Requirement Specification

Editor: Pontus Erlesand

Version 1.0

Status

Reviewed	Pontus Erlesand	2016-10-04
Approved	Danyo Danev	2016-10-07

PROJECT IDENTITY

2016/HT, TSKS05-POZYX
Linköping University, (ISY)

Group members

Name	**Responsibility**	**Phone**	**Email**
Rasmus Vilhelmsson	Project Leader (PL)	0700 423 400	rasvi146@student.liu.se
Pontus Erlesand	Head of documentation (DOC)	0707 964 955	poner538@student.liu.se
Markus Petersson	Head of sensor fusion	0703 823 630	marpe163@student.liu.se
Ching-Hsiang Yang	Head of hardware	0737 227 489	chiya469@student.liu.se
Susmita Saha	Head of testing	0720 451 029	sussa301@student.liu.se

Group e-mail: pozyx.cdio@gmail.com
Customer: ISY, Linköping University, 581 83 Linköping
Customer contact: Danyo Danev, 013-281335, danyo.danev@liu.se
Course leader: Danyo Danev, 013-281335, danyo.danev@liu.se
Supervisor: Trinh Van Chien, 013-282643, trinh.van.chien@liu.se

Contents

Document history

Version	Date	Changes	Sign	Reviewed
0.1	2016-09-28	First draft		PE
0.2	2016-10-04	Revised version		PE

1 Introduction

The following requirement specification concerns the CDIO project for the course TSKS05, where the purpose of the project is to create a system, using the Pozyx platform, to allow a user to position and track a tag. To be able to verify the quality of the system a requirements specification is made, where each requirement is carefully specified. This should make each requirement easy to test and verify.

All the requirements are structured as in the table below.

Req. x	**Original or revised**	**Description of requirement**	**priority (1,2,3)**

Requirements with priority 1 must be fulfilled for the final delivery. Requirements with priority 2 will be implemented if time and other resources allow. Finally requirements with priority 3 is merely suggestions for further development and will only be initiated if all requirements with higher priority is met.

1.1 Who is involved

In this project the group members of this project is presented in the beginning of this document and consists of five master's students at Linköping University taking the course TSKS05 (CDIO, communications systems). The client and customer is the same person, Danyo Danev, and the supervisor is Trinh Van Chien. Danyo Danev is an Associate Professor and Trinh Van Chien is a PhD student, both at the division for communications at the department of electrical engineering at Linköping University.

1.2 Goals

The goal of this project is to design a system, based on the Pozyx system, that enables tracking and positioning of a tag, and to present the results visually using a GUI written in Matlab.

1.3 Usage

The user should be able to get the position of a tag, and continuously track it. Estimates of position and trajectory should be presented in a GUI.

1.4 Background information

This project is part of the CDIO project course TSKS05. The project group consists of five master's students majoring in Communication Systems.

1.5 Definition of terms

Term	Description
2D	Two dimensions
3D	Three dimensions
LOS	Line of sight
NLOS	No line of sight
KF	Kalman filter
EKF	Extended Kalman filter
UKF	Unscented Kalman filter
PF	Particle filter
IMU	Inertial measurement unit
GUI	Graphical user interface
TOA	Time of arrival

2 Overview of the system

The system will consist of two main modules, which are the *hardware* and *software* modules. The hardware module concerns the Arduino, the Pozyx shield and the anchors. The software module consists of two sub modules which are data processing, used for positioning and tracking, and the GUI.

An overview of the system is found in figure 1.

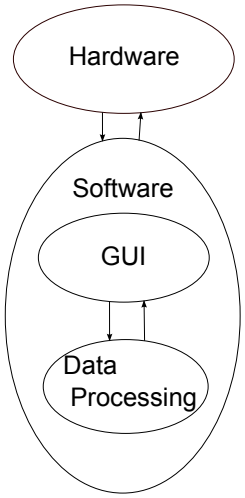

Figure 1: Overview of the system.

2.1 Description of the system

The hardware module will collect data from the tag and send those to the computer, i.e. the software module. From there all the data processing will be done and the result passed to the GUI. In the GUI it will be possible to set parameters and settings for the data processing.

2.2 Product components

The system is divided into two modules, the hardware module and the software module. The hardware module concerns the hardware of the system, i.e. the Arduino, the Pozyx shield and the Pozyx anchors. The software module concerns the data processing and the GUI of the system.

2.3 Dependency of other systems

The dependencies between modules are described in this section.

2.3.1 Hardware

The hardware is mostly non-dependant of the software. The only dependency that is present is that the tag needs to know the positions of the anchors, these positions are uploaded to the tag from the software module.

2.3.2 Software

The software depends on the hardware as all the data is collected from the tag. As for the sub modules, the GUI is dependant of the processed data to be able to visualize any results. The data processing has a minor dependency of the GUI, these are some settings and parameters used in the data processing, which results in a two-way communication between the software sub modules.

2.4 Design philosophy

We plan to develop our project in two major modules, where the software module consists of two sub modules. The idea is to efficiently allow different sub groups of the project group to work in parallel.

2.5 General demands of the complete system

The system should be able to localize and track a tag for different anchor configurations.

3 Hardware

The requirements for the hardware are found in the table below.

Req. 1	**Original**	The hardware should connect to the computer over serial port emulated by USB	**1**
Req. 2	**Original**	The hardware should be able to forward all available data to the computer	**1**
Req. 3	**Original**	2D positioning and tracking should work with at least 3 trackers	**1**
Req. 4	**Original**	3D positioning and tracking should work with at least 4 trackers	**1**
Req. 5	**Original**	Find the optimal placement of Pozyx anchors in the corridor	**2**
Req. 6	**Original**	Data processing algorithms should be implemented on Arduino	**3**

4 Software

In this section the requirements regarding the software is listed. The data processing sub module concerns positioning and tracking of the tag, whilst the GUI concerns the visualization of such data.

4.1 Data Processing

Positioning

The functional requirements for the positioning are found in the table below.

Req. 7	**Original**	The system should be able to estimate the position of a tag in 2D	**1**
Req. 8	**Original**	The system should be able to estimate the position of a tag in 3D	**2**
Req. 9	**Original**	The system should use TOA algorithms to estimate the position of a tag	**1**
Req. 10	**Original**	The system should use averaging from the few latest estimates given from the tag	**2**
Req. 11	**Original**	The system should be able to estimate the position of a tag with an error margin less than 2 meters	**1**
Req. 12	**Original**	The system should be able to estimate the position of a tag with an error margin less than 1 meter	**2**
Req. 13	**Original**	The system should be able to provide the covariance of an estimate	**2**
Req. 14	**Original**	The system should be able to weight it's estimates based on the received signal strength from the different anchors	**1**
Req. 15	**Original**	The system should notice when the tag is not in LOS	**2**

Tracking

The functional requirements for the tracking are found in the table below.

Req. 16	**Original**	The system should be able to estimate the trajectory of a tag in 2D using the obtained estimates of it's position	**1**
Req. 17	**Original**	The system should be able to estimate the trajectory of a tag in 3D using the obtained estimates of it's position	**2**
Req. 18	**Original**	The system should be able to estimate the current velocity of the tag	**1**
Req. 19	**Original**	The system should be able to estimate the trajectory of a tag using a Kalman filter (KF)	**1**
Req. 20	**Original**	The system should be able to estimate the trajectory of a tag using an Extended Kalman filter (EKF)	**2**
Req. 21	**Original**	The system should be able to estimate the trajectory of a tag using an Unscented Kalman filter (UKF)	**2**
Req. 22	**Original**	The system should be able to estimate the trajectory of a tag using a Particle Filter	**2**
Req. 23	**Original**	The system should be able to utilize the Inertial measurement unit (IMU) to provide better estimates of the trajectory	**2**

4.2 GUI

The functional requirements for the GUI are found in the table below.

Req. 24	**Original**	The GUI should be made using Matlab	**1**
Req. 25	**Original**	The GUI should show a map of the Communications Systems corridor	**1**
Req. 26	**Original**	The GUI should show the positions of the anchors	**1**
Req. 27	**Original**	The GUI should show the estimated position of the tag	**1**
Req. 28	**Original**	The GUI should show the estimated position error of the tag	**2**
Req. 29	**Original**	The GUI should show the estimated distances to each anchor from the tag	**2**
Req. 30	**Original**	The GUI should indicate the estimated direction in which the tag is moving	**2**
Req. 31	**Original**	The GUI should present the predicted movement of the tag.	**2**
Req. 32	**Original**	The GUI should show available sensor data	**1**
Req. 33	**Original**	The GUI should show a warning-like signal when the tag goes out of LOS	**2**

5 Interfaces

The functional requirements for the interfaces between modules and sub modules are listed below.

Req. 34	**Original**	The anchors should send data to the tag	**1**
Req. 35	**Original**	The tag should send data to the computer for data processing	**1**
Req. 36	**Original**	The GUI should receive processed data	**1**
Req. 37	**Original**	The GUI should be able to send parameter settings to the data processing module	**1**

6 Possibility of upgrading

Requirements regarding the upgradability of the system can be found in the table below.

Req. 38	**Original**	The system should be module based so that each model can be upgraded separately if desired	**1**

7 Economy

Requirements regarding economy in the project can be found in the table below.

Req. 39	**Original**	Each member of the project group has 240 working hours at their disposal for this project	**1**

8 Delivery

The delivery requirements are listed below.

Req. 40	**Original**	The Requirements Specification, Design Sketch and the Project Plan should be passed no later than 7th October	**1**
Req. 41	**Original**	The Time Plan should be passed no later than 7th October	**1**
Req. 42	**Original**	The final product, the technical documentation and the user manual should be handed in and passed no later than 20th December	**1**

9 Documentation

Requirements regarding the documentation of the project can be found in the table below.

Req. 43	**Original**	The documentation should be according to the LIPS model	**1**
Req. 44	**Original**	The documentation should be written for people that are familiar with the technical aspects that are used.	**1**
Req. 45	**Original**	All documents in the project should be delivered to the sponsor. The dates for all deliveries are according to what is mutually decided between the group and the sponsor.	**1**

All documents that, according to the LIPS model, will be in the project can be found in table the table below.

Document name	Description	Target group	File format
Requirement specification	The sponsor's requirements for the project	Sponsor, Project group	PDF
Project plan	An overview of the project containing e.g. planned tasks and areas of responsibility for the group members	Sponsor, Project group	PDF
Time plan	Describing the how the work load will be distributed between group members and what they should be working on	Sponsor, Project group	XLS
Design sketch	Providing an overview of the sub systems of the product and how these will interact in the final product	Sponsor, Project group	PDF
Design specification	A more detailed design sketch that goes into more detail on how the sub systems will interact	Sponsor, Project group	PDF
Technical documentation	A more detailed design sketch that goes into more detail on how the sub systems will interact	Sponsor, Project group	PDF
User manual	Describes how the product is used	Sponsor	PDF
After study	Accounts for the group's experiences, concerning cooperation and planning and usage of the LIPS model, during the project	Sponsor, Project group	PDF

10 Training

The training for the group members in the project will consist of the items listed below.

- All group members will read through all the documentation on the Pozyx's website, i.e. data sheets and manuals etc.
- The group members responsible for the data processing will read up on sensor fusion.
- The group members responsible for the GUI will read up on making a graphic interface in Matlab.
- The group members responsible for the hardware will read up on Arduino usage and functionality.

11 Quality

Requirements regarding the quality of the system can be found in the table below.

Req. 46	**Original**	The system should be able to achieve the desired accuracies for different anchor configurations	**1**

12 Maintainability

Some requirements for usage is present for this project. The GUI will be written in Matlab and will not be a standalone executable file, this might be upgraded if time allows, the user will therefore need Matlab installed. Furthermore the user needs the driver installed from the Arduino software, to enable serial over USB on the computer. Additionally, if any anchors break down and needs to be replaced the source code needs to adjust to these new changes, i.e. change the ID to the new anchor in the code. This applies with a new set of anchors as well. If only the tag is replaced no code needs to be adjusted, however the source code needs to be uploaded to the new tag.

The requirements for the final product's maintainability is listed below.

Req. 47	**Original**	The GUI will be a standalone exe-file	**3**
Req. 48	**Original**	The user should only need Matlab and Arduino software to run the system	**1**
Req. 49	**Original**	A new user should be able to access the source code	**1**

References

[1] *LIPS – nivå 1. Version 1.0.* Tomas Svensson och Christian Krysander. Compendium, LiTH, 2002.