



The Client shall have a state machine (text) editor.

The Client shall have a field displaying the log.

The Client shall have a diagram for each type of data (cars stopped, pedestrian signal)

The Server shall be able to receive a state machine from the user.

The Server shall translate the received state machine into a runnable program.

The Server shall be able to receive data from the controller in the form *key: value*.

The Server shall log the received data to the database.

The Server shall be able to receive events with parameters from the user.

The Server shall forward events with parameters to the controller.

The Server shall authenticate users based on usernames and passwords.

The Server shall only allow *police* users to send an event.

The Server shall allow user creation from the user *admin*.

The mailbox API shall provide the following features:

- sendData(key, value)
- onData(key, handler)

The Server shall call the data handlers every 5s, if any data has accumulated in that time.

A user shall be able to upload user software to the controller.

A user shall be able to start and stop user software.

The user software shall have access to the following hardware features:

- Traffic light **out**
- Pedestrian signal **in**
- Cars stopped meter **in**

The user software shall have access to the following software features:

- Police Interrupt **in**
- State set **in**

The control server shall publish measured hardware features to the server component.

The traffic light API shall provide the following features:

- setRed
- setYellow
- setGreen
- setPedRed
- setPedGreen
- testRed
- testYellow
- testGreen
- testPedRed
- testPedGreen
- getPedestrianSignal
- getStoppedCars

The mailbox API shall provide the following features:

- sendData(key, value)
- onData(key, handler)