**Requirement**

Software Requirements engineering are the descriptions of what the system should do[1]. These roles calcify the kernel requirements stakeholders need to integrate in this product and details of system functions which guide how system will be implemented. This mobile system is designed for the pregnant women to monitor their vital signs and help them keep healthy during the pregnancy. To be easily used by most pregnant women, the application should be easy enough to operate and understand. And the system should be long-term running to support the data mining. Another key direction is interaction with user, which help users to know the system and remind them.

**.1 Requirement Elicitation**

The customers of our project are Professor Vladimir’s research group. Our project is a part of the whole research group about mental health monitoring. Our purpose is developing a mobile system to support PhD’s algorithm about data processing and research. The main methods we decided to gain the requirements are literature research and interviews or investigations with customers in the workshop. According to research, we need to find the features of target users (pregnant women) and the feedback of some existing systems. Then we will make the draft of requirement documentation and discuss with our customers to modify the details, validate roles, and add some extra roles. Following the software engineering process, we plan to continually gather the key requirements for several weeks and finally confirm the final requirement documentation.

**.2 User Requirement**

This part specifies the requirements considered by the target users: pregnant women. And here are the requirements from research and meeting communication with customers.

**Functional Requirement**

* 1. **Connection**

The user could use the application to connect with the wearable devices.

* 1. **Monitoring**

Pregnant users are concerned about their health conditions and want to know the current body states and if they are healthy.

* 1. **Comprehensive Report**

Users want to be given a comprehensive report about their bodies which could help them to understand their health conditions and life advices.

**Non-functional Requirement**

* 1. **Useability**

Users want their body data visualized and easy to understand. Users also want a low threshold to use the application and to be reminded to use the application.

* 1. **Emergency dealing**

Users do not want to wear the wearable devices too long throughout the day. They might take off the wearable devices sometimes.

**.3 System Specification**

#### Functional Specification

1. **Data Capture Function.**

The mobile system (Android Application) can connect with the wearable devices which include heart rate belts, brain ware, smart scale.

1.1 The vital data will be sent from wearable device to mobile application every minute.

1.2 The bodyweight data is captured by manual record.

1.3 The year and height are recorded from users.

1.4 The data will be sent to the integration module for the next step.

1.5 When the application detects that the data signals is interrupt or zero, the capturing will stop.

1. **Data Integration Function.**

2.1 The data will be processed and temporarily store in the application's memory.

2.2 The initial data will be checked, and the error data will be cleaned.

2.3 Multiply types of data can be integration in the mobile system.

2.4 The data will be split into different categories and shown to the users.

2.5 The data will be sent to the analysis module for the next step.

1. **Data report and analysis Function**

3.1 When the application captures and process data more than 30 minutes, the current report will be generated. This report is about analysis of the past period.

3.2 The report is about his/her body data. The report will include every-day vital signs data after processing and will give some advices according to the analysis algorithm such as having more sleep or having abnormal status and need to see doctor.

3.3 The system could generate the report by the suggestion data and analysis algorithm daily and monthly (or weekly).

3.4 The report could be preserved by users and exported.

3.5 The analysis in report is just for reference and cannot be correct and accurate. Report will ask user to see doctor for the further diagnosis.

3.6 When the system detects the short interrupt (less than 1 minutes), the data collection is continued without 30 minutes re-collection.

3.7 The data transmission between wearable devices and mobile application is based on blue-tooth protocol.

1. **Data storage Function**

4.1 The data from users should be stored both in PC and mobile system.

4.2 The received data will be stored for a month in the application memory for users' views.

4.3 Considering the limited memory, the details of past data will be cleaned up and the daily report will be stored in the mobile system.

4.4 The user could connect with the PC monthly and send the data for the long-term storage.

1. **Task management (interface)**

5.1 The system will push the every-day reports, suggestions, and analysis at 8 a.m.

5.2 The system will send a message to user if user does not. wear the devices.

5.3 The data capture function can be switch on/off.

5.4 When the application cannot find the data, application will remind user to check the wearable device wearing.

5.5 The system will send a message if the user does not record the body weight until 8 p.m. every day.

5.6 The user can view four mode of application:

* + - * + current/last measured data
        + today's report
        + Long term report
        + Settings

#### Non-functional Specification

* + - **Security**
  1. To prove the security of personal data, the storage module needs to be private with the outside system.
  2. The wearable device connections should be confirmed by users. User can cancel the connections at any time.
     + **Performance**
  3. The advice and message should be accurate and on time.
  4. Use the suitable data analysis algorithm to prove the correctness of data, and error data need to be cleaned.
     + **compatibility**
  5. The application should be compatible for android platform.
  6. The application User interface should be compatible for different size of screen.
  7. The advice should be given according to user status
  8. System should deal with exception of data.

### Technical Specification

1. The mobile system should support the device types:
   * + - (Heart Rate) Chest belt: **Polar H10**
       - (Brain Wave) Brain belt: **NeuroSky TGAM**
       - Smart Scale: **Yunmai**
2. The wearable devices official SDK is open source for the development and will be used during the project.
3. The development platform is android studio, and the development language is Java.
4. The mobile system will operate in PC for demonstration (simulator) and on android cellular phone with Android 11 system.
5. The simulator of mobile system is GenyMotion/AVD emulator. It will be used to simulate the process of mobile system for demonstration and test.
6. The data to test is from the research group which include the weight change of pregnant women and references.

**.4 System Design**

The whole mobile system according to the requirement specifications can be divided into five main modules: data capture module, data integration module, data analysis module, data storage module, task management module. The system’s task is to connect with mobile devices and capture data from them. Then shows the data after processing in the stage of application for user view and finally generate a report that includes vital signs daily review, analysis, and advices.

**Data Capture Module**

This module is used to capture the data from wearable devices.

**Data Integration Module**

This module is used to process, clean, and integrate the initial data from capture module.

**Data Analysis Module**

This module is used to process the integrated data by algorithm and generate a report of users’ health conditions.

**Data Storage Module**

This module is used to store the data and report.

**Task Management Module**

This module is GUI and tasks of user messages.

Somerville, Ian. (2011). Software Engineering. 9th ed. Addison-Wesley. pp 147-202