**Documentation**

*The Invincible*

*07..07.2021*

***Important Notes:***

* ***The descriptions in italics in this document (except for some section headings) are exemplary and explanatory and must be removed from the completed report.***
* ***Identify which section of this report was created by which team member***

# Team members

Listing of the individual team members

# Introduction

What is the Truck Platooning Simulation about?

# Product description

Our project consists of three main parts, which are communication, formation of platooning and driving. For each of them, we have planned some simulation.

These are some of the communications scenarios that might be simulated:

* Trucks can share information for their location and real-time positions.
* Trucks can share information for general street conditions, especially in difficult weather conditions.
* Trucks can share information for an emergency caused by external factors in a specific part of the street (Working canter, Infrastructure damage etc.)
* Trucks can share information for their directions and trajectory so the driving maneuvers can be done smoother.

List of processes that might be simulated in platooning:

* Trucks can join the platooning as a follower as well as a leader.
* At a specific point in time, a truck might need to leave the platooning.
* The leader selection is made by using a specific election algorithm.

List of processes that might be simulated in the driving domain.

* A brake simulation is performed when a user through the main controller gives this command.
* An accelerates simulation is performed when a user through the main controller gives this command.
* A full stop simulation is performed when a user through the main controller gives this command.

# Technologies

# *Describe the technological approaches you will use to implement your project.*

* *architecture patterns*
* *programming languages*
* **Programming Languages**
* All microservices will be programmed in C++ 11 or later versions.
* Local server side and querying will be programmed with JSon.
* In a dynamic web page, we might use HTML, JavaScript, CSS
* *frameworks*
* *DevOps tools*
* *...*

# Use-Cases

*Describe the use cases of your simulation environment with a use case diagram and a brief description of the features.*

1. **Communication Use-Case Diagram**

In the communication use case is demonstrated how the trucks can share different information based on their needs. One case might be turning information in this case they can perform V2V communication with on board communication system if they are in range. If the trucks are not in range, they still can communicate, but this time through a cloud-based control unit. And in this case, we make sure that communication is established, and it is stable in our system.

Diagram

Description automatically generated

# Structure of the software

*Describe the static structure of the simulation environment. Provide a class diagram for this purpose and briefly explain the classes or modules.*

**Truck Driving System Class Diagram**

The main components in driving controller that might be simulated are motion data and collision. Motion has three main sub-classes which are braking/stopping, Turning left/right and acceleration. Each sub-class has its own attributes. The second class is Collision which has two sub-classes. The sub-classes are collision warning and collision avoiding. Each sub-class has its own attributes as well.

Diagram

Description automatically generated

# Communication flow

*Describe the communication flow during the simulation between the individual trucks. Also describe the communication flow between the monitoring website and the trucks. Use one or more sequence diagrams for this purpose and briefly explain them.*

*Describe the connection or discrepancies to your event storming workshop outcome.*

**V2V Sequence Diagram-Data share**

Based in the use case we developed two different sequence diagrams. In this case the communication is done through on-board communication system that each truck has. Truck B collect some information regarding a specific scenario (Speed, obstacles, Road signs, Position). This information is forwarded to on board control unit of the truck B. The information is processed and verified in the control unit. After this step the control unit can broadcast this information to Truck A.in the next step the truck A send a confirmation to control unit if the information was received successfully.

**Diagram

Description automatically generated**

In the second sequence diagram the communication is done through a cloud based main control unit. In this case the truck sends information to the SW agent maybe an update in location or speed or another type of information. In the next step the SW agent deliver this information to the main control unit. The web-based control unit can now distribute the information to different trucks if they need this type of information. After the control unit receive the information it sends back to the truck a notification that the information was received successfully.

Diagram, box and whisker chart

Description automatically generated

# Election algorithms

*Which approach to choosing the lead vehicle will you take? Explain it briefly.*

# Operating instructions

*Give instructions on how to use the simulation. Potentially using an/more example(s).*

**Web interface**

All necessary files required for the web Interface are under one folder. It is required to open the index (index\_web\_Interface.html)file to perform all activities. The html file could be open in any kind of browser. For more information check the picture below. It is important to gain access a third part application for “allow cross-origin resource sharing” **Allow-Control-Allow-Origin: \* Extension in Chrome**. It depends on the browser that is used. If the access is denied than the information will not be displayed. Or this is realized by creating manually a web configuration by following the steps below (Not recommended).

For IIS6

1. Open Internet Information Service (IIS) Manager
2. Right click the site you want to enable CORS for and go to Properties
3. Change to the HTTP Headers tab
4. In the Custom HTTP headers section, click Add
5. Enter Access-Control-Allow-Origin as the header name
6. Enter \* as the header value
7. Click Ok twice

A screenshot of a computer

Description automatically generated with medium confidence

# Installation

*Describe how to install/unpack, launch and use the simulation.*

Prerequisites

*What are the requirements to run the software?*

# Allocation of tasks

*Breakdown: who did what*

*Describe which team member did which tasks.*

|  |  |  |
| --- | --- | --- |
| Member Names | Tasks | Task description |
| Enkeledi Mema | Web-Interface | -designing the all-web Interface using HTML,CSS,JS  -Building Json script for Get and Post method for consuming the microservices API  -Connecting the API with web interface  -Backend and frontend synchronization. |
|  | C++ Microservices library configuration | -Assisting in c++ library configuration and installation.  -Updating and creating new c-make file. |

# Sources

*Provide sources on the technologies and algorithms you want to use.*