**SOFTWARE REQUIREMENT SPECIFICATIONS**

**Introduction**

A software requirements specification is a comprehensive description of the intended purpose and environment for software under development. The SRS fully describes what the software will do and how it will be expected to perform.

An SRS minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines how an application will interact with system hardware, other programs and human users in a wide variety of real-world situations. Parameters such as operating speed, response time, availability ,portability, maintainability, footprint, security and speed of recovery from adverse events are evaluated.

**1.1Purpose**

**1.1.1Customer Feedback**

The software requirement specification ensure the project management stakeholders and client that the development team has really understood the business requirements and documentation  properly. This also provides confidence that the team will develop the functionality which has been detailed.

**1.1.2Breakdown Structure**

The Software Requirement Specification is documented in such a way that it breaks the deliverables into smaller components. The information is organized in such a way that the developers will not only understand the boundaries within which they need to work, but also what functionality needs to be developed and in what order.

**1.1.3Facilitating document**

The SRS forms the basis for a load of other important documents such as the Software Design Specification.

**1.1.4 Product Validation**

It basically helps in validating with the client that the product which is being delivered, meets what they asked for.

## 1.2 Document Conventions

The Font of the heading title is 16.The subheading font is 14 and the content of the font is 12.We have used the times new roman.

## 1.3Intended Audience and Reading Suggestions

The SRS is intended for developer who could understand the requirement and the functionalities of the project.Also the developer and manager should know the expectation of the client and customer and also the feature should be a s according to customer.The client should get satisfaction and if improvement required should be made.It is like proof for the client so that it could validate the product or when project is done.

**1.4Product Scope**

The project is basically focused on working class parents.It will benefit the parent and it could have greater social impact and business advantages.

**2.Overall description**

**2.1 Product Overview**

The project basically tries to solve the problem faced by modern generation parents who are from working class society.Due to their professional life,they could be present beside the child all the time and also this project will reduce the dependence of parents on nanny so could help in saving of money and sometime the nanny which are hired for child may not be person with good intention.So,This project I guess,will solve the problem of parents to some extent.As you can see everyone has smartphone now a days so we are usin it to have the status of the E-Cradle.It will send message and warn the parents about their child.The E-cradle has also got some sensor which let child self absorbed.

Also the child orphanage will be greatly bnefited by this initiative where the no. of children is so high and there is less people to care them so,I think this E-Cradle concept will help the people of orphanage to have better care of children.

**3.External interface Requirements**

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

**3.2.1User interfaces**

The mobile ,smartphone is used as interface between user and system.The smartphone has app which notify user.

**3.2.2Hardware interfaces**

****

The **Bluetooth** is used to connect to the smartphone and thus message could be sent.It is reliable and has multi functional.



The **wifi interface** is used to connect the microcontroller with internet so that information about e\_cradle could be accessed through app on the smartphone.The data of the sensor is sent to cloud from where it is uploaded on the app.

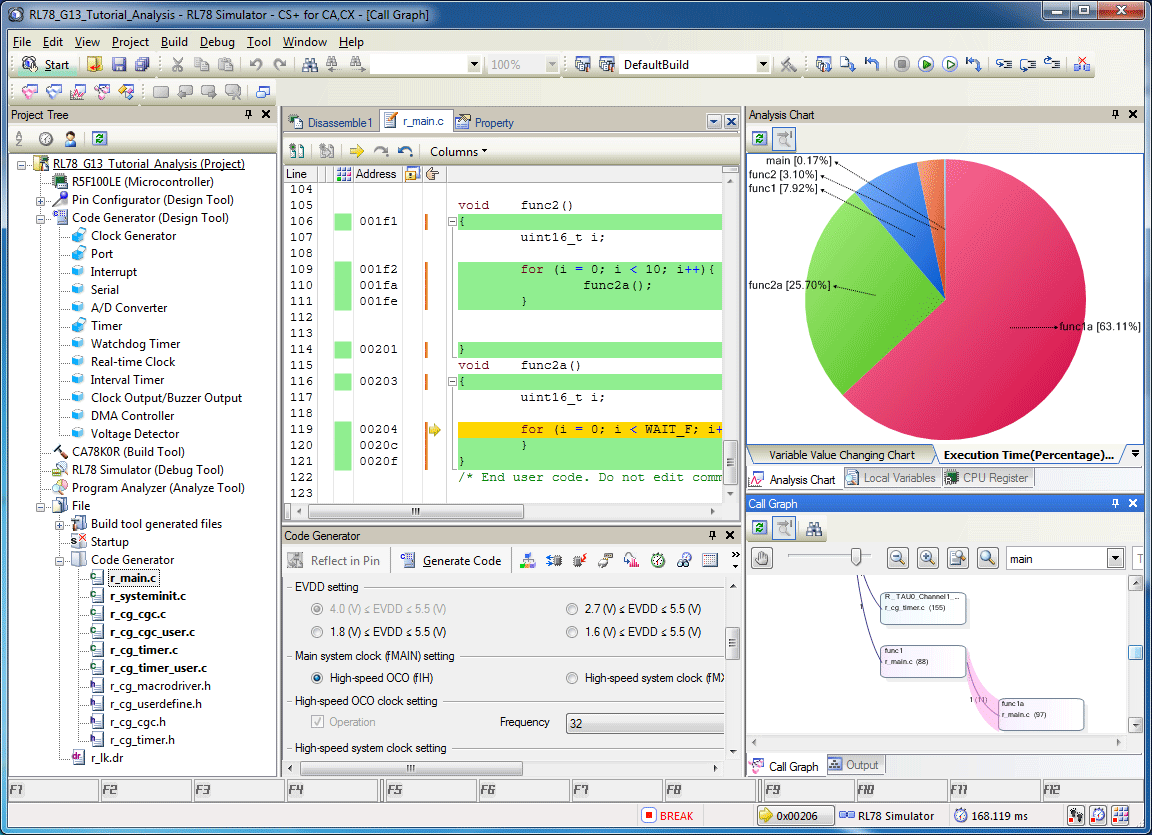
**RL78 series 16-bit Microcontroller**

RL78 is a 16-bit CPU core by Renesas Electronics with a CISC architecture for embedded microcontrollers developed and manufactured by Renesas Electronics and introduced in 2011.It has input and output pin.

**Dc Motor**

It is used to rotate the cradle so that boy could stop crying.it takes very less power and run efficiently.

**3.2.3Software interfaces**

****

CubeSuite +,provides the graphical means of interaction for programmer and user.It also act as means of interaction between user and the hardware.it also provide the interface to run the code to be used for microcontroller.

**3.2.4Communications interfaces**

The communication between the different parts of the system is important since they depend on each other. However, in what way the communication is achieved is not important for the system and is therefore handled by the underlying operating systems for both the mobile application and the web portal.They communicate and provide the required result.The Bluetooth and wifi helps in communication and have better understanding of system.

**3.3Functional requirements**

**3.3.1.1Functional requirement 1.1**

A user should be able to get the notification on the email or the some sound produced by the app if some problem occur.The user should get message regarding the E-cradle.Like if the child cry for long time and the wetting of pants of the child.

**3.3.2.Functional requirement 1.2**

The cradle should be installed with cry detector and temperature sensor which notify user about the condition.It should read the data and send the data to the cloud and further upload to the app.

**3.3.3Functional requirement 1.3**

The Cradle should have Two Dc motor ,one for the swinging the cradle if child start crying and the second for the toy tied up at the top.

**3.3.4Functional requirement1.4**

The cloud is the platform which store the data on virtual memory and and get the data from sensor and send the data to the email or cloud and then further to the app.

**3.4.Software system attributes:**

**3.4.1Reliability**

GIST: The reliability of the system. SCALE: The reliability that the system gives the right result when temperature change. METER: Measurements obtained is in some range.. MUST: More than 98% of change should be catched.. PLAN: More than 99% of the searches. WISH: 100% of the changes.Also the sound detected should have less range.

**3.4.2 Availability**

SystemAvailability GIST: The availability of the system when it is used. SCALE: The average system availability (not considering network failing). METER: Measurements obtained from 1000 hours of usage during testing. MUST: More than 98% of the time. PLAN: More than 99% of the time. WISH: 100% of the time.

**3.4.3 Security**

CommunicationSecurity GIST: Security of the communication between the system and server. SCALE: The messages should be encrypted for log-in communications, so others cannot get user-name and password from those messages. METER: Attempts to get user-name and password through obtained messages on 1000 log-in session during testing. MUST: 100% of the Communication Messages in the communication of a log-in session should be encrypted. Communication Messages: Defined: Every exchanged of information between client and server.

**3.4.5 Maintainability**

TITLE: Application extendibility DESC: The application should be easy to extend. The code should be written in a way that it favors implementation of new functions. RAT: In order for future functions to be implemented easily to the application.

**3.4.6 Portability**

Application portability DESC: The application should be portable with iOS and Android. RAT: The adaptable platform for the application to run on.

**3.5.Performance requirements**

The requirements in this section provide a detailed specification of the user interaction with the software and measurements placed on the system performance. First, this application will be used by a single useror multiple user. There will be no multiple user handling since the application runs on a single portable device using any network. The amount of the input is moderate since the input data of the application is the varying temperature data,sound data,running of motor and RFID data so that child movement and presence coud be traacke down. We have to monitor changes in the temperature of bed and notify the user.The information among the data will be achieved by reducing input into a set of features of the interested scenario.The major issue here is the application should answer in real-time, namely, the recognizing and labeling operations has to be handled in less than 1 second. Also the application should be able to recognize changes in the data.

**3.6.Database requirement**

The requirement is used very less or not required. Only the data variation is used so that if the corresponding changes comes in the status of cradle,it notify the user.

**3.7. Design constraints**

This section includes the design constraints on the software caused by the hardware. We coded program in python,which provide multiple design.The portability of the system depends on portability of the devices we have used. Since the system will run on a single machine and it will not be dependent to a bigger system or any other computers, there will not be any security or reliability problems. E-cradle project is designed to be extensible, that is open to changes. Whenever new patterns are added to the memory of the program, by the help of some additions, the application will handle recognizing them.

**3.8 Other Requirement**

**3.8.1 Hard drive space**

Hard Drive Space GIST: Hard drive space. SCALE: The application’s need of hard drive space. METER: MB. MUST: No more than 20 MB. PLAN: No more than 15 MB. WISH: No more than 10 MB. MB: DEFINED: Megabyte

**3.8.2 Application memory usage**

Application Memory Usage GIST: The amount of Operate System memory occupied by the application. SCALE: MB. METER: Observations done from the performance log during testing MUST: No more than 20 MB. PLAN: No more than 16 MB WISH: No more than 10 MB Operate System: DEFINED: The mobile Operate System which the application is running on. MB: DEFINED: Megabyte.