



# Distributed Road Traffic Speed Monitoring

An EE4-T Final Year Project

Jeremy Chan

w/ Dr. Ed Stott

w/ Dr. Christos Papavassiliou

# At a glance...

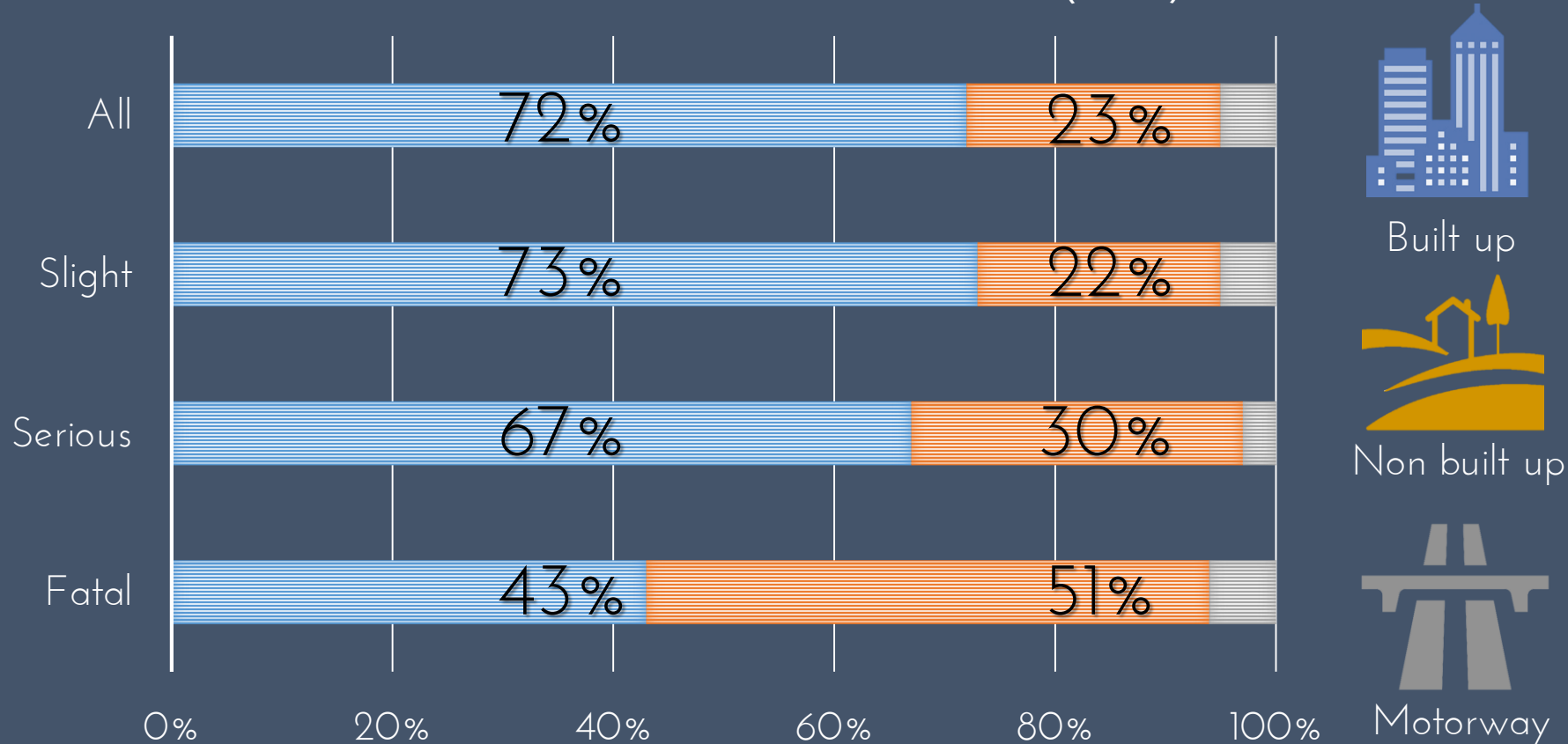


- Implementation based project
- Catch speeding motorists using a camera and Raspberry Pi
- Existing license plate recognition (LPR) technologies
- New peer to peer network (P2P)
- No government controlled central server
- Social posting
- Motivation / Specification / High Level Design / Implementation and Testing / Demo / Evaluation and Conclusion



# Motivation - Accident areas

CASUALTIES BY SEVERITY AND ROAD TYPE (2015)



- 95% of accidents not on motorway
- Target residential areas

# Motivation - Fixed point speed cameras



# Motivation - Fixed point speed cameras



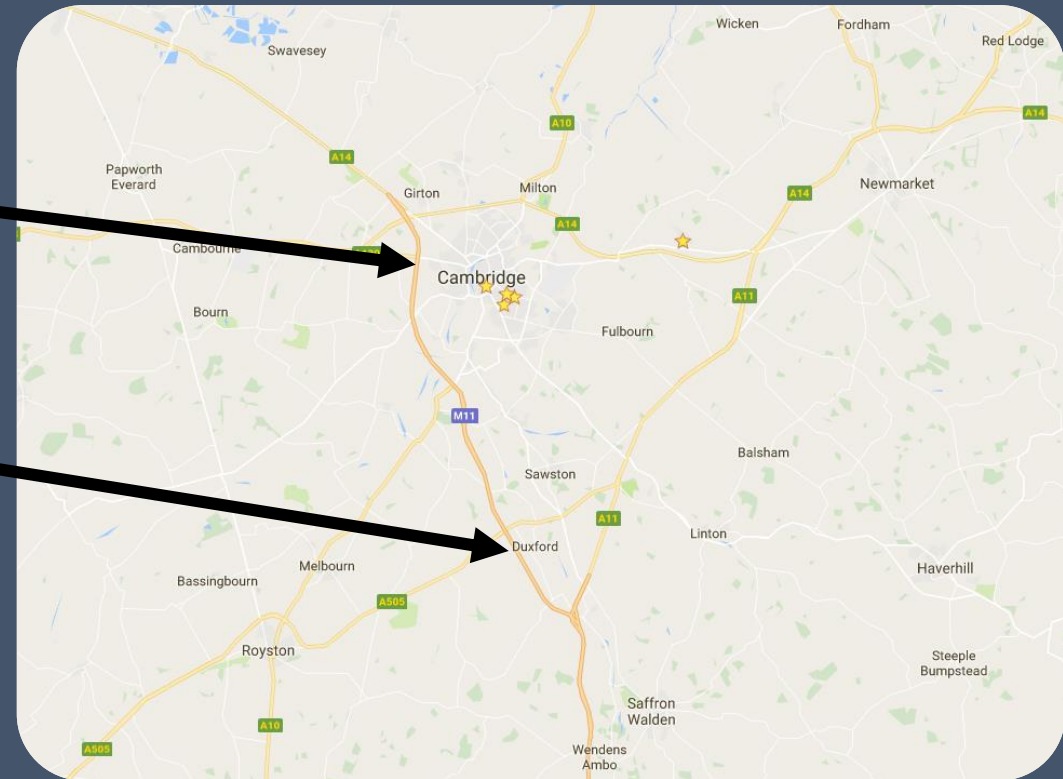
# Motivation - Fixed point speed cameras



- Detrimental to streetscape
- Prefer discreet appearance
- Only take speed at a point



# Motivation - Average speed cameras



- None on residential routes

# Motivation - Existing license plate cameras



- Hard to buy for general public
- >£ 1000
- Proprietary storage of plates



# Motivation - Summary



New system should:

- Provide average speed check
- Target general public
- Priced reasonably
- Not aim to directly ticket offender
- Use as evidence and leverage for demands

# Product Specification - Main Deliverables



1. Implement a UK number plate recognition system using existing computer vision algorithms on a low-cost, readily available hardware platform.
2. Set up a peer-to-peer network to share vehicle passing times and detect violations without the need for a central server.
3. Publish photo evidence of any violations

# Product Specification - Main Deliverables



1. Implement a UK number plate recognition system using existing computer vision algorithms on a low-cost, readily available hardware platform.
2. Set up a peer-to-peer network to share vehicle passing times and detect violations without the need for a central server.
3. Publish photo evidence of any violations

# Product Specification - Main Deliverables



1. Implement a UK number plate recognition system using existing computer vision algorithms on a low-cost, readily available hardware platform.
2. Set up a peer-to-peer network to share vehicle passing times and detect violations without the need for a central server.
3. Publish photo evidence of any violations

# Product Specification - Main Deliverables



1. Implement a UK number plate recognition system using existing computer vision algorithms on a low-cost, readily available hardware platform.
2. Set up a peer-to-peer network to share vehicle passing times and detect violations without the need for a central server.
3. Publish photo evidence of any violations



# Product Specification - Advanced Deliverables



- Have a standalone mode in case there is no one in the network
- Be able to join/leave/return to the network easily
- Make the network secure against rogue peers
- Package the system so that it can be easily installed in a home by an inexperienced user

# Product Specification - Advanced Deliverables



- Have a **standalone mode** in case there is no one in the network
- Be able to join/leave/return to the network easily
- Make the network secure against rogue peers
- Package the system so that it can be easily installed in a home by an inexperienced user

# Product Specification - Advanced Deliverables



- Have a standalone mode in case there is no one in the network
- Be able to join/leave/return to the network easily
- Make the network secure against rogue peers
- Package the system so that it can be easily installed in a home by an inexperienced user

# Product Specification - Advanced Deliverables



- Have a standalone mode in case there is no one in the network
- Be able to join/leave/return to the network easily
- Make the network **secure** against **rogue peers**
- Package the system so that it can be easily installed in a home by an inexperienced user

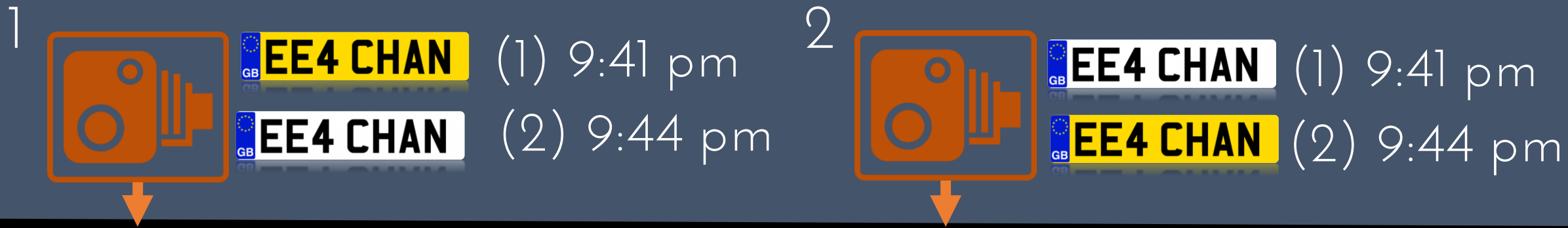
# Product Specification - Advanced Deliverables



- Have a standalone mode in case there is no one in the network
- Be able to join/leave/return to the network easily
- Make the network secure against rogue peers
- Package the system so that it can be easily installed in a home by an inexperienced user



# Envisioned System



Yellow - locally detected  
White - peer detected



# High Level Design



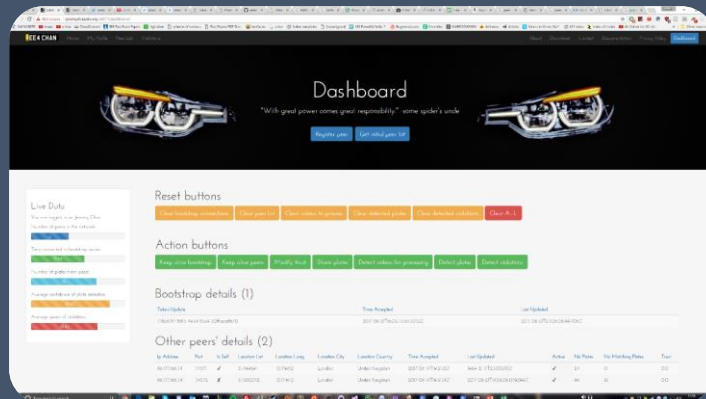
License Plate Detection



Peer to Peer Network

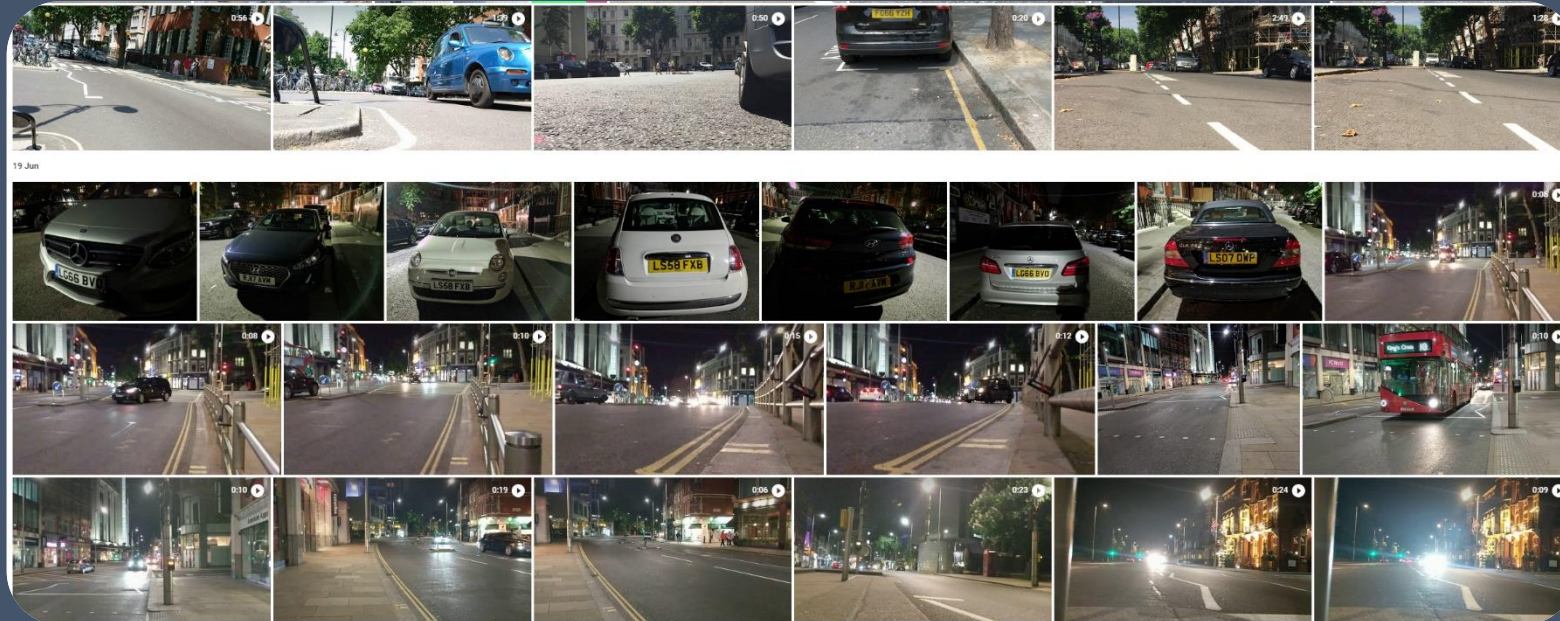
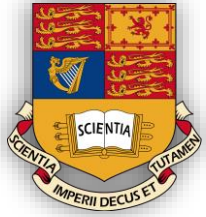


Web Interface



- 3 well defined modules
- Readily available components
- Users have control over data

# License Plate Recognition - Aim



- Get license plates of cars
- Function in multiple angles and lighting conditions

# License Plate Recognition - Design



Store videos locally using motion detection

Process videos when idle

Detect license plates using **OpenALPR**

Extract unique plates

Store plates with metadata in database

- Multiple parameters for tweaking

- Consecutive frames will likely be the same car

- Time
- Location

- OpenALPR computationally expensive
- Buffer videos
- Survive busy periods and power outages

# License Plate Recognition - Fuzzy Grouping



<<<<<1< LG65FBO <2< LG65FBP <3< LG65FBO  
<4< FE16RBX <5< LTZ1077 <6< LTZ077 <7< LTZ107

	Plate1	Plate 2	String Similarity (0.75)
1-2	LG65FBO	LG65FBP	0.857
2-3	LG65FBP	LG65FBO	0.857
3-4	LG65FBO	FE16RBX	0.285
4-5	FE16RBX	LTZ1077	0.143
5-6	LTZ1077	LTZ077	0.923
6-7	LTZ077	LTZ107	0.857

- Robust separation of unique vehicles
- Works with misreads
- Sort groups by confidence



# License Plate Recognition - Fuzzy Grouping



<<<<<1< LG65FBO <2< LG65FBP <3< LG65FBO  
<4< FE16RBX <5< LTZ1077 <6< LTZ077 <7< LTZ107

	Plate1	Plate 2	String Similarity (0.75)
1-2	LG65FBO	LG65FBP	0.857
2-3	LG65FBP	LG65FBO	0.857
3-4	LG65FBO	FE16RBX	0.285
4-5	FE16RBX	LTZ1077	0.143
5-6	LTZ1077	LTZ077	0.923
6-7	LTZ077	LTZ107	0.857

- Robust separation of unique vehicles
- Works with misreads
- Sort groups by confidence

# License Plate Recognition - Testing



LF64FBO



LM55BCK

# License Plate Recognition - Testing



BX66HHO



LTZ1077



# License Plate Recognition - Testing



# License Plate Recognition - Testing



5SHUTTLE



18001



# License Plate Recognition - Matching

@ @ # # @ @ @ or @ @ @ # # # #

@ Letter  
# Number

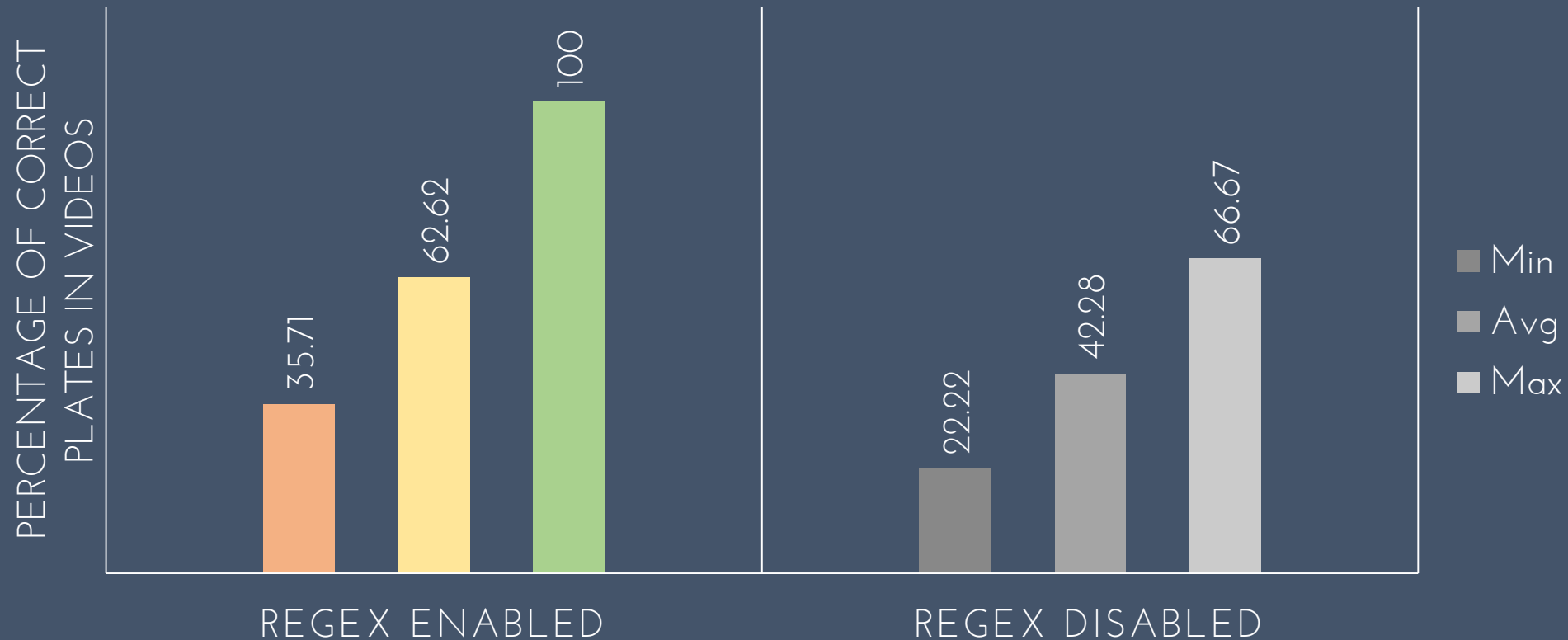


# License Plate Recognition - Matching

@@##@@@ or @@@####

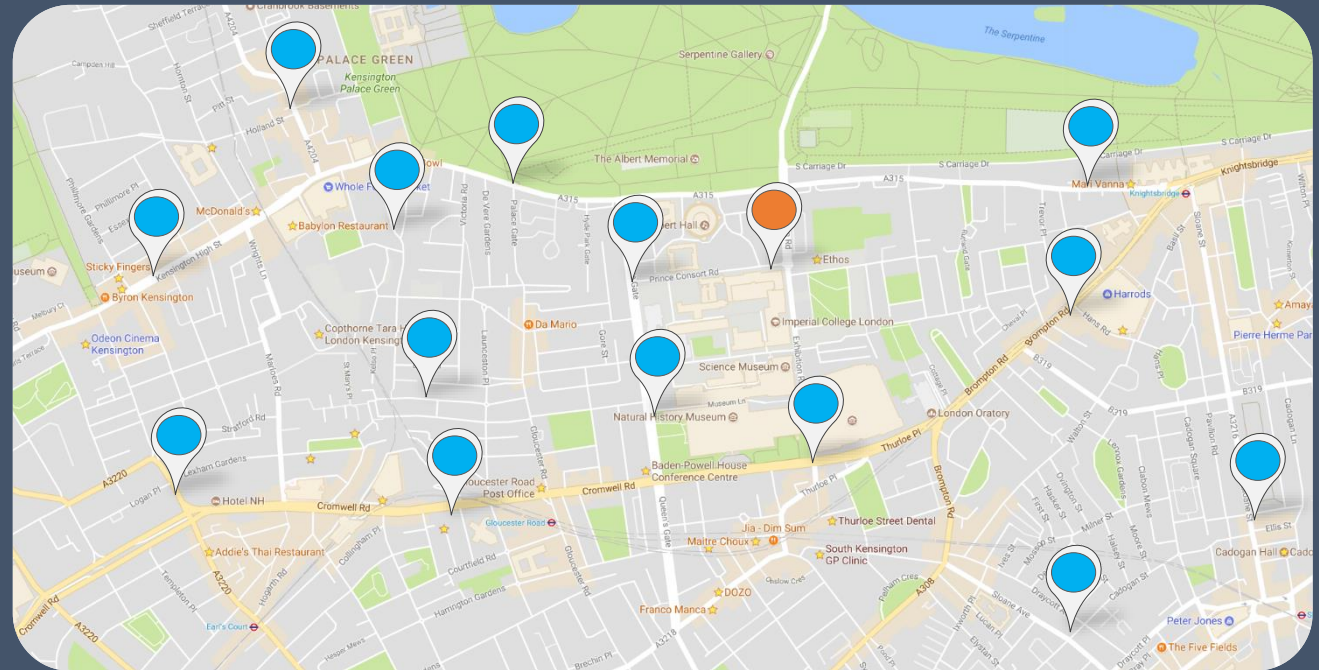
@ Letter  
# Number

AUTOMATED TEST SUITE OF 20 VIDEOS



- LPR output compared with manual
- Max 100%

# Peer to Peer Network - Aim



- Transmit detected plates to other peers
- Work out average speed between peer locations

# Peer to Peer Network – Network transactions



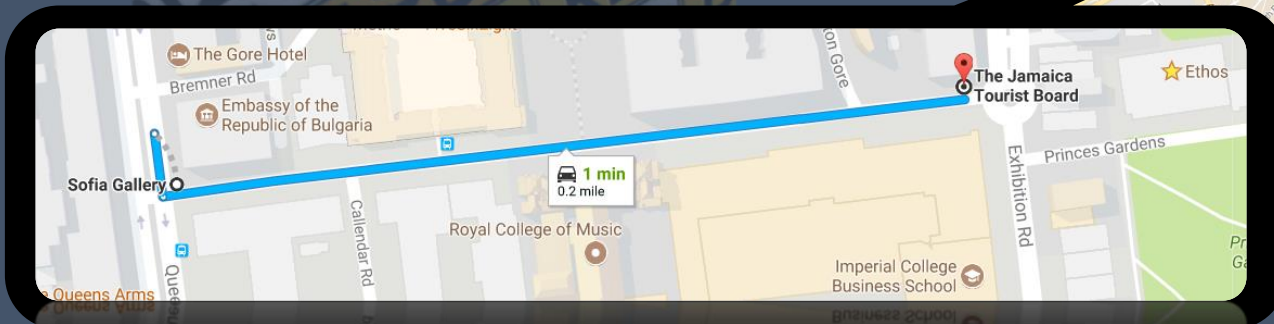
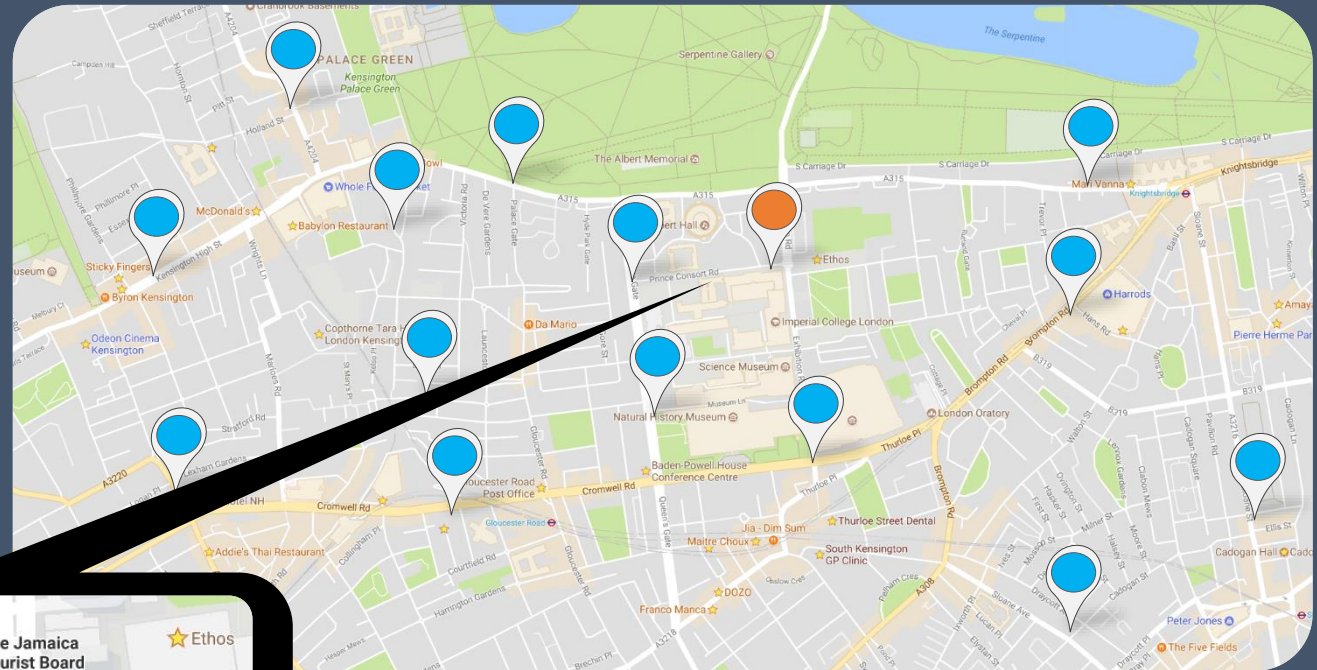
Register*
Get list of peers
Keep alive
Send plates
Detect violations
Modify trust of peer

- Bootstrap the network
- Maintain the network
- Security issues

\*) Requires central server

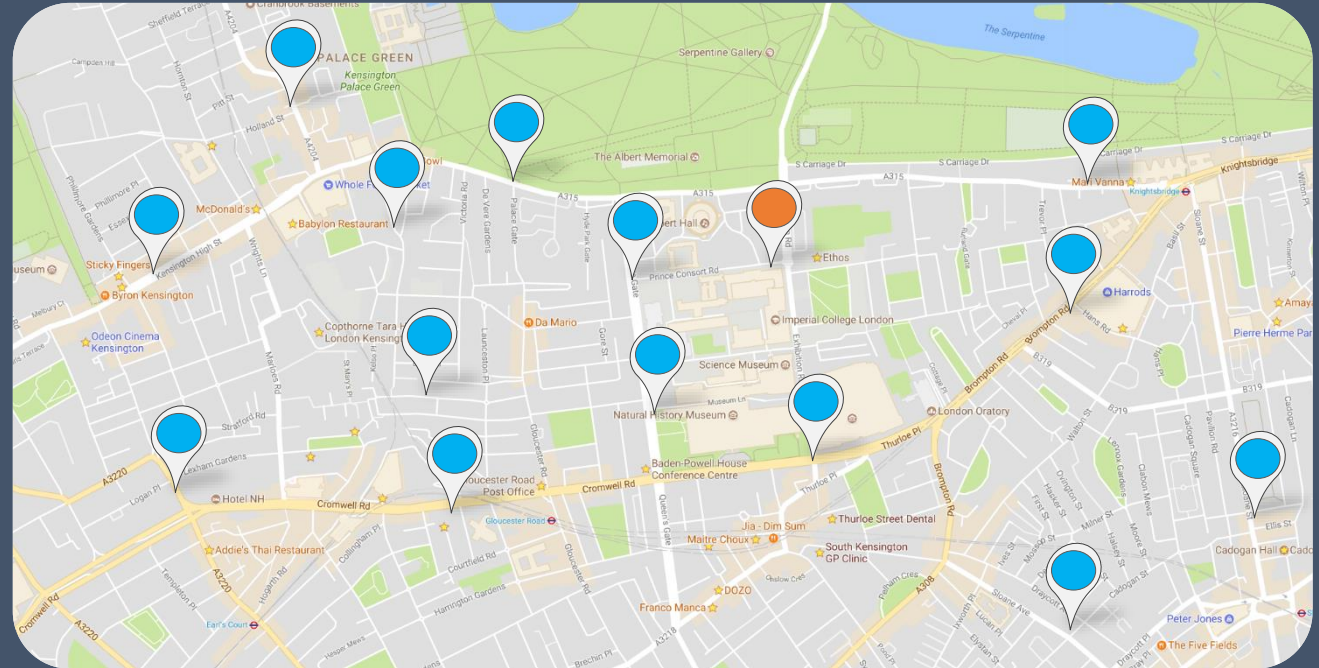


# Peer to Peer Network - Average speed



Google Maps APIs

# Peer to Peer Network - Average speed





# Peer to Peer Network - Server



ID: fbbb1cbf-7887-4cc6-9bbf-16707e11367e



GET  
POST  
PATCH  
DELETE



authentication

- P2P on top of web server
- API
- Unique ID authentication
- Local encryption



ID: 01ed4cfa-6c5b-476f-a6f8-4c6d0bcd5975



ID: 4feb662e-f50a-4088-bafe-ce75b421d50c



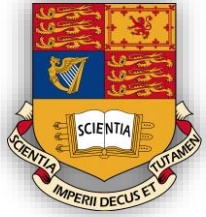
# Peer to Peer Network - Design



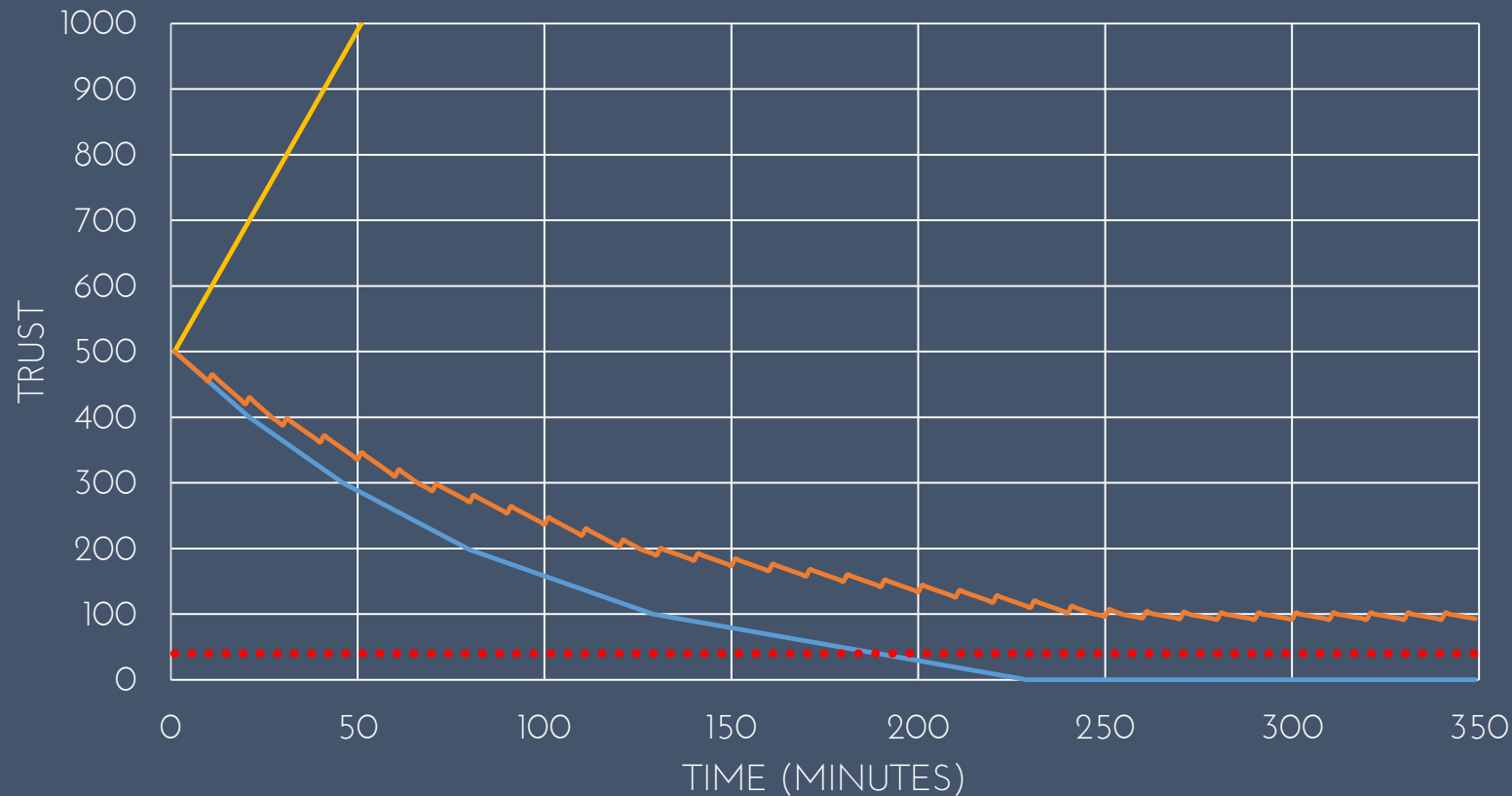
- Tried, tested, scalable framework
- Most security issues taken care of



# Peer to Peer Network - Trust system



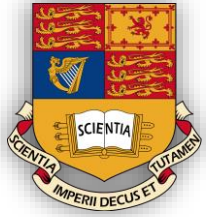
TRUST DECAY (0.99)



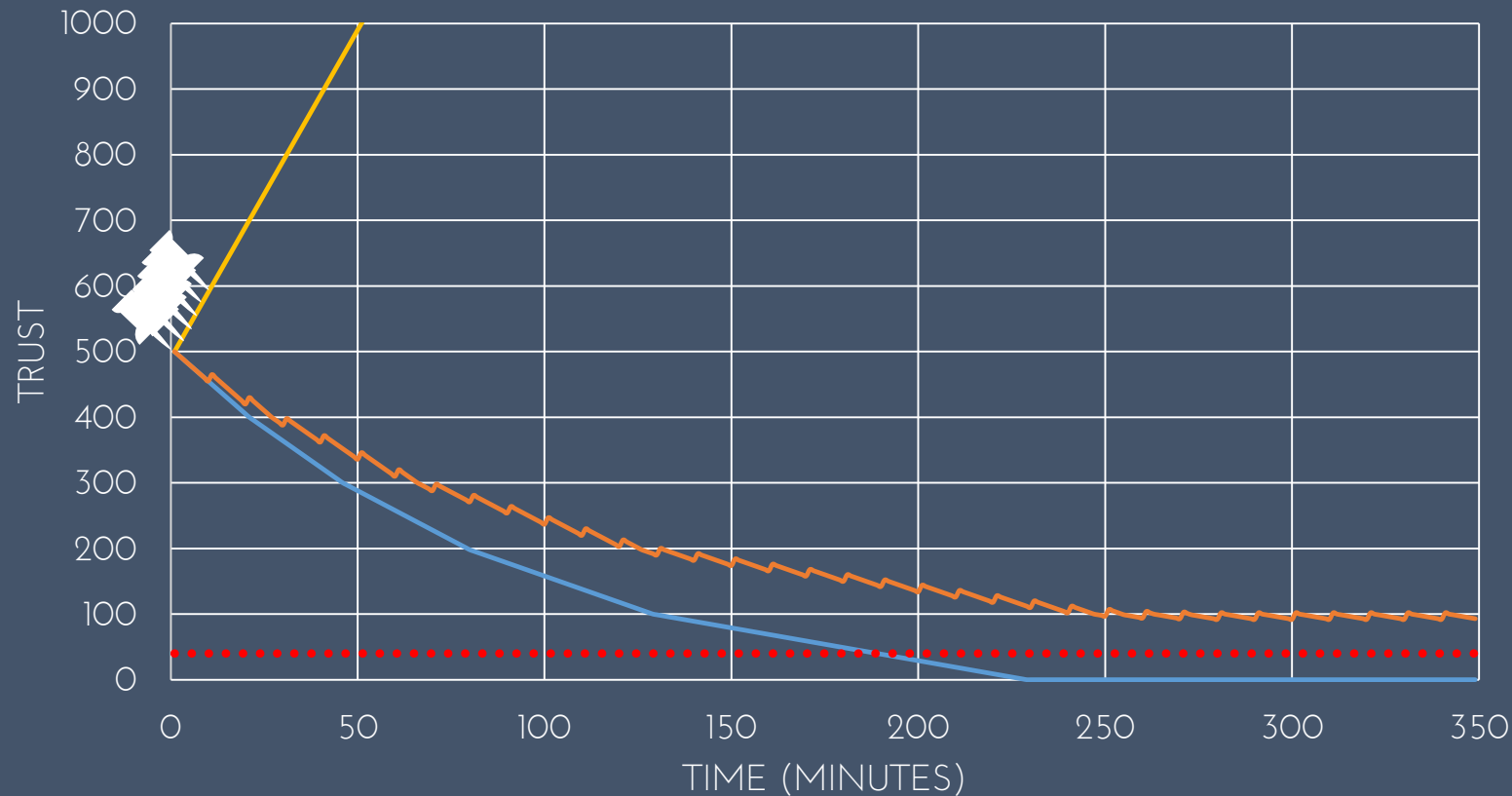
- Rogue Peer
- Genuine Peer (1min)
- Genuine Peer (10min)
- ... Threshold

Self	Peer (1min)
LK14VJV	
LX04CZT	
RF66BEU	
KN14BVY	
SH14PVZ	
LL12EFH	

# Peer to Peer Network - Trust system



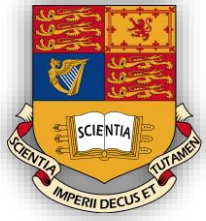
TRUST DECAY (0.99)



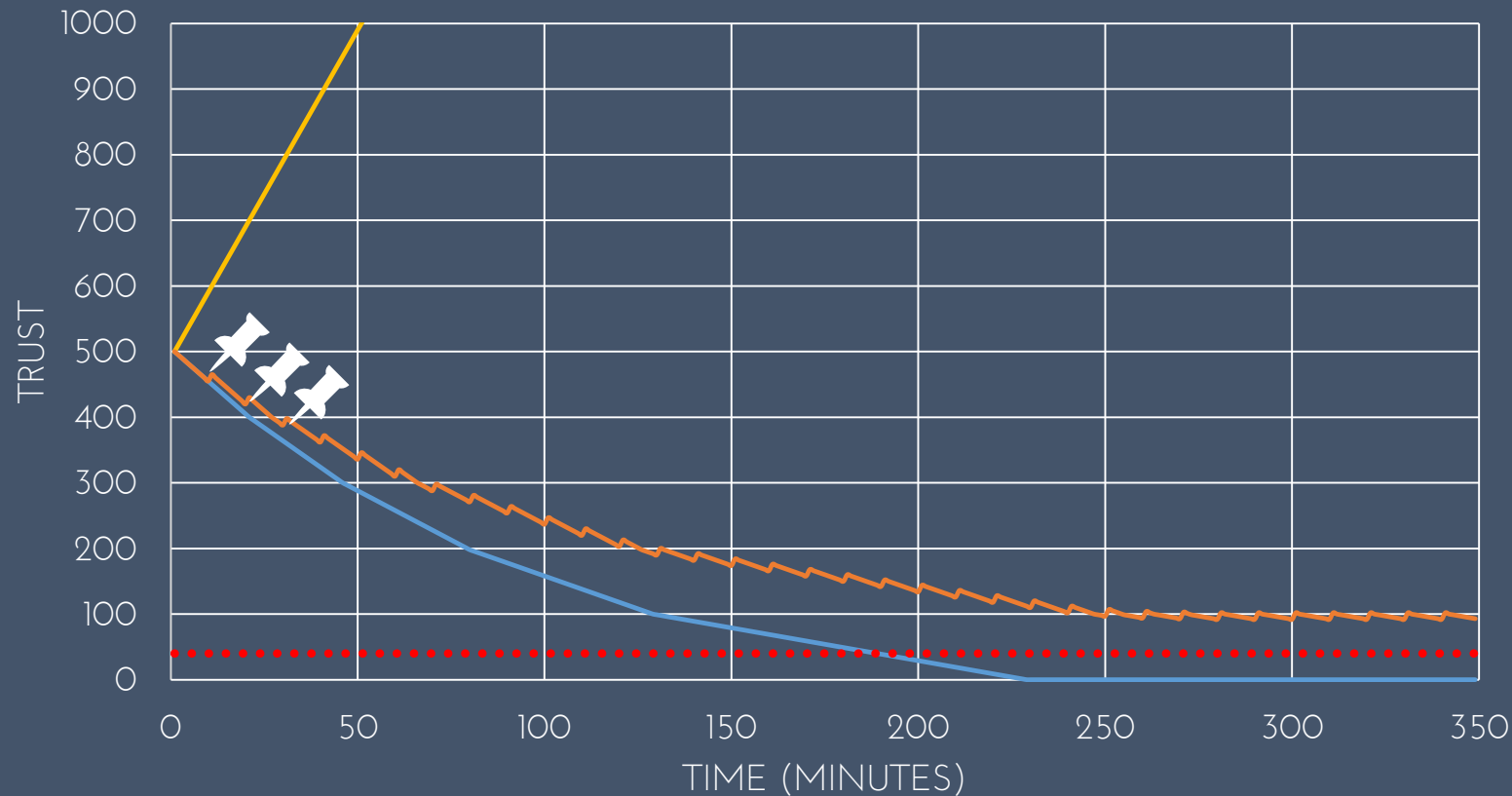
- Rogue Peer
- Genuine Peer (1min)
- Genuine Peer (10min)
- ... Threshold

Self	Peer (1min)
LK14VJV	LK14VJV
LX04CZT	LX04CZT
RF66BEU	RF66BEU
KN14BVY	KN14BVY
SH14PVZ	SH14PVZ
LL12EFH	LL12EFH

# Peer to Peer Network - Trust system

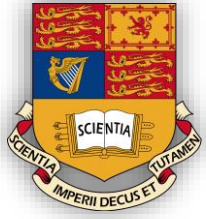


TRUST DECAY (0.99)

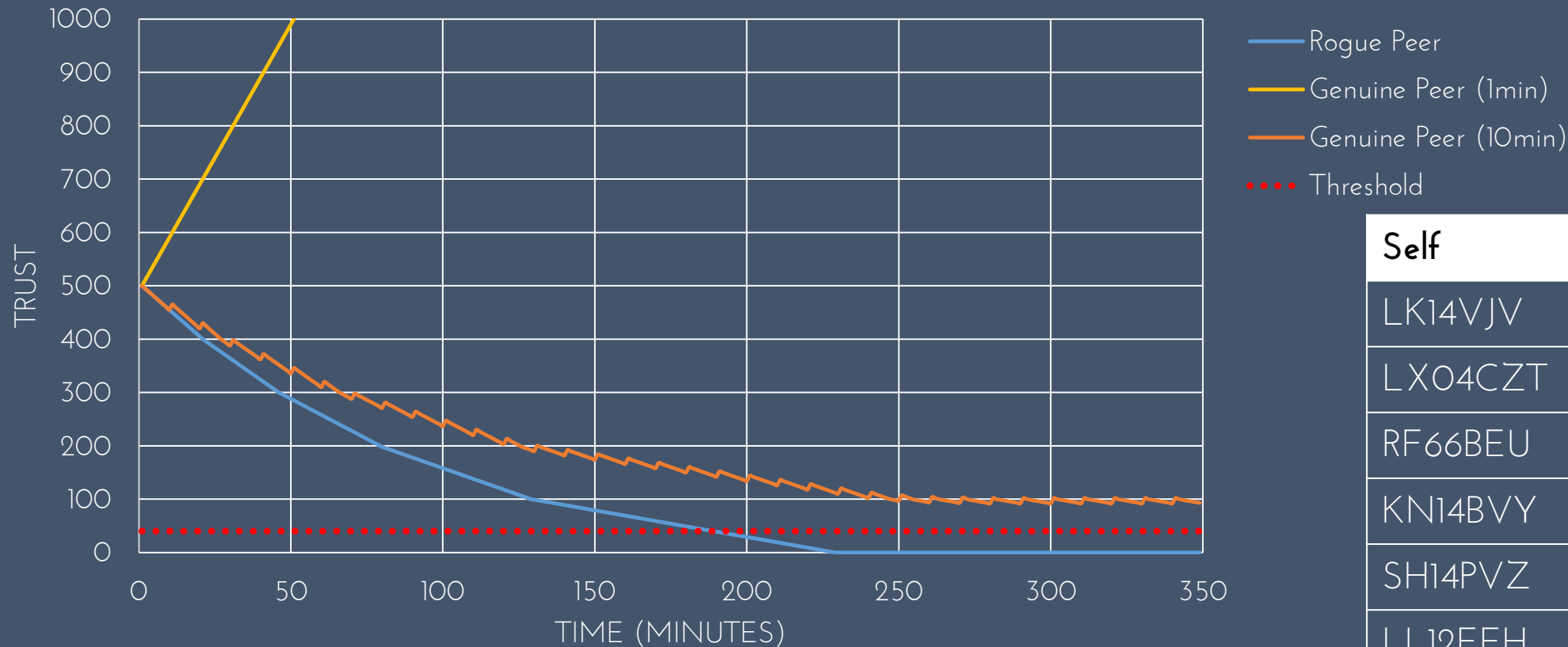


Self	Peer (10min)
LK14VJV	LK14VJV
LX04CZT	
RF66BEU	RF66BEU
KN14BVY	KN14BVY
SH14PVZ	
LL12EFH	

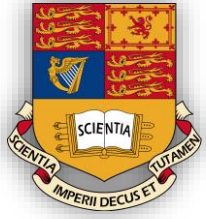
# Peer to Peer Network - Trust system



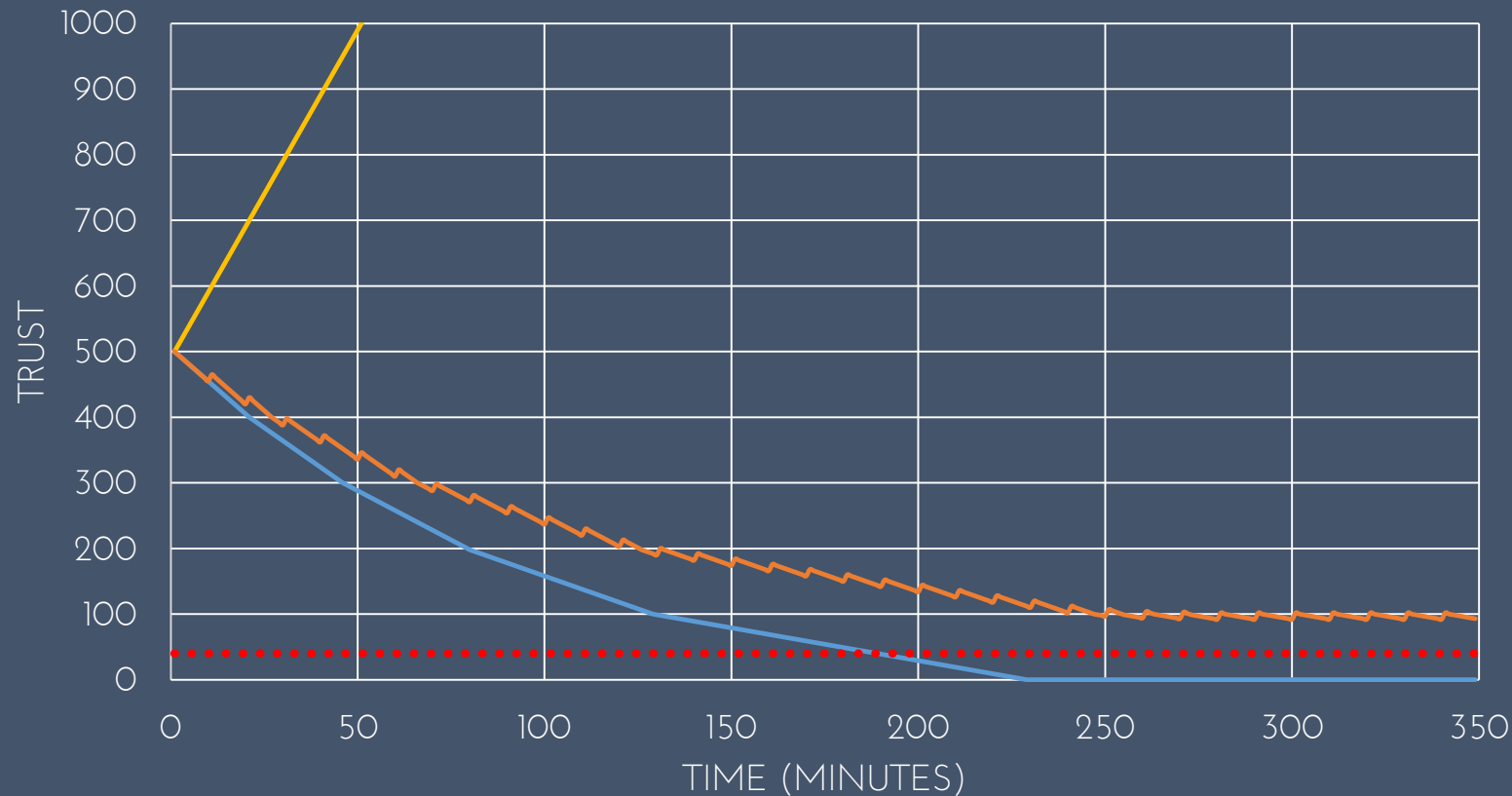
TRUST DECAY (0.99)



# Peer to Peer Network - Trust system

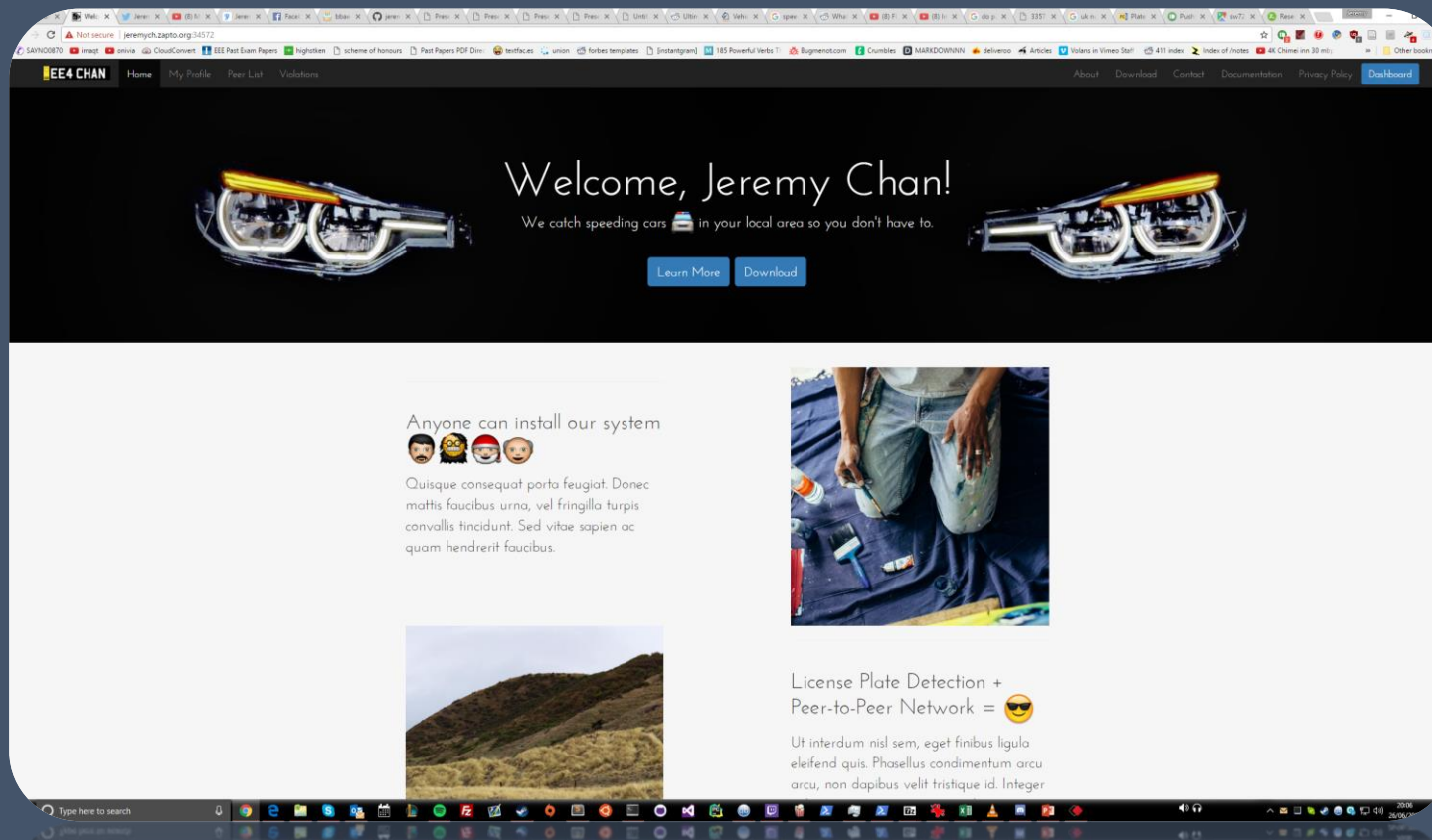


TRUST DECAY (0.99)



- Ignore rogue
- Reward genuine, matching plates
- Prevent fake plates
- Prevent leeching

# Web Interface - Aim



- What the user sees
- Dashboard to control network
- Post detected violations on social networks

# Demo



## Modifications

1. Pre-recorded video
2. Live motion detection unused



# Evaluation



- ✓ 1. Implement a number plate recognition system using existing computer vision algorithms on a low-cost, readily available hardware platform.
- ✓ 2. Set up a peer-to-peer network to share vehicle passing times and detect violations without the need for a central server\*.
- ✓ 3. Publish photo evidence of any violations

• All main deliverables achieved

\*) Except when bootstrapping

# Evaluation



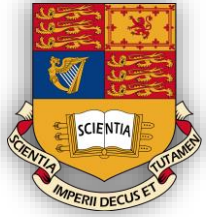
- ✓ ▪ Have a **standalone mode** in case there is no one in the network
- ✓ ▪ Be able to **join/leave/return to** the network **easily**
- ✓ ▪ Make the network **secure** against **rogue peers**
- **Package** the system so that it can be **easily installed** in a home by an inexperienced user

- All but 1 advanced deliverable achieved

# Conclusion



- Challenging project
- Very open ended on implementation
- Usable immediately
- Revisit performance and night time issues with LPR
- Consider refactoring P2P to be standalone
- Scalability testing
- Attempt overnight test



# Thank you

<http://github.com/jeremychn1000/ee4-FYP/>

©2017 Jeremy Chan. All rights reserved. | Powered by Django and OpenALPR | Created with love from the United Kingdom