







A85308 Filipe Miguel Teixeira Freitas Guimarães A85242 Maria Miguel Albuquerque Regueiras A86271 Renata Gomes Dias Ribeiro



Universidade do Minho Escola de Engenharia



Abordagem usada para obter o modelo da cidade de Braga



Abordagem usada para o *deploy* dos sensores e portais

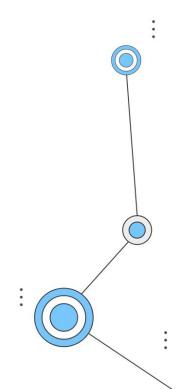


Estado do desenvolvimento do projeto



Atuais tarefas e dificuldades

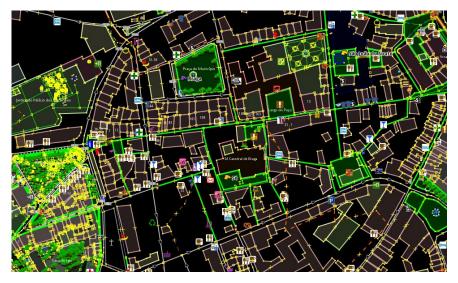
Conteúdo





Modelo de Braga

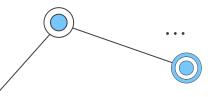
Para obter o modelo de Braga recorremos JOSM (Java OpenStreetMap Editor), que é um editor de mapas presentes no OpenStreetMap.



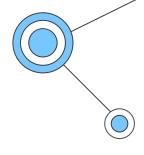
Decidimos por optar um mapa que cobrisse todo o centro de Braga e periferia.

Como se pode ver na imagem à esquerda (apenas uma porção do nosso mapa) está cheia de conteúdo que não nos interessa para este projeto.





Simplificação do mapa



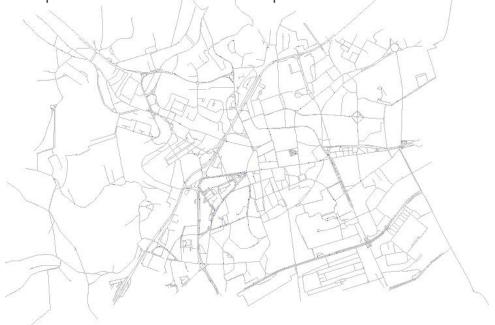


Do mapa obtido apenas necessitamos das estradas e equiparáveis para simular a circulação dos carros. Para isto recorremos, mais uma vez, ao *JOSM* para apagar as estruturas (e.g. edifícios) resultando no mapa ao lado apresentado.



Prepara o mapa para o *The One*

Para conseguir adicionar o mapa criado ao *The one* precisamos de converter para um formato que o simulador conseguisse abrir. Para isso recorremos a uma ferramenta disponível chamada *OSM2WKT* para transformar o nosso ficheiro .osm em .wkt.



Com o ficheiro convertido e mudando as default_settings para o abrir o nosso mapa conseguimos agora o mapa no simulador.



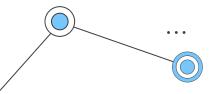
Sensores, Portais e Carros

O *The One* trata estes elementos como grupos. Para os definir recorremos ao ficheiro default_settings.txt, bem como à documentação disponível online para criar um groups.txt com todas as configurações para os sensores, portais e carros.

```
******************
# Propriedades para todos os Grupos #
Group.movementModel = ShortestPathMapBasedMovement
Group.router = EpidemicRouter
Group.bufferSize = 5M
Group.waitTime = 0, 120
# All nodes have the bluetooth interface
Group.nrofInterfaces = 1
Group.interface1 = btInterface
# Walking speeds
\#Group.speed = 0.5, 1.5
Group.speed = 0.0, 0.0
# Message TTL of 300 minutes (5 hours)
Group.msgTtl = 300
```

```
# Grupo 1 - Sensores #
Group1.nrofHosts = 100
Group1.groupID = S
# Grupo 2 - Portais #
Group2.nrofHosts = 5
Group2.groupID = P
Group2.interface2 = highspeedInterface
# Grupo 2 - Carros #
# group2 specific settings
Group3.nrofHosts = 40
Group3.groupID = C
# cars can drive only on roads
Group3.okMaps = 1
# 10-50 km/h
Group3.speed = 2.7, 13.9
```





Sensores

Para fazer os sensores gerarem 10kb de dados por minutos tivemos de criar um evento por sensor. Como são 100 sensores são 100 eventos. Para não ser um processo repetitivo decidimos criar um programa em Java que replica os eventos, fazendo as respetivas alterações. Escreve num ficheiro events.txt como podemos ver nas seguintes imagens.

```
Events.nrof = 100
Events1.class = MessageEventGenerator
Events1.interval = 60,60
Events1.size = 10k,10k
Events1.hosts = 0,0
Events1.tohosts = 100,105
Events1.prefix = M
Events2.class = MessageEventGenerator
Events2.interval = 60,60
Events2.size = 10k,10k
Events2.hosts = 1,1
Events2.tohosts = 100,105
Events2.prefix = M
```

```
Events99.class = MessageEventGenerator
Events99.interval = 60,60
Events99.size = 10k,10k
Events99.hosts = 98.98
Events99.tohosts = 100,105
Events99.prefix = M
Events100.class = MessageEventGenerator
Events100.interval = 60,60
Events100.size = 10k,10k
Events100.hosts = 99,99
Events100.tohosts = 100,105
Events100.prefix = M
```



Dúvidas e dificuldades

 Necesitamos criar um evento por sensor para garantir a taxa de dados de 10kb por minuto?

 Sensores podem comunicar diretamente com os portais? Para já a nossa implementação permite isso.

Podemos colocar Portais/Sensores em sitios estratégicos?

