremos a capacidade de resposta e a intuitividade da interface da aplicação. No entanto, a estética da GUI não será avaliada. Portanto, como dito acima, os estudantes devem manter a GUI o mais simples possível enquanto fornecem as informações necessárias.

Existem dois marcos importantes do projecto:

- **04 de Novembro de 2024:** Entrega da versão intermédia do projecto
- **05 de Novembro de 2024:** discussão (presença obrigatória de todos elementos do grupo)
- **29 de Dezembro de 2024:** Entrega da versão final do projecto – um protótipo totalmente funcional do BinasJC e um relatório final. As fontes do protótipo e o relatório devem ser enviados através de email e do GitHub, isto é, o relatório deve enviar via email (no relatório deve conter o link do repositório GitHub do projecto) e o código-fonte do projecto via GitHub (deve enviar um convite para a username `joaojdacosta`). Um modelo do relatório será partilhado. O relatório deve descrever o que foi e o que não foi implementado e é limitado a 5 páginas (excluindo a capa). A capa deve indicar o número do grupo, nome e número de cada um dos elementos do grupo. Obs.: O repositório do projecto deve ser privado.
- **04 e 06 de Janeiro de 2024:** Demo e Discussão.

6. Colaboração, Fraude e Classificação em Grupo

Grupos de estudantes são permitidos e incentivados a discutir as soluções técnicas de seus projectos sem mostrar, partilhar ou copiar código com outros grupos ou qualquer outra pessoa de dentro ou de fora do curso. Ferramentas de detecção de fraude serão utilizadas para detectar semelhanças de código. Todas as instâncias de fraude desqualificarão todos os grupos envolvidos e serão relatadas ao Departamento. Além disso, dentro de cada grupo, espera-se que todos os estudantes tenham um conhecimento prático de todo o código do projecto.