

# **Universidade do Minho**

Escola de Engenharia  
Mestrado em Engenharia de Telecomunicações e Informática

Redes Veiculares

## **Relatório Final**

Grupo 1 - METI 24/25

- Bruno Miguel Fernandes Araújo PG55806
- Fernando João Santos Mendes PG55807
- Diogo José Ribeiro Araújo A101778

# Índice

---

<b>Índice.....</b>	<b>1</b>
<b>1. Introdução e Objetivos.....</b>	<b>1</b>
<b>2. Mapa.....</b>	<b>2</b>
<b>2.1. Entidades da rede.....</b>	<b>3</b>
<b>3. Tecnologias e Protocolos de comunicação.....</b>	<b>4</b>
<b>3.1. AD-HOC.....</b>	<b>4</b>
<b>3.2. Celular.....</b>	<b>4</b>
<b>4. Funcionalidades.....</b>	<b>4</b>
<b>4.1. Mapas internos ( LDM).....</b>	<b>4</b>
<b>4.2. Processamento de Dados e Atuação.....</b>	<b>5</b>
<b>5. Testes e resultados.....</b>	<b>6</b>
<b>6. Conclusões e trabalho futuro.....</b>	<b>8</b>
<b>Referências.....</b>	<b>8</b>

## 1. Introdução e Objetivos

---

Neste relatório foi desenvolvido um sistema de segurança rodoviária inteligente baseado em redes veiculares. O projeto foi implementado com recurso à plataforma Eclipse MOSAIC, simulando um cenário urbano da cidade de Guimarães. A comunicação entre veículos e infraestrutura é feita através de trocas de mensagens, permitindo a deteção de comportamentos de condução perigosos. Com base nestas mensagens, o nó Fog processa os dados recolhidos pela RSU e envia alertas DENM para os veículos afetados, contribuindo para a melhoria da segurança e gestão do tráfego.

Os objetivos propostos para este projeto são os seguintes:

- Configurar e simular um cenário de tráfego real ( mapa, rotas, tipos de veículos);
- Estabelecer a comunicação desde os veículos até ao nó FOG e vice-versa (considerando encaminhamento com mais que um salto (FOG - RSU - Veículo) ;
- Em cada elemento da rede (veículo, RSU e FOG) deve ser implementados os mapas internos “LDM” (lista de vizinhos);
- Processar os dados e as respetivas informações em prol da verificação das condições de segurança rodoviária (verificação dos valores da velocidade dos veículos);
- Mediante os resultados devem ser implementadas medidas de atuação passiva e ativa;

## 2. Mapa

---

Na figura 1 é possível observar o mapa utilizado para realizar a simulação. De modo a obter este mapa foi usado um website que faz a captura das ruas e estruturas assim como os limites de velocidade [\[1\]](#) de seguida foi necessário realizar uma conversão do tipo de ficheiro para um compatível com o netedit (map.net.xml). Através deste processo conseguimos obter um mapa de uma zona movimentada da cidade de Guimarães. De modo a correr esta simulação será necessário adicionar este ficheiro “C:\...\eclipse-mosaic-24.1\map.net.xml”.



Figura 1: Mapa da cidade de Guimarães.

De modo a simular trânsito de veículos decidimos criar três rotas que passam por ruas principais e interceptam-se. Estas rotas são possíveis de observar na figura 2 onde a rosa temos a rota 0, a amarelo a rota 1 e a verde a rota 2.



Figura 2: Mapa das rotas implementadas.

## 2.1. Entidades da rede

Para realizar uma simulação onde haja trocas e processamento de mensagens é necessário que existam dois tipos de nós, um nó RSU (Road-Side Unit) que será responsável por ser o intermediário entre os veículos e o nó de processamento. O nó responsável pelo processamento de dados é o nó FOG. Na figura 3 é possível observar o posicionamento do nó RSU e FOG que foi utilizado estando estes dois no mesmo local. Embora nesta simulação seja utilizada apenas um RSU este projeto é escalável, sendo possível adicionar mais RSUs e nós FOGs.

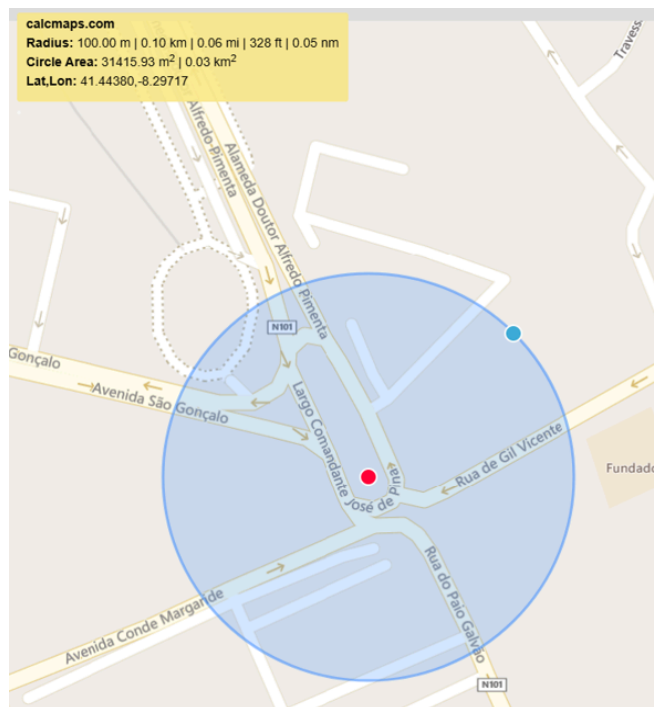


Figura 3: Posição do RSU e FOG.

De modo a que esta simulação decorra com troca e processamento de mensagens é necessário que existam veículos para as gerar. De acordo com o objetivo definido foi necessária a implementação de dois tipos de veículos, um veículo que circula a velocidade legal e sem comportamentos de risco e o segundo caracterizado por circular em excesso de velocidade e eventualmente exercer comportamentos de risco.

## 3. Tecnologias e Protocolos de comunicação

---

### 3.1. AD-HOC

A comunicação entre veículos e de veículos para RSU rege-se segundo o conceito de V2X (vehicle to everything), cujas mensagens são enviadas segundo o formato CAM, sendo a tecnologia de comunicação AD-HOC. uma comunicação wireless sem dependência a uma estrutura fixa. Estas mensagens são enviadas de forma periódica com periodicidade de 1 ms.

As mensagens de aviso provenientes do nó FOG são retransmitidas, pelo RSU, através desta mesma tecnologia de comunicação para os veículos dentro da sua área de cobertura.

### 3.2. Celular

A comunicação entre os nós RSU e FOG é realizada através de tecnologia de comunicação celular. O protocolo utilizado nas mensagens varia consoante a situação, podendo ser CAM ou DENM.

O formato CAM é usado quando o RSU retransmite para o nó FOG, as mensagens CAM enviadas pelos veículos.

As mensagens de aviso relativas a infrações de velocidade, construídas com base no protocolo DENM, têm origem no nó FOG. Após a deteção dos infratores, o FOG envia os avisos em broadcast para as RSUs, que, por sua vez, os retransmitem também em broadcast para os veículos dentro da sua área de cobertura.

## 4. Funcionalidades

---

### 4.1. Mapas internos ( LDM)

Por cada mensagem recebida quer seja num veículo, RSU ou FOG irá ser criado e atualizado um mapa interno para cada entidade ou seja caso um veículo envie uma CAM a outro este que a recebeu irá adicionar o nome do veículo que a enviou assim como o timestamp da receção desta mensagem, após estas inserções. irá percorrer o hasmap a verificar se ainda está em contacto com todos os elementos da mesma, caso não tenha recebido uma mensagem de uma entidade em quatro tempos de transmissão essa será removida do mapa interno. Tanto o veículo como o RSU e o FOG irão ter um função semelhante à que poderá verificar em “`updateVizinhosMap()`” da classe “`UserTaggedValueReadingApp`”.

## 4.2. Processamento de Dados e Atuação

Quando o nó FOG recebe mensagens da RSU o mesmo procura retirar as informações como a velocidade, id, rota do veículo e o limite de velocidade da respetiva estrada do local de quando foi enviada a mensagem. O parâmetro da velocidade do veículo é então comparado com o valor do limite de velocidade e feita uma análise do resultado, tal é possível averiguar através da função “**Speeding()**” da classe “**FogValueReadingApp**”.

- Caso a velocidade instantânea seja maior que o limite da velocidade é formulada uma mensagem DENM com o propósito de ser enviado para todos os veículos que estiverem na rota do veículo em excesso de velocidade e este inclusive.
- A mensagem a ser enviada varia mediante o agravamento da diferença da condição em cima enunciada, compreende-se uma mensagem para uma velocidade até cerca de 20km/h a mais do que o limite e outra para valores superiores a essa referência.

Estas condições são possíveis de verificar na função “**sendDenmToRsu()**” presente na classe “**FogValueReadingApp**”.

Após a mensagem DENM ser enviada para a RSU esta deverá retransmitir para os veículos na sua área e quando os mesmos recebem é realizada a seguinte filtragem:

- Os veículos retiram as informações como o id da rota e o id do veículo em excesso de velocidade.
- Cada veículo verifica se a sua rota e o seu id coincide com aqueles que foram retirados da mensagem DENM, em caso de apenas a rota ser verificada como igual os carros recebem uma mensagem de aviso (que por sinal é escrito nos logs dos mesmos).  
Caso ambos os parâmetros sejam verificados significa que o veículo em questão se trata daquele que está em limite de velocidade e como tal além de uma atuação passiva (recebe uma mensagem para que reduza a velocidade).

Foi implementado também uma atuação ativa, ou seja, o carro automaticamente inicia a travagem até o mesmo atingir a velocidade legal, isto foi possível dando uso do método “**changeSpeedWithInterval()**” presente na classe abstrata “**VehicleOperatingSystem**”.

## 5. Testes e resultados

A implementação do programa foi concebida a pensar na escalabilidade, permite a adição e remoção de RSUs, FOGs ou veículos, o sistema funciona corretamente independentemente dessas alterações.

Vamos analisar passo a passo um teste simples para facilitar a compreensão do sistema, temos presentes 1 RSU, 1 FOG e 2 veículos na mesma rota em que um deles tem uma grande aceleração (Tipo Speeder) com intenção de eventualmente exceder o limite de velocidade.

Verifica-se a comunicação entre os dois veículos, na figura 4 podemos ver o envio das mensagens CAM que originam no veículo 0 e na figura 5 a receção destas pelo veículo 1.

```
2025-05-29 13:58:38,397 INFO - Set up (at simulation time 6.000,000,000 s)
2025-05-29 13:58:38,401 INFO - Sending CAM (at simulation time 6.100,000,000 s)
2025-05-29 13:58:38,468 INFO - Sending CAM (at simulation time 6.200,000,000 s)
2025-05-29 13:58:38,470 INFO - Sending CAM (at simulation time 6.300,000,000 s)
2025-05-29 13:58:38,481 INFO - Sending CAM (at simulation time 6.400,000,000 s)
2025-05-29 13:58:38,487 INFO - Sending CAM (at simulation time 6.500,000,000 s)
2025-05-29 13:58:38,501 INFO - Sending CAM (at simulation time 6.600,000,000 s)
```

Figura 4: Excerto do ficheiro log do veículo com id 0.

```
2025-05-29 13:58:38,543 INFO - Set up (at simulation time 7.000,000,000 s)
2025-05-29 13:58:38,559 INFO - CAM message arrived: CAM:Hello from veh_0 velocidade 48.46158333333333 tempo 7.0E9
aceleracao -4.5 paragem 0.0 direcao 347.36197318435194 posicao CartesianPoint{x=1003.80,y=788.65,z=0.00}
roadidorg.eclipse.mosaic.lib.routing.database.LazyLoadingRoadPosition@83c9ac9e rota r_2 speedlimit 50.004000000000005 (at
simulation time 7.000,400,000 s)
```

Figura 5: Excerto do ficheiro log do veículo com id 1 (Vizinho do veículo 0).

Na figura 6 é possível observar a receção e retransmissão das mensagens CAM no RSU que originaram no veículo 1.

```
2025-05-29 13:58:38,433 INFO - RSU recebeu uma mensagem V2X: Cam (at simulation time 6.100,900,000 s)
2025-05-29 13:58:38,436 INFO - CAM message arrived: CAM:Hello from veh_0 velocidade 52.96158333333333 tempo 6.1E9 aceleracao 0.0 paragem 0.0 direcao
346.64395540785137 posicao CartesianPoint{x=1014.40,y=741.36,z=0.00}roadidorg.eclipse.mosaic.lib.routing.database.LazyLoadingRoadPosition@19dcedc6 rota
r_2 speedlimit 50.004000000000005 (at simulation time 6.100,900,000 s)
2025-05-29 13:58:38,438 INFO - Retransmissao da CAM como InterVehicleMsg através de CELL: CAM:Hello from veh_0 velocidade 52.96158333333333 tempo
6.1E9 aceleracao 0.0 paragem 0.0 direcao 346.64395540785137 posicao CartesianPoint{x=1014.40,y=741.36,z=0.00}
roadidorg.eclipse.mosaic.lib.routing.database.LazyLoadingRoadPosition@19dcedc6 rota r_2 speedlimit 50.004000000000005 (at simulation time 6.100,900,000
s)
```

Figura 6: Excerto do ficheiro log do nodo RSU (Retransmissão da mensagem CAM).

O nó FOG recebe essa informação e, após realizar os devidos cálculos, envia a mensagem de aviso. A RSU procede então à retransmissão da mensagem. Estes procedimentos podem ser observados nas Figuras 7 e 8.

```
2025-05-29 13:58:38,474 INFO - Received a V2X message: InterVehicleMsg[senderPosition=GeoPoint{lat=41.443746,lon=-8.297167,alt=0.00}, name=rsu_0,
dados=CAM:Hello from veh_0 velocidade 52.96158333333333 tempo 6.1E9 acelaracao 0.0 paragem 0.0 direcao 346.64395540785137 posicao CartesianPoint{x=
1014.40,y=741.36,z=0.00}roadidorg.eclipse.mosaic.lib.routing.database.LazyLoadingRoadPosition@19dcedc6 rota r_2 speedlimit 50.004000000000005] (at
simulation time 6.302,374,999 s)
2025-05-29 13:58:38,478 INFO - Sender Name: rsu_0 (at simulation time 6.302,374,999 s)
2025-05-29 13:58:38,478 INFO - Message CAM:Hello from veh_0 velocidade 52.96158333333333 tempo 6.1E9 acelaracao 0.0 paragem 0.0 direcao
346.64395540785137 posicao CartesianPoint{x=1014.40,y=741.36,z=0.00}roadidorg.eclipse.mosaic.lib.routing.database.LazyLoadingRoadPosition@19dcedc6 rota
r_2 speedlimit 50.004000000000005 (at simulation time 6.302,374,999 s)
2025-05-29 13:58:38,478 INFO - Veículo id: veh_0 (at simulation time 6.302,374,999 s)
2025-05-29 13:58:38,480 INFO - Fog node sent DENM (at simulation time 6.302,374,999 s)
2025-05-29 13:58:38,480 INFO -

Speeding:

Veiculo veh_0 desloca-se a 52.96158333333333 km/h numa zona com limite de velocidade 50.004000000000005
```

Figura 7: Excerto do ficheiro log do nodo FOG.

```
2025-05-29 13:58:38,489 INFO - RSU recebeu uma mensagem V2X: Denm (at simulation time 6.554,281,276 s)
2025-05-29 13:58:38,489 INFO - Denm message arrived, :
V2XMessage{routing=MessageRouting{sourceAddressContainer=SourceAddressContainer{sourceAddress=NetworkAddress{address=/10.2.0.1}, sourceName=rsu_1,
sourcePosition=GeoPoint{lat=41.443747,lon=-8.297166,alt=0.00}},
destinationAddressContainer=DestinationAddressContainer{address=NetworkAddress{address=/255.255.255.255}, type=CELL_GEOCAST, channelId=null,
protocolType=UDP, timeToLive=null, disseminationArea=GeoCircle{center=GeoPoint{lat=41.443747,lon=-8.297166,alt=0.00},radius=1000.00}}, id=6} (at
simulation time 6.554,281,276 s)
2025-05-29 13:58:38,494 INFO - Denm message sent:
V2XMessage{routing=MessageRouting{sourceAddressContainer=SourceAddressContainer{sourceAddress=NetworkAddress{address=/10.2.0.1}, sourceName=rsu_1,
sourcePosition=GeoPoint{lat=41.443747,lon=-8.297166,alt=0.00}},
destinationAddressContainer=DestinationAddressContainer{address=NetworkAddress{address=/255.255.255.255}, type=CELL_GEOCAST, channelId=null,
protocolType=UDP, timeToLive=null, disseminationArea=GeoCircle{center=GeoPoint{lat=41.443747,lon=-8.297166,alt=0.00},radius=1000.00}}, id=6} (at
```

Figura 8: Excerto do ficheiro log do nodo RSU (Retransmissão da mensagem DENM).

Por fim, o veículo recebe a mensagem DENM. O veículo 0, sendo o infrator, é notificado e é forçado a reduzir a sua velocidade, enquanto o veículo 1 apenas recebe a notificação de que existe, na sua rota, um veículo a exceder o limite de velocidade. Nas figuras 9 e 10 vemos o sucedido.

```
2025-05-29 13:58:38,495 INFO - Denm message arrived (at simulation time 6.556,681,276 s)
2025-05-29 13:58:38,495 INFO -

Por favor diminua a sua velocidade! Está a pôr em perigo a sua segurança e a dos outros condutores.
```

Figura 9: Excerto do ficheiro log do veículo 0 (Notificação da infração).

```
2025-05-29 13:58:38,567 INFO - Denm message arrived (at simulation time 7.023,167,787 s)
2025-05-29 13:58:38,568 INFO -

Cuidado! Um veiculo encontra-se em excesso de velocidade na sua rota!
```

Figura 10: Excerto do ficheiro log do veículo 1 (Notificação de um malfeitor na sua rota).



## 6. Conclusões e trabalho futuro

---

Este projeto demonstrou a eficácia de um sistema de segurança rodoviária baseado em redes veiculares, recorrendo à plataforma Eclipse MOSAIC para simular um cenário urbano em Guimarães. Através da comunicação V2X (CAM/DENM) entre veículos, RSUs e o nó FOG, foi possível detetar excessos de velocidade e emitir alertas em tempo real. O processamento no FOG permitiu diferenciar infrações leves e graves, desencadeando notificações ou ações corretivas automáticas nos veículos.

Os testes comprovaram a escalabilidade do sistema, com troca eficiente de mensagens e respostas adequadas a eventos dinâmicos. Como trabalho futuro, propõe-se a expansão da rede com a inserção de métodos de routing entre os próprios veículos e eventualmente orientar as comunicações do nó FOG para as devidas RSU 's.

Enfrentámos algumas limitações inerentes ao próprio software Eclipse MOSAIC. O facto de o programa ser sequencial limita as funcionalidades que podem ser implementadas. Para além disso, os intervalos de tempo fornecidos como argumento à função responsável pela desaceleração dos veículos têm de ser cuidadosamente definidos, uma vez que certos valores podem levar o programa a falhar. Apesar destas e de outras limitações, conseguimos desenvolver as aplicações com as funcionalidades esperadas.

## Referências

---

[1] openstreetmap - Disponível em

WWW:<URL:<https://www.openstreetmap.org/#map=15/41.44215/-8.29704>:>

[2] AdaCompNUS - Disponível em

WWW:<URL: [https://adacompnus.github.io/summit-docs/tutorials/preparing\\_maps/](https://adacompnus.github.io/summit-docs/tutorials/preparing_maps/)>

[3] Github eclipse mosaic - Disponível em

WWW:<URL:<https://github.com/eclipse-mosaic/mosaic>>