Requirement Report - Details

Version ●

CREATED WITH ENTERPRISE ARCHITECT

Traffic Light System Model

Package in package ''

Traffic Light System Model
Version 1.0 Phase 1.0 Proposed
chioma created on 5/23/2022. Last modified 5/23/2022

Index

Package in package 'Traffic Light System Model'

Index
Version 1.0 Phase 1.0 Proposed
chioma created on 2/19/2022. Last modified 2/19/2022

Index diagram

Custom Diagram Style diagram in package 'Index'

Index
Version 1.0
chioma created on 2/19/2022. Last modified 5/22/2022

Figure 1: Index
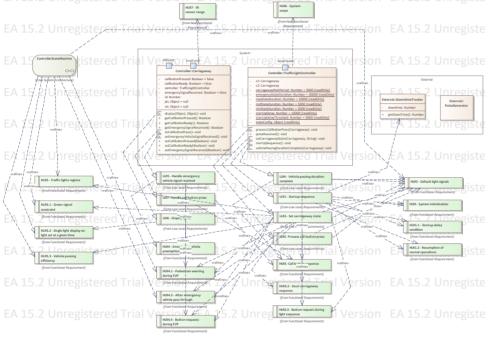
Design

Package in package 'Traffic Light System Model'

<div align="right">

Design
Version 1.0 Phase 1.0 Proposed
chioma created on 5/5/2022. Last modified 5/5/2022

</div>

Relationship Map diagram

Custom diagram in package 'Design'

<div align="right">

Relationship Map
Version 1.0
chioma created on 5/22/2022. Last modified 5/23/2022

</div>

Figure 2: Relationship Map

Low Level Requirements

Package in package 'Design'

<div align="right">

Low Level Requirements
Version 1.0 Phase 1.0 Proposed

</div>

Low Level Requirements diagram

Requirements diagram in package 'Low Level Requirements'

Low Level Requirements
Version 1.0
chioma created on 5/4/2022. Last modified 5/21/2022

Figure 3: Low Level Requirements

LLR1 - Startup sequence

Requirement «Functional» in package 'Low Level Requirements'

This operation gets the downtime (in milliseconds) from the external system, DowntimeTracker. If the downtime is less than the configured startup delay threshold, it waits for the configured startup delay before changing the state to 'vehiclesPassingReady'. Otherwise, if the downtime is greater than the configured startup delay threshold, it changes the state immediately to 'vehiclesPassingReady'. This operation returns no value.

LLR1 - Startup sequence
Version 1.0 Phase 1.0 Proposed
chioma created on 5/21/2022. Last modified 5/22/2022

INCOMING STRUCTURAL RELATIONSHIPS
⇒ Realization from TrafficLightController to «Functional» LLR1 - Startup sequence [Direction is 'Source -> Destination'.]

CONNECTORS
↗ **Abstraction** «Refine»   Source -> Destination From: LLR1 - Startup sequence : Requirement, Public To: HLR1.1 - Startup delay condition : Requirement, Public

CONNECTORS

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: HLR1 - System initialization : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: HLR1.2 - Resumption of normal operations : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: HLR1.1 - Startup delay condition : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: HLR1 - System initialization : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: HLR1.2 - Resumption of normal operations : Requirement, Public

↗ **Dependency**   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: DowntimeTracker : Class, Public

LLR2 - Set carriageway state

Requirement «Functional» in package 'Low Level Requirements'

The operation accepts as parameters the carriageway on which the state is to be changed and a string indicating the state. It sets the callButtonReady flag to false, indicating that any button presses should be ignored for the time being. It also retrieves the state's properties (i.e. colour and flashing states of the vehicle and pedestrian lights, duration of the state, next state, next action) from the state config object and then invokes the carriageway display function to display the appropriate lights. After this, it delays for the time specified in the state's duration attribute and then gets the emergencySignalReceived flag from the carriageway class. If the flag is true, and the state is not 'pedestriansCrossing' and 'pedestriansCrossingAlmostDone', then it performs the following operations:

- Sets the emergencySignalReceived flag to false
- Sets the nextState property to 'emergencyVehicleApproaching'
- Sets the nextAction property to false

If the nextState property exists, it calls itself with the carriageway argument and the nextState. However, if the nextAction property exists, indicating that the "vehiclesPassing" state duration is complete, the operation vehiclePassingDurationComplete is called. The operation returns no value.

LLR2 - Set carriageway state
Version 1.0 Phase 1.0 Proposed
chioma created on 5/5/2022. Last modified 5/22/2022

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Realization from TrafficLightController to «Functional» LLR2 - Set carriageway state

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.3 - After emergency vehicle pass through : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR5 - Traffic lights regime : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR3.3 - Button request during light sequence : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.3 - After emergency vehicle pass through : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR3.3 - Button request during light sequence : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.4 - Button requests during EVP : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR5.3 - Vehicle passing efficiency : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.2 - EVP state transition : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.1 - Pedestrian warning during EVP : Requirement, Public

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR5.1 - Green signal constraint : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.4 - Button requests during EVP : Requirement, Public

---

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.1 - Pedestrian warning during EVP : Requirement, Public

---

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

LLR3 - Process call button press

Requirement «Functional» in package 'Low Level Requirements'

This operation accepts the carriageway on which the call button press is to be processed as a parameter. It gets the callButtonPressed flag from the carriageway class. If the callButtonPressed flag is true, it performs the following operations:

- It changes the callButtonPressed flag to false
- Because the carriageway whose call button was pressed is passed as an argument, it gets a reference to the 'other' carriageway.
- It invokes the setCarriagewayState function with the state 'vehiclesPassingSlowDown' for the carriageway passed in.
- It calls the setCarriagewayState function for the 'other' carriageway after a defined carriagewayWaitPeriod.

If the callButtonPressed flag is false, indicating that there are no button presses to process, no action is taken. This operation is only performed when the state, vehiclePassing's wait duration has expired, or after the system has received a call button press and the callButtonReady flag is set to true. This operation returns no value.

LLR3 - Process call button press
Version 1.0 Phase 1.0 Proposed
chioma created on 5/21/2022. Last modified 5/22/2022

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Realization from TrafficLightController to «Functional» LLR3 - Process call button press
[Direction is 'Source -> Destination'.]

CONNECTORS

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR3 - Process call button press : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR3 - Process call button press : Requirement, Public
To: HLR3.2 - Dual carriageway response : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR3 - Process call button press : Requirement, Public
To: HLR3.2 - Dual carriageway response : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR3 - Process call button press : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

Call Button Event Sequence diagram

Interaction diagram in package 'Low Level Requirements'

Call Button Event Sequence
Version 1.0
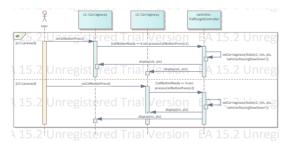chioma created on 5/5/2022. Last modified 5/22/2022

Figure 4: Call Button Event Sequence

LLR4 - Vehicle passing duration complete

Requirement «Functional» in package 'Low Level Requirements'

This operation accepts the carriageway on which the call button press is to be processed as a parameter. It sets the callButtonReady flag to true, which indicates that the minimum wait period for vehicle passing has elapsed. It also invokes the processCallButtonPress function passing in a reference to the carriageway on which the call button was pressed. This operation returns no value.

LLR4 - Vehicle passing duration complete
Version 1.0 Phase 1.0 Proposed

chioma created on 5/21/2022. Last modified 5/22/2022

INCOMING STRUCTURAL RELATIONSHIPS

➡ Realization from TrafficLightController to «Functional» LLR4 - Vehicle passing duration complete

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR4 - Vehicle passing duration complete : Requirement, Public
To: HLR3 - Call button response : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR4 - Vehicle passing duration complete : Requirement, Public
To: HLR3 - Call button response : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR4 - Vehicle passing duration complete : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR4 - Vehicle passing duration complete : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

LLR5 - Handle emergency vehicle signal received

Requirement «Functional» in package 'Low Level Requirements'

When invoked, this operation sets the emergencySignalReceived flag to true. It returns no value. It is important to note that when the controller is changing the carriageway state, it checks to see if the emergencySignalReceived flag is true. If it is, it changes the state to the 'emergencyVehicleApproachingState'.

LLR5 - Handle emergency vehicle signal received
Version 1.0 Phase 1.0 Proposed
chioma created on 5/21/2022. Last modified 5/22/2022

INCOMING STRUCTURAL RELATIONSHIPS

➡ Realization from Carriageway to «Functional» LLR5 - Handle emergency vehicle signal received

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR5 - Handle emergency vehicle signal received : Requirement, Public
To: HLR4.4 - Button requests during EVP : Requirement, Public

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR5 - Handle emergency vehicle signal received : Requirement, Public
To: HLR4.2 - EVP state transition : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR5 - Handle emergency vehicle signal received : Requirement, Public
To: HLR4 - Emergency vehicle preemption (EVP) : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR5 - Handle emergency vehicle signal received : Requirement, Public
To: HLR4 - Emergency vehicle preemption (EVP) : Requirement, Public

LLR6 - Display lights

Requirement «Functional» in package 'Low Level Requirements'

This operation accepts the colour and flashing state configuration for the vehicle and pedestrian lights on a specific carriageway. It then changes the current colour and flashing effect state to the new one. It returns no value.

LLR6 - Display lights
Version 1.0 Phase 1.0 Proposed
chioma created on 5/21/2022. Last modified 5/22/2022

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Realization from Carriageway to «Functional» LLR6 - Display lights
[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR6 - Display lights : Requirement, Public
To: HLR5.2 - Single light display on light set at a given time : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR6 - Display lights : Requirement, Public
To: HLR4.1 - Pedestrian warning during EVP : Requirement, Public

LLR7 - Handle call button press

Requirement «Functional» in package 'Low Level Requirements'

When invoked, this operation sets the callButtonPressed flag to true. If the callButtonReady flag is true, it then proceeds to call the processCallButtonPress function on the controller. It passes a reference to itself as an argument. If the callButtonReady flag is false, it does nothing. This operation returns no value.

LLR7 - Handle call button press
Version 1.0 Phase 1.0 Proposed
chioma created on 5/21/2022. Last modified 5/22/2022

INCOMING STRUCTURAL RELATIONSHIPS

➡ Realization from Carriageway to «Functional» LLR7 - Handle call button press

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR7 - Handle call button press : Requirement, Public
To: HLR3 - Call button response : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR7 - Handle call button press : Requirement, Public
To: HLR3.1 - Pedestrian crossing wait time : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR7 - Handle call button press : Requirement, Public
To: HLR3 - Call button response : Requirement, Public

System Architecture

Package in package 'Design'

<div align="right">

System Architecture
Version 1.0 Phase 1.0 Proposed
chioma created on 5/4/2022. Last modified 5/5/2022

</div>

Controller

Package in package 'System Architecture'

<div align="right">

Controller
Version 1.0 Phase 1.0 Proposed
chioma created on 5/4/2022. Last modified 5/4/2022

</div>

Controller diagram

Class diagram in package 'Controller'

<div align="right">

Controller
Version 1.0
chioma created on 5/4/2022. Last modified 5/21/2022

</div>

Figure 5: Controller

External

Package in package 'System Architecture'

<div align="right">

External
Version 1.0 Phase 1.0 Proposed
chioma created on 5/4/2022. Last modified 5/4/2022

</div>

External diagram

Class diagram in package 'External'

<div align="right">

External
Version 1.0
chioma created on 5/4/2022. Last modified 5/21/2022

</div>

Figure 6: External

High Level Requirements

Package in package 'Traffic Light System Model'

High Level Requirements
Version 1.0 Phase 1.0 Proposed
chioma created on 2/19/2022. Last modified 5/4/2022

Functional Requirement

Package in package 'High Level Requirements'

Functional Requirement
Version 1.0 Phase 1.0 Proposed
chioma created on 5/4/2022. Last modified 5/4/2022

Functional Requirement Hierarchy diagram

Requirements diagram in package 'Functional Requirement'

Functional Requirement Hierarchy
Version 1.0
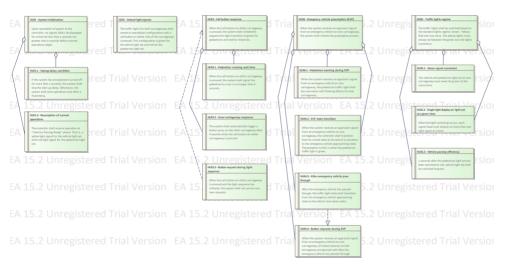chioma created on 2/19/2022. Last modified 5/4/2022

Figure 7: Functional Requirement Hierarchy

HLR1 - System initialization

Requirement «Functional» in package 'Functional Requirement'

Upon resumption of power to the controller, no signals SHALL be displayed for a time not less than x

seconds nor greater than y seconds before normal operations begin.

HLR1 - System initialization
Version 1.0 Phase 1.0 Proposed
chioma created on 2/19/2022. Last modified 5/21/2022

ELEMENTS OWNED BY HLR1 - System initialization

🗎 HLR1.1 - Startup delay condition : Requirement «Functional»

If the system has already been turned off for more than x seconds, the system shall skip the start up delay. Otherwise, the system shall start operations only after a fixed delay.

🗎 HLR1.2 - Resumption of normal operations : Requirement «Functional»

The controller shall resume operation at "Vehicles Passing Ready" phase. That is, a yellow light signal for the vehicle light set and a red light signal for the pedestrian light set.

INCOMING STRUCTURAL RELATIONSHIPS

➡ Aggregation from «Functional» HLR1.1 - Startup delay condition to «Functional» HLR1 - System initialization

[Direction is 'Source -> Destination'.]

➡ Aggregation from «Functional» HLR1.2 - Resumption of normal operations to «Functional» HLR1 - System initialization

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: HLR1 - System initialization : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: HLR1 - System initialization : Requirement, Public

HLR1.1 - Startup delay condition

Requirement «Functional» owned by 'HLR1 - System initialization', in package 'Functional Requirement'

If the system has already been turned off for more than x seconds, the system shall skip the start up delay. Otherwise, the system shall start operations only after a fixed delay.

HLR1.1 - Startup delay condition
Version 1.0 Phase 1.0 Proposed
chioma created on 2/19/2022. Last modified 5/13/2022

OUTGOING STRUCTURAL RELATIONSHIPS

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Aggregation from «Functional» HLR1.1 - Startup delay condition to «Functional» HLR1 - System initialization

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: HLR1.1 - Startup delay condition : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: HLR1.1 - Startup delay condition : Requirement, Public

HLR1.2 - Resumption of normal operations

Requirement «Functional» owned by 'HLR1 - System initialization', in package 'Functional Requirement'

The controller shall resume operation at "Vehicles Passing Ready" phase. That is, a yellow light signal for the vehicle light set and a red light signal for the pedestrian light set.

HLR1.2 - Resumption of normal operations
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Aggregation from «Functional» HLR1.2 - Resumption of normal operations to «Functional» HLR1 - System initialization

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: HLR1.2 - Resumption of normal operations : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR1 - Startup sequence : Requirement, Public
To: HLR1.2 - Resumption of normal operations : Requirement, Public

HLR2 - Default light signals

Requirement «Functional» in package 'Functional Requirement'

The traffic lights for both carriageways shall remain in one default configuration until a call button on either

side of the carriageway is pressed. This configuration is green for the vehicle light set and red for the
pedestrian light set.

<div align="right">

HLR2 - Default light signals
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR3 - Process call button press : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR4 - Vehicle passing duration complete : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR4 - Vehicle passing duration complete : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR3 - Process call button press : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: ControllerStateMachine : StateMachine, Public
To: HLR2 - Default light signals : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR2 - Default light signals : Requirement, Public

HLR3 - Call button response

Requirement «Functional» in package 'Functional Requirement'

When the call button on either carriageway is pressed, the system shall initiate the sequence for light
transition to green for pedestrians and red for motorists.

<div align="right">

HLR3 - Call button response
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

ELEMENTS OWNED BY HLR3 - Call button response

ELEMENTS OWNED BY HLR3 - Call button response

▤ HLR3.1 - Pedestrian crossing wait time : Requirement «Functional»

When the call button on either carriageway is pressed, the system shall signal the pedestrian to cross in no longer than X seconds.

▤ HLR3.2 - Dual carriageway response : Requirement «Functional»

The system shall automatically trigger a button press on the other carriageway after X seconds when the call button on either carriageway is pressed

▤ HLR3.3 - Button request during light sequence : Requirement «Functional»

When the call button on either carriageway is pressed and the light sequence has initiated, the system shall not service any new requests.

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Aggregation from «Functional» HLR3.1 - Pedestrian crossing wait time to «Functional» HLR3 - Call button response

[Direction is 'Source -> Destination'.]

⇒ Aggregation from «Functional» HLR3.2 - Dual carriageway response to «Functional» HLR3 - Call button response

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR4 - Vehicle passing duration complete : Requirement, Public
To: HLR3 - Call button response : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR7 - Handle call button press : Requirement, Public
To: HLR3 - Call button response : Requirement, Public

↗ **Dependency**   Source -> Destination
From: HLR3.3 - Button request during light sequence : Requirement, Public
To: HLR3 - Call button response : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR4 - Vehicle passing duration complete : Requirement, Public
To: HLR3 - Call button response : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR7 - Handle call button press : Requirement, Public
To: HLR3 - Call button response : Requirement, Public

HLR3.1 - Pedestrian crossing wait time

Requirement «Functional» owned by 'HLR3 - Call button response', in package 'Functional Requirement'

When the call button on either carriageway is pressed, the system shall signal the pedestrian to cross in no longer than X seconds.

<div align="right">

HLR3.1 - Pedestrian crossing wait time
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Aggregation from «Functional» HLR3.1 - Pedestrian crossing wait time to «Functional» HLR3 - Call button response

<div align="right">[Direction is 'Source -> Destination'.]</div>

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR7 - Handle call button press : Requirement, Public
To: HLR3.1 - Pedestrian crossing wait time : Requirement, Public

---

HLR3.2 - Dual carriageway response

Requirement «Functional» owned by 'HLR3 - Call button response', in package 'Functional Requirement'

The system shall automatically trigger a button press on the other carriageway after X seconds when the call button on either carriageway is pressed

<div align="right">

HLR3.2 - Dual carriageway response
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Aggregation from «Functional» HLR3.2 - Dual carriageway response to «Functional» HLR3 - Call button response

<div align="right">[Direction is 'Source -> Destination'.]</div>

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR3 - Process call button press : Requirement, Public
To: HLR3.2 - Dual carriageway response : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR3 - Process call button press : Requirement, Public
To: HLR3.2 - Dual carriageway response : Requirement, Public

HLR3.3 - Button request during light sequence

Requirement «Functional» owned by 'HLR3 - Call button response', in package 'Functional Requirement'

When the call button on either carriageway is pressed and the light sequence has initiated, the system shall not service any new requests.

<div align="right">

HLR3.3 - Button request during light sequence
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

CONNECTORS
↗ **Dependency**   Source -> Destination   From: HLR3.3 - Button request during light sequence : Requirement, Public   To: HLR3 - Call button response : Requirement, Public
↗ **Refine** «refine»   Source -> Destination   From: LLR2 - Set carriageway state : Requirement, Public   To: HLR3.3 - Button request during light sequence : Requirement, Public
↗ **Abstraction** «Refine»   Source -> Destination   From: LLR2 - Set carriageway state : Requirement, Public   To: HLR3.3 - Button request during light sequence : Requirement, Public

HLR4 - Emergency vehicle preemption (EVP)

Requirement «Functional» in package 'Functional Requirement'

When the system receives an approach signal from an emergency vehicle on one carriageway, the system shall initiate the preemption process

<div align="right">

HLR4 - Emergency vehicle preemption (EVP)
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

ELEMENTS OWNED BY HLR4 - Emergency vehicle preemption (EVP)
📄 HLR4.1 - Pedestrian warning during EVP : Requirement «Functional»   When the system receives an approach signal from an emergency vehicle on one carriageway, the pedestrian traffic light shall be a red colour with flashing effects for that carriageway.
📄 HLR4.2 - EVP state transition : Requirement «Functional»   When the system receives an approach signal from an emergency vehicle on one carriageway, the controller shall transition from its current state at the end of its duration to the emergency vehicle approaching state. The exception to this is when the pedestrian traffic light is green.

ELEMENTS OWNED BY HLR4 - Emergency vehicle preemption (EVP)

▣ HLR4.3 - After emergency vehicle pass through : Requirement «Functional»

After the emergency vehicle has passed through, the traffic light state shall transition from the emergency vehicle approaching state to the vehicle slow down state.

▣ HLR4.4 - Button requests during EVP : Requirement «Functional»

When the system receives an approach signal from an emergency vehicle on one carriageway, all button presses on that carriageway are ignored until after the emergency vehicle has passed through.

INCOMING STRUCTURAL RELATIONSHIPS

➡ Aggregation from «Functional» HLR4.3 - After emergency vehicle pass through to «Functional» HLR4 - Emergency vehicle preemption (EVP)

[Direction is 'Source -> Destination'.]

➡ Aggregation from «Functional» HLR4.4 - Button requests during EVP to «Functional» HLR4 - Emergency vehicle preemption (EVP)

[Direction is 'Source -> Destination'.]

➡ Aggregation from «Functional» HLR4.2 - EVP state transition to «Functional» HLR4 - Emergency vehicle preemption (EVP)

[Direction is 'Source -> Destination'.]

➡ Aggregation from «Functional» HLR4.1 - Pedestrian warning during EVP to «Functional» HLR4 - Emergency vehicle preemption (EVP)

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR5 - Handle emergency vehicle signal received : Requirement, Public
To: HLR4 - Emergency vehicle preemption (EVP) : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR5 - Handle emergency vehicle signal received : Requirement, Public
To: HLR4 - Emergency vehicle preemption (EVP) : Requirement, Public

HLR4.1 - Pedestrian warning during EVP

Requirement «Functional» owned by 'HLR4 - Emergency vehicle preemption (EVP)', in package 'Functional Requirement'

When the system receives an approach signal from an emergency vehicle on one carriageway, the pedestrian traffic light shall be a red colour with flashing effects for that carriageway.

HLR4.1 - Pedestrian warning during EVP

Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/22/2022

OUTGOING STRUCTURAL RELATIONSHIPS

⇐ Aggregation from «Functional» HLR4.1 - Pedestrian warning during EVP to «Functional» HLR4 - Emergency vehicle preemption (EVP)

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR6 - Display lights : Requirement, Public
To: HLR4.1 - Pedestrian warning during EVP : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.1 - Pedestrian warning during EVP : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: ControllerStateMachine : StateMachine, Public
To: HLR4.1 - Pedestrian warning during EVP : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.1 - Pedestrian warning during EVP : Requirement, Public

HLR4.2 - EVP state transition

Requirement «Functional» owned by 'HLR4 - Emergency vehicle preemption (EVP)', in package 'Functional Requirement'

When the system receives an approach signal from an emergency vehicle on one carriageway, the controller shall transition from its current state at the end of its duration to the emergency vehicle approaching state. The exception to this is when the pedestrian traffic light is green.

HLR4.2 - EVP state transition
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/22/2022

OUTGOING STRUCTURAL RELATIONSHIPS

⇐ Aggregation from «Functional» HLR4.2 - EVP state transition to «Functional» HLR4 - Emergency vehicle preemption (EVP)

[Direction is 'Source -> Destination'.]

CONNECTORS

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR5 - Handle emergency vehicle signal received : Requirement, Public
To: HLR4.2 - EVP state transition : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.2 - EVP state transition : Requirement, Public

HLR4.3 - After emergency vehicle pass through

Requirement «Functional» owned by 'HLR4 - Emergency vehicle preemption (EVP)', in package 'Functional Requirement'

After the emergency vehicle has passed through, the traffic light state shall transition from the emergency vehicle approaching state to the vehicle slow down state.

HLR4.3 - After emergency vehicle pass through
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/22/2022

EXTERNAL REQUIREMENTS

☑ Requirement. HLR4.4 - Button requests during EVP

When the system receives an approach signal from an emergency vehicle on one carriageway, all button presses on that carriageway are ignored until after the emergency vehicle has passed through.

[Stereotype is «Functional».]

OUTGOING STRUCTURAL RELATIONSHIPS

⇐ Realization from «Functional» HLR4.3 - After emergency vehicle pass through to «Functional» HLR4.4 - Button requests during EVP

[Direction is 'Source -> Destination'.]

⇐ Aggregation from «Functional» HLR4.3 - After emergency vehicle pass through to «Functional» HLR4 - Emergency vehicle preemption (EVP)

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.3 - After emergency vehicle pass through : Requirement, Public

---

↗ **Abstraction** «Refine»   Source -> Destination
From: ControllerStateMachine : StateMachine, Public
To: HLR4.3 - After emergency vehicle pass through : Requirement, Public

---

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.3 - After emergency vehicle pass through : Requirement, Public

---

HLR4.4 - Button requests during EVP

Requirement «Functional» owned by 'HLR4 - Emergency vehicle preemption (EVP)', in package 'Functional Requirement'

When the system receives an approach signal from an emergency vehicle on one carriageway, all button presses on that carriageway are ignored until after the emergency vehicle has passed through.

<div align="right">

HLR4.4 - Button requests during EVP
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/22/2022

</div>

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Aggregation from «Functional» HLR4.4 - Button requests during EVP to «Functional» HLR4 - Emergency vehicle preemption (EVP)

<div align="right">[Direction is 'Source -> Destination'.]</div>

INCOMING STRUCTURAL RELATIONSHIPS

➡ Realization from «Functional» HLR4.3 - After emergency vehicle pass through to «Functional» HLR4.4 - Button requests during EVP

<div align="right">[Direction is 'Source -> Destination'.]</div>

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR5 - Handle emergency vehicle signal received : Requirement, Public
To: HLR4.4 - Button requests during EVP : Requirement, Public

---

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.4 - Button requests during EVP : Requirement, Public

---

↗ **Abstraction** «Refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR4.4 - Button requests during EVP : Requirement, Public

---

HLR5 - Traffic lights regime

Requirement «Functional» in package 'Functional Requirement'

The traffic lights shall be switched based on the standard lights regime: Green - Yellow - Red and vice versa. The yellow lights must always be between the green and red lights transitions.

<div align="right">

HLR5 - Traffic lights regime
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

ELEMENTS OWNED BY HLR5 - Traffic lights regime

▦ HLR5.1 - Green signal constraint : Requirement «Functional»

The vehicle and pedestrian light set on one carriageway must never be green at the same time.

▦ HLR5.2 - Single light display on light set at a given time : Requirement «Functional»

When the light switching occurs, each signal head must display no more than one light signal at a time.

▦ HLR5.3 - Vehicle passing efficiency : Requirement «Functional»

x seconds after the pedestrian light set has been switched to red, vehicle light set shall be switched to green.

INCOMING STRUCTURAL RELATIONSHIPS

⇒ Aggregation from «Functional» HLR5.2 - Single light display on light set at a given time to «Functional» HLR5 - Traffic lights regime

<div align="right">[Direction is 'Source -> Destination'.]</div>

⇒ Aggregation from «Functional» HLR5.3 - Vehicle passing efficiency to «Functional» HLR5 - Traffic lights regime

<div align="right">[Direction is 'Source -> Destination'.]</div>

⇒ Aggregation from «Functional» HLR5.1 - Green signal constraint to «Functional» HLR5 - Traffic lights regime

<div align="right">[Direction is 'Source -> Destination'.]</div>

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR5 - Traffic lights regime : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: ControllerStateMachine : StateMachine, Public
To: HLR5 - Traffic lights regime : Requirement, Public

HLR5.1 - Green signal constraint

Requirement «Functional» owned by 'HLR5 - Traffic lights regime', in package 'Functional Requirement'

The vehicle and pedestrian light set on one carriageway must never be green at the same time.

HLR5.1 - Green signal constraint
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Aggregation from «Functional» HLR5.1 - Green signal constraint to «Functional» HLR5 - Traffic lights regime

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Abstraction** «Refine»   Source -> Destination
From: ControllerStateMachine : StateMachine, Public
To: HLR5.1 - Green signal constraint : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR5.1 - Green signal constraint : Requirement, Public

HLR5.2 - Single light display on light set at a given time

Requirement «Functional» owned by 'HLR5 - Traffic lights regime', in package 'Functional Requirement'

When the light switching occurs, each signal head must display no more than one light signal at a time.

HLR5.2 - Single light display on light set at a given time
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

OUTGOING STRUCTURAL RELATIONSHIPS

⬅ Aggregation from «Functional» HLR5.2 - Single light display on light set at a given time to «Functional» HLR5 - Traffic lights regime

[Direction is 'Source -> Destination'.]

CONNECTORS

↗ **Refine** «refine»   Source -> Destination
From: LLR6 - Display lights : Requirement, Public
To: HLR5.2 - Single light display on light set at a given time : Requirement, Public

↗ **Abstraction** «Refine»   Source -> Destination
From: ControllerStateMachine : StateMachine, Public
To: HLR5.2 - Single light display on light set at a given time : Requirement, Public

HLR5.3 - Vehicle passing efficiency

Requirement «Functional» owned by 'HLR5 - Traffic lights regime', in package 'Functional Requirement'

x seconds after the pedestrian light set has been switched to red, vehicle light set shall be switched to green.

<div align="right">

HLR5.3 - Vehicle passing efficiency
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

OUTGOING STRUCTURAL RELATIONSHIPS

⇐ Aggregation from «Functional» HLR5.3 - Vehicle passing efficiency to «Functional» HLR5 - Traffic lights regime

<div align="right">

[Direction is 'Source -> Destination'.]

</div>

CONNECTORS

↗ **Abstraction** «Refine»   Source -> Destination
From: ControllerStateMachine : StateMachine, Public
To: HLR5.3 - Vehicle passing efficiency : Requirement, Public

↗ **Refine** «refine»   Source -> Destination
From: LLR2 - Set carriageway state : Requirement, Public
To: HLR5.3 - Vehicle passing efficiency : Requirement, Public

Non-Functional Requirement

Package in package 'High Level Requirements'

<div align="right">

Non-Functional Requirement
Version 1.0 Phase 1.0 Proposed
chioma created on 5/4/2022. Last modified 5/4/2022

</div>

Non-Functional Requirement Hierarchy diagram

Requirements diagram in package 'Non-Functional Requirement'

<div align="right">

Non-Functional Requirement Hierarchy
Version 1.0
chioma created on 2/20/2022. Last modified 5/23/2022

</div>

Figure 8: Non-Functional Requirement Hierarchy

HLR10 - System recovery after failure

NonfunctionalRequirement «NonfunctionalRequirement» in package 'Non-Functional Requirement'

The system shall recover normal operations after a system failure in no more than X seconds.

<div align="right">

HLR10 - System recovery after failure
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

HLR11 - Fail-safe initialization

NonfunctionalRequirement «NonfunctionalRequirement» in package 'Non-Functional Requirement'

The system shall initiate fail-safe operation during system failures in no more than X seconds.

<div align="right">

HLR11 - Fail-safe initialization
Version 1.0 Phase 1.0 Proposed

</div>

chioma created on 2/20/2022. Last modified 5/13/2022

HLR6 - System scope

ArchitecturalRequirement «ArchitecturalRequirement» in package 'Non-Functional Requirement'

The system's scope shall consist of a single controller that controls a pair of signal heads for motorists and a pair of signal heads for pedestrians on both carriageways

<div align="right">

HLR6 - System scope
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

CONNECTORS
↗ **Abstraction** «Refine»   Source -> Destination From: TrafficLightController : Class, Public To: HLR6 - System scope : ArchitecturalRequirement, Public

HLR7 - IR sensor range

ArchitecturalRequirement «ArchitecturalRequirement» in package 'Non-Functional Requirement'

The system shall detect infrared signals through passive infrared sensors within a minimum range of X meters.

<div align="right">

HLR7 - IR sensor range
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

CONNECTORS
↗ **Abstraction** «Refine»   Source -> Destination From: Carriageway : Class, Public To: HLR7 - IR sensor range : ArchitecturalRequirement, Public

HLR8 - System response time

NonfunctionalRequirement «NonfunctionalRequirement» in package 'Non-Functional Requirement'

The system shall perform commands issued to it in no more than X milliseconds.

<div align="right">

HLR8 - System response time
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

HLR9 - System availability

NonfunctionalRequirement «NonfunctionalRequirement» in package 'Non-Functional Requirement'

The system shall aim for 99.999% availability on a yearly basis i.e the system can only be down for only 6 minutes per year.

<div align="right">

HLR9 - System availability
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>

NonfunctionalRequirement «NonfunctionalRequirement» in package 'Non-Functional Requirement'

<div align="right">

HLR9 - System availability
Version 1.0 Phase 1.0 Proposed
chioma created on 2/20/2022. Last modified 5/13/2022

</div>