Software Requirements Specification

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

PHM 4 TRUCK

Carlos Rojas

**Version: 1.0 preliminary** **Date: 01/27/2018**

**Table of Contents**

[**1. Introduction**](#_mqezhov2jj73) **3**

[1.1 Purpose](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.bl3ygcdltovw) [3](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.bl3ygcdltovw)

[1.2 Document conventions](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.hd2ii96zhxaz) [3](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.hd2ii96zhxaz)

[1.3 Intended audience](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.aljt3s47w34r) [3](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.aljt3s47w34r)

[1.4 Additional information](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.1qest09bnyr0) [3](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.1qest09bnyr0)

[1.5 Contact information/SRS team members](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.nwy0d6slgn6) [3](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.nwy0d6slgn6)

[1.6 References](#_lpmuk02ihac3) 3

[**2. Overall Description**](#_g64ah5wb0mv7) **4**

[2.1 Product perspective](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.shy7b3zg0s84) [4](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.shy7b3zg0s84)

[2.2 Product functions](#_iw1ez8234ju) 4

[2.3 User classes and characteristics](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.b7pmdrmnhte7) [4](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.b7pmdrmnhte7)

[2.4 Operating environment](#_b57ek5tb4klh) 4

[2.5 User environment](#_fkgw5o5zich9) 4

[2.6 Design/implementation constraints](#_6em82c3lszzc) 4

[2.7 Assumptions and dependencies](#_81yzjx9522r7) 5

[**3. External Interface Requirements**](#_d1at8ddakpuc) **5**

[3.1 User interfaces](#_dx6wbep17fr2) 5

[3.2 Hardware interfaces](#_hkf6wwebt8x7) 5

[3.3 Software interfaces](#_m5vqr2np4sf) 5

[3.4 Communication protocols and interfaces](#_kbxaqdygi7ol) 5

[**4. System Features**](#_hta8fhgwbyd1) **5**

[4.1 Planning features](#_otrhtah7ps4u) 5

[4.1.1 Check vehicles compatibility with a mission](#_s8owrkvphf3) 6

[4.1.2 Check vehicle’s spares parts compatibility with a mission](#_pg4vgnf7i09s) 6

[4.1.3 Send request for a spare part replacement](#_p22hywkmz0xo) 6

[4.1.4 Generate a dispatch order for a mobile repair unit](#_nuvo3fwygn26) 6

[4.2 Monitoring and Verifying features](#_9v2b78qn00zk) 7

[4.2.1 Check the timeliness of the planned operations (global)](#_q7ivvbczjitt) 7

[4.2.2 Estimate the increase on the assets effectiveness vs the traditional maintenance](#_yb2yg1b1wkfq) 7

[4.2.3 Monitor the level of consumption of spare parts and workload of repairers teams](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.d96ki36lda4t) [7](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.d96ki36lda4t)

[4.2.4 Generate reports on the current trends of the major maintenance parameters](#_8czvbfvwabx5) 7

[**5. Other Nonfunctional Requirements**](#_rs69lxdkatt4) **8**

[5.1 Performance requirements](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.z7xnifsqsfvm) [8](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.z7xnifsqsfvm)

[5.2 Safety requirements](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.fexj4gkwsg59) [8](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.fexj4gkwsg59)

[5.3 Security requirements](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.fexj4gkwsg59) [8](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.fexj4gkwsg59)

[5.4 Software quality attributes](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.fexj4gkwsg59) [8](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.fexj4gkwsg59)

[5.5 Project documentation](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.fexj4gkwsg59) [8](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.fexj4gkwsg59)

[5.6 User documentation](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.fexj4gkwsg59) [8](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.fexj4gkwsg59)

[**6. Other Requirements**](#_2mm0nrnolw3r) **8**

[Appendix A: Terminology/Glossary/Definitions list](#_2iy6omsp9ijb) 8

[Appendix B: To be determined](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.u4ljbunglz1z) [8](https://docs.google.com/document/d/1HBJQaeqDSpeDYI4VEhSISuDjuXgkNPxsiDlrz2JztOI/edit#heading=h.u4ljbunglz1z)

# 1. Introduction

## 1.1 Purpose

The purpose of the following document is to provide detailed description of the requirements for the “Prognostic assisted vehicle fleet maintenance” software. It will illustrate the purpose and complete declaration for the development of the system. Moreover it will describe system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

## 1.2 Document conventions

This document features some terminology which readers may be unfamiliar with. For this reason the following table was constructed:

|  |  |
| --- | --- |
| **User** | Someone who interacts with the mobile phone application |
| **PHM** | Prognostics Health Management |
| **SoH** | State of Health |
| **CBD** | Condition-based Data |
| **Stakeholder** | Any person who has interaction with the system who is not a developer. |
| **ECU** | Electronic Computing Units |
| **IVHM** | Integrated Vehicle Health Management (IVHM) |
| **CAN** | Controller Area Network |
| **GPS** | Global Positioning System |
| **GPS-Navigator** | An installed software on vehicles which could provide GPS connection and data, show locations on map and find paths from current position to defined destination |
| **Fault** | an unpermitted deviation of at least one characteristic property (feature) of the system from the acceptable, usual, standard operating condition. |
| **Failure** | a permanent interruption of a system ability to perform a required function under specified operating conditions. |
| **Fault Detection** | determination of the faults present in a system. |
| **Fault Detection** | determination of the faults present in a system |
| **Fault Diagnosis** | Determination of the kind, size, location and time of a fault. Fault Diagnosis includes fault isolation and fault identification |
| **Fault Isolation** | Determination of the kind, location and time of detection of a fault. Follows fault detection. |
| **Fault Identification** | Determination of the size and time variant behaviour of a fault. Follows fault isolation. |
| **Monitoring** | A continuous real-time task of determining the conditions of a physical system, by recording information, recognising and indicating anomalies in the behaviour. |
| **Prognosis** | Determination of whether a fault or failure is impending and estimate how soon a fault or failure will occur. |
| **Spare parts** | Parts of engine, exhaust system, transmission, chassis, fuel system. |

## 1.3 Intended audience

This document is intended for development team, project managers, testers, documentation writers in order to better understand requirements, goals of developing system as well as its constraints and dependencies.

Section 2 provides an overall description of the software, and section 3 describes the functional requirements of the project. Next, section 4 discusses the external interface requirements, and finally, in section 5, you can find the nonfunctional requirements of the project. Each section is divided in subsections where different matters are discussed.

## 1.4 Additional information

This document was developed under Software Architecture course in Innopolis University by master students of Software Engineering. All researches and findings have been made for educational purposes.

## 1.5 Contact information/SRS team members

|  |  |  |
| --- | --- | --- |
| **N** | **Team member** | **Contact information** |
| 4 | Carlos Rojas | c.monserrat@innopolis.ru |

## 1.6 References

IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications, In IEEE Xplore Digital Library. http://ieeexplore.ieee.org/Xplore/guesthome.jsp

Mokhov, S. (2010). Selected Project Requirements. In Concordia University. http://users.encs.concordia.ca/~c55414/selected-project-requirements.txt

# 2. Overall Description

## 2.1 Product perspective

The prognostic assisted vehicle fleet maintenance system is an additional component to existing customer logistic management which will report the health of engines and will make possible to detect faults and problems as they develop rather than after they occur. Particularly, it will provide great benefits for charity organization by helping to detect different faults, prevent severe failures or interruptions of operation, discover new knowledge and utilize data collected onboard from regular operations. The system consist of several components:

* components for gathering data from vehicles
* back-end data processing component
* component for presenting results to end-user in a convenient way

## 2.2 Product functions

The system performs the following functions. The functions depend on the user’s level and permission package, as explained in the user characteristics.

* Assist in convoy formation
* Spare parts order planning assistance
* Show list vehicles with spare parts statuses
* Show list of vehicles suitable for specific mission

## 2.3 User classes and characteristics

The software is intended for humanitarian organisation which deliver humanitarian relief to those in need.

Two type of users exist:

* Operators - plan and monitor mission, format convoys and manage stocks of spare parts
* Supervisors - with the support of software monitor prognostic effectiveness

Specifically, **prognostic operators** in the “Smart logistics” workstations with the support of the software, check the residual life of all the assets (vehicles) in the convoy and define the compatibility with the considered mission considering the attrition factor of the specific route. A **supervision operator** will proceed to monitor the effectiveness of the prognostic activity.

Both operator and supervisor should be specifically trained to use the new software functions and tools.

## 2.4 Operating environment

Prognostics system must be deployed on the existing resources of the control rooms. In the list below, Main Control Room working functionalities are described:

1. Monitor and track relief operation lifecycle (Convoy preparation, tracking, delivery, return)
2. Monitor and track mission operation
3. Monitor Hostile and critical conditions and provide warning and alternatives
4. Manage criticality and accidents involving the mission
5. Generate activity reports
6. Define operators schedules and turnover
7. Monitor and reporting on stocking areas

## 2.5 User environment

Up to date PC based individual work stations with enhanced graphic capability multiple screens.

## 2.6 Design/implementation constraints

System must run on the following hardware. From Operations Control assets side, 3 high reliability server clusters , 20 up to date PC based individual work stations, connected by a Local High speed bus are used to power Main Control room.

Moreover, 2 second Level Control rooms supported by 2 servers and 5 PC based stations with multiple screens. High speed communication link (Fiber) with the Main CR is provided.

Furthermore, operators devoted to operations and Convoy assembly and movements have 100 remote terminal (Android). Normally, 4G service is available (up 60%), but in some locations and areas GSM can be used as a backup degraded mode service (up 95%) 50 terminal have parallel Satellite communication capability for “difficult” areas. In addition, all Vehicles have HF Radio on board.

Android software must run on any versions of Android OS from 4.1.

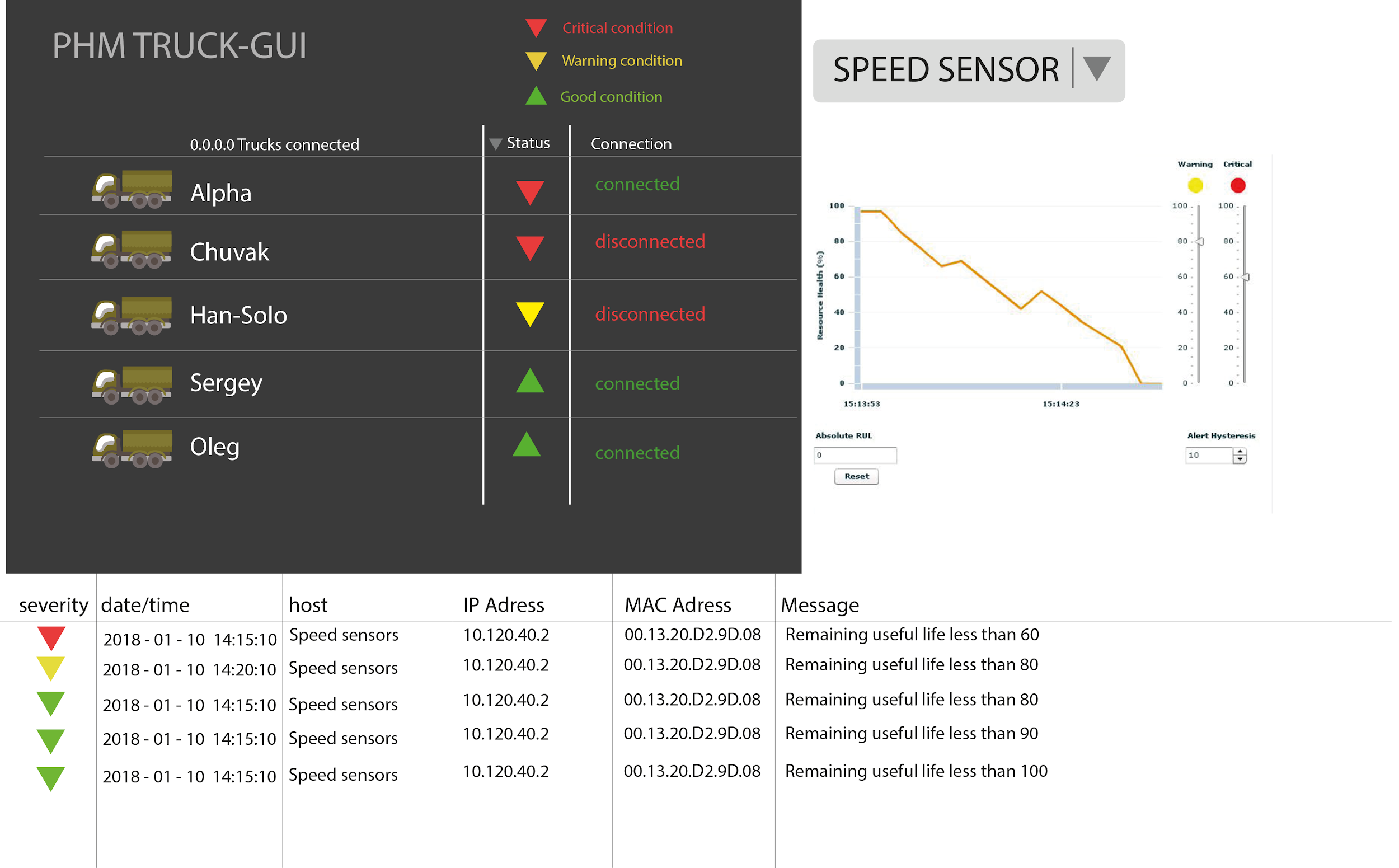
## 2.7 Assumptions and dependencies

The system depends on existing software and should be integrated in it so that to have an access to all existing data.

# 3. External Interface Requirements

# 3.1 User interfaces

The GUI should be easy to learn and use by users of any technical background. A built-in help feature should be available in all program states, to guide the users with the available functions on that state. An easy to understand documentation should be provided with the system.



* Key failures detected and ranked.
* Precursor signatures or multivariate inputs extracted using prognostics & health management (PHM) software tool.
* Historic and trending data are used to fine-tune the analysis.
* Prognostic metrics are generated, such as current state-of-health (SoH) and remaining useful life (RUL).

## 3.2 Hardware interfaces

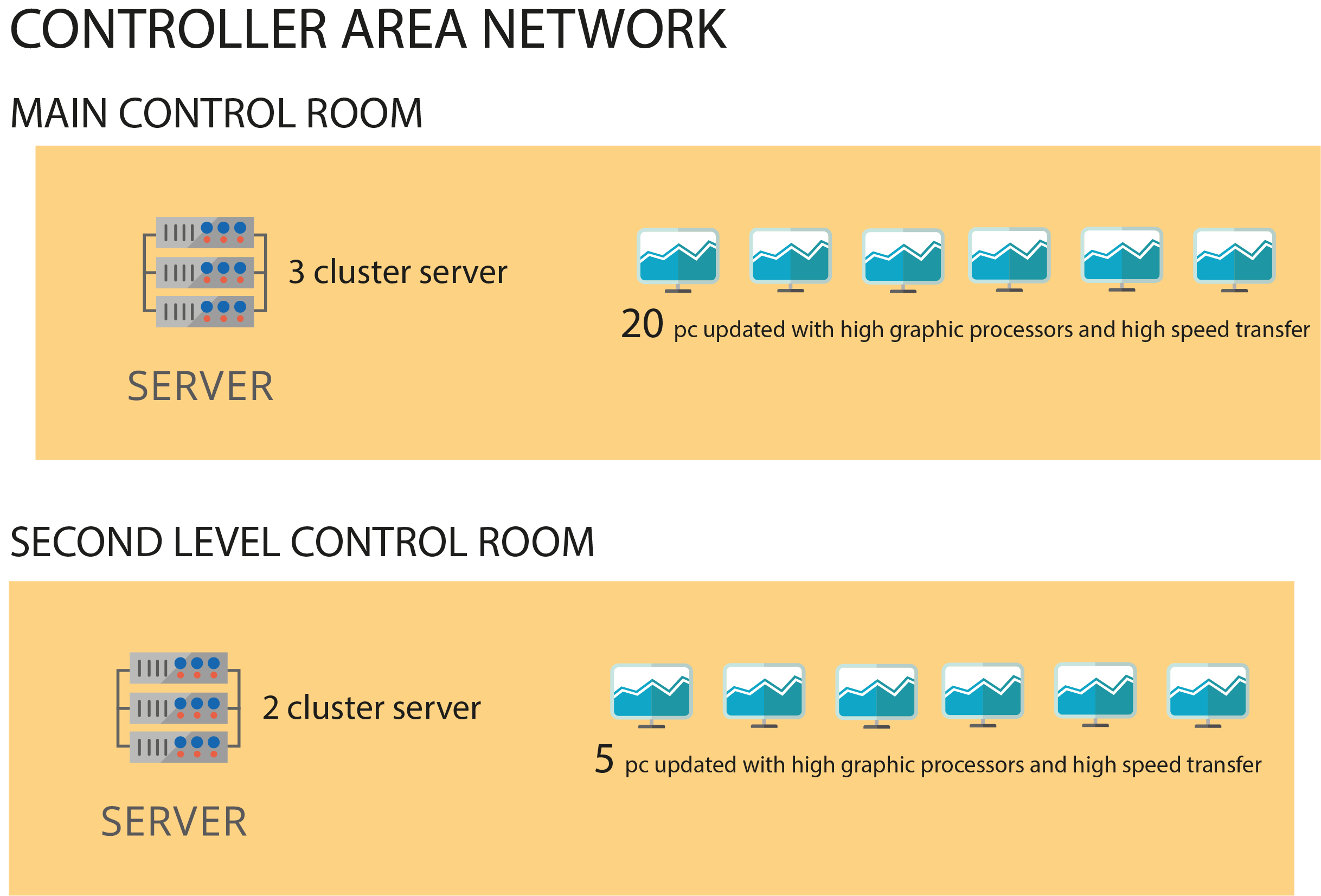
A vehicle is a complex mechatronic systems composed of subsystems such as brakes, engine and gearbox. A typical subsystem consists of sensors, actuators and an electromechanical process which needs to be controlled.

**Required sensors for each truck**



1. Collision avoidance, night vision, and front crash detection, forward obstacle sensor
2. Vehicle distance sensor
3. Road condition sensor
4. Side obstacle sensor rear radar, rear obstacle sensor
5. Oil/fuel pressure and flow monitoring
6. Speed sensors comfort control
7. Tire pressure monitoring
8. Fire detection sensor
9. Driver monitoring sensor
10. Steering angle sensor and stability control
11. Yaw and acceleration sensors for airbag deployment
12. Side crash detection
13. Side airbag deployment sensor
14. Angular acceleration (suspension)
15. Emissions sensor
16. Back-up collision, rear vision camera,
17. Temperature and humidity sensor, and comfort control
18. Rollover detection
19. Rain sensor and wiper control
20. Power train control module
21. Throttle position monitoring and control
22. Battery monitoring
23. Ignition and engine control monitoring

**Required for CAN (as specified by customer)**

****

****

## Required vehicles

* 100 heavy weight trucks.
* 100 light weight trucks.

## 3.3 Software interfaces

## 3.4 Communication protocols and interfaces

# 4. System Features

## 4.1 Planning features

These features are intended for a prognostic operator who needs to check the residual life of all the assets (vehicles) in the convoy and define the compatibility with the considered mission considering the attrition factor of the specific route.

### 4.1.1 Check vehicles compatibility with a mission

When the application is running the operator is presented with the main dashboard, which includes a button “Missions". The operator clicks the button and proceeds to a page which shows all the coming missions. The operator clicks on a particular mission. A list of vehicles, which are included in the mission, is displayed on the screen as a table. The table includes the following columns (among others):

* residual life;
* compatibility with the mission.

The rows of the table are also highlighted by different colors according to the compatibility with the mission.

### 4.1.2 Check vehicle’s spares parts compatibility with a mission

When the application is running the operator is presented with the main dashboard, which includes a button “Missions". The operator clicks the button and proceeds to a page which shows all the coming missions. The operator clicks on a particular mission. A list of vehicles, which are included in the mission, is displayed on the screen as a table. The operator clicks on the vehicle and a list of the spare parts of the vehicle is displayed as a table. The table includes the following columns (among others):

* residual life;
* compatibility with the mission.

The rows of the table are also highlighted by different colors according to the compatibility with the mission.

### 4.1.3 Send request for a vehicle replacement

When the application is running the operator is presented with the main dashboard, which includes a button “Missions". The operator clicks the button and proceeds to a page which shows all the coming missions. The operator clicks on a particular mission. A list of vehicles, which are included in the mission, is displayed on the screen as a table. Every row has a button “Replace". The operator clicks the button and the vehicle (the row) is highlighted. The button “Replace" of this row is replaced by “Cancel" button.

### 4.1.4 Generate a dispatch order for a mobile repair unit

When the application is running the operator is presented with the main dashboard, which includes a button “Missions". The operator clicks the button and proceeds to a page which shows all the coming missions. The operator clicks on a particular mission. A list of vehicles, which are included in the mission, is displayed on the screen as a table. The operator clicks on the vehicle and a list of the spare parts of the vehicle is displayed as a table. Every row has a button “MRU order". The operator clicks the button and the vehicle (the row) is highlighted. The button “MRU order" of this row is replaced by “Cancel" button.

## 4.2 Monitoring and Verifying features

These features are intended for a supervisor, who monitors the effectiveness of the prognostic activity.

### 4.2.1 Check the timeliness of the planned operations

When the application is running the supervisor is presented with the main dashboard, which has a menu panel at the top. The menu has a “Analysis" button. The operator clicks the button and a dropdown menu appears. The menu includes an option “Missions". The operator clicks this option and proceeds to a page which shows all the completed missions as a table. The table includes the following columns (among others):

* delay (h);
* delay (%);.

Those attributes show deviation from the planned time of the mission in hours and percents.

### 4.2.2 Estimate the increase on the assets effectiveness vs the traditional maintenance

When the application is running the supervisor is presented with the main dashboard, which has a menu panel at the top. The menu has a “Analysis" button. The operator clicks the button and a dropdown menu appears. The menu includes an option “Effectiveness". The operator clicks this option and proceeds to a page which shows a bar chart comparing the consumption of spare parts and workload between new system and traditional maintenance (for example, consumption per 100 km).

### 4.2.3 Generate reports on the current trends of the major maintenance parameters

When the application is running the supervisor is presented with the main dashboard, which has a menu panel at the top. The menu has a “Analysis" button. The operator clicks the button and a dropdown menu appears. The menu includes an option “Reports". The operator clicks this option and and a dropdown menu appears with options:

* MTBS;
* Time off line;
* Time to repair;
* Spare parts usage.

The operator clicks on one of the options and proceeds to a page on which he specifies additional parameters of the report (term, format).

# 5. Other Nonfunctional Requirements

## 5.1 Quality requirements

### 5.1.1 Reliability

The system should work reliably, with automatic backup and recovery features. In case of unexpected termination of a session, the unsaved data should be recovered without loss and displayed to the respective users for saving into the system or continuing with the work. At any time, audit file and all db and mailing information are required to be updated in the backup.

### 5.1.2 Availability

The entire system should be available round the year, except for a periodic maintenance. The maintenance period should be pre scheduled and short. The users should be reminded of the unavailability period, well in advance.

### 5.1.3 Perfomance

Performance is not critical issue. Data analyze and making prognoses should take not more than first preparation phase of mission planning which often takes about several hours.

Only one of available server should be enough to compute prognosis even if it takes more time.

The system is required to support multiple terminals simultaneously. The system should handle reasonable number of users without break or inconsistency.

### 5.1.4 Security

Application assumes that only owner company has access to data and permissions for interact with data. Network communications should use cryptographic protocols such as SSL. The system is required to end the session automatically, when an open session is not used for a specific period of time.

### 5.1.5 Maintainability

The document should be easy for the users who execute the system day to day, for the developers who wish to edit or develop further, and for the personnel who is in charge of the maintenance.

### 5.1.6 Portability

The system should be supported on standard PC. The application for android must work on Android 4.1+ OS.

### 5.1.7 Safety

95% of information transmission should be securely transmitted to server without any changes in information.

Stored data should be consistent otherwise prognoses quality will be less than expected.

## 5.2 Software quality attributes

Accuracy of prognoses should be at least 70%

There will be zero tolerance for errors in the algorithm that computes prognoses.

Every method should have maintainability index of 40%

Every class should have maintainability index of 40%

System is easy to handle and navigates in the most expected way with no delays.

## 5.3 Project documentation

Software requirements specification

Software Development Plan

High Level Design

## 5.4 User documentation

Manual for a Prognostics Operator

Manual for a Supervision Operator

# 6. Other Requirements

## Appendix A: Terminology/Glossary/Definitions list

## Appendix B: To be determined