**ABILISENSE**

**Information collection system**

Royi Journo

Or Hagbi

1. [**Introduction**](#Introduction) **……………………………………………………………………………………………………………3**
   1. [Purpose](#Purpose) ……………………………………………………………………………………………………………...3
   2. [Definitions, Acronyms, and Abbreviations](#Definitions) ………………………………………………………..…3
   3. [System overview](#SystemOverview)……………………………………………………………………………………………..…..3
   4. [References and Related Documents](#References)…………………………………………………………..………..3
2. [**Overall description**](#OverallDescription) **…………………………………………………………………………………………………..4**
   1. [Product perspective](#ProductPerspective)…………………………………….………………………………………………………..4
   2. [Product functions](#ProductFunctions)……………………………………………………………………………………………..…..4
   3. [User characteristics](#UserCharacteristics)…………………………………………………………………………………………..…..4
   4. [Constraints, assumptions and dependencies](#Constraints)……………………………………………………..….4
3. [**Specific requirements**](#SpecificRequirements)**……………………………………………………………………….……………………….5**
   1. [External interface requirements](#ExternalInterfaceRequirements)…………………………………………………………………………….5
   2. [Functional requirements](#FunctionalRequirements)…………………………………………………………………………..……………5
4. **Introduction**  
   The essence of the project is to create a system that will use sensors input, with them the system can collect information and store it in cloud.  
   1. Purpose  
      This document contains all the software requirement specifics. It contains a general description of the types of users who will be using our application, how it is going to work and what technologies we are using to make it work.
   2. Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Term** | **Definition** |
| SW | Software |
| DB | Data Base |
| SRS | Software Requirement Specification |
| CL | Computer Language |
| UX | User Experience |
| UI | User Interface |
| MSSQL | Microsoft SQL server |
| JS | JavaScript |
| BE | Back End |
| UML | Unified Modeling Language |
| UC | Use Case |

Table .1 – Definitions & abbreviations.

* 1. System Overview  
     Our program need to be connected to the device's sensors, with them, record the data (photo, sound etc.) and store them in cloud.

we will split the program to client side and server side.

the client-side UI will be implemented using React for web and React Native for IOS&Andriod.

the server side will be implemented using C# (WebService).

our DB will be the cloud, that will provided from ABILISENSE.

* 1. References and Related Documents

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Document ID(file name)** | **Document Title** | **Comments** |
| [1] | <https://reactjs.org/docs/hello-world.html> |  | React Documentations. |
| [2] | <https://facebook.github.io/react-native/docs/getting-started.html> |  | React Native Documentations. |
| [3] | <https://www.udemy.com/cloud-storage-with-aws/> |  | Cloud Storage Course on Udemy |

1. **Overall description**
   1. Product perspective

The product is part of a larger system that has a machine learning system that should receive a response from the environment, analyze and classify it, and as a result will alert the user.

The purpose of our product is to provide the information to teach the system how to analyze and classify the alerts.

* 1. Product functions

The product functions are described in use-cases.

**UC1: Record Data**

Description:

The user will be able to upload data to the cloud using his phone's sensors, then the user need to choose tag to his data record and finally upload it to the cloud.

**UC2: Create account**

Description:

The user creates an account to be able to use the system services.

**UC3: Login**

Description:

The user enters username and password to gain access to the system.

* 1. User characteristics

**End Users:**

End users are users who registered to the system in order to use the system services. For example: The user takes a photo of bicycle, attach the tag "Bicycle" and upload it to the cloud.

* 1. Constraints, assumptions and dependencies

**Limitations:**  
1. Quality of the record data.

2. Quantity & Quality of collaborators.

**Interfaces for communication types:**

1. Phone/Computer Network to connect to the cloud.

**Operating the software:**

Will be from the users of the application, after downloading the application to the phone/computer, will have the Ability to record data, add tagging and upload them to the cloud.

**System reliability needs:**

We must rely on the users to take the information in an optimal manner and to correctly label the data.

1. **Specific requirements**
   1. External interface requirements

**Description of the environment:**

1. Client side will be Visual Code, the development will be in the CL React & React Native.

2. Server side will be in Visual Studio, the development will be in the CL C#.

* 1. Functional requirements

Login

Sensors Already Scanned?

Yes

No

User Record Data and Choose Tag For it.

Figure 3.1 – Flow Chart