

**Software Requirement Specification**

**For**

**TeachingHub-BD**

**Version 1.0**

**Submitted by**

Badhon Kumar Sarkar (172-35-2092)

**Submitted to**

ASIF KHAN SHAKIR

Senior Lecturer

Department of Software Engineering

Date:

**Table of Contents**

[1. Introduction 4](#_Toc55226826)

[1.1 Purpose 4](#_Toc55226827)

[1.2 Intended Audience and Reading Suggestions 4](#_Toc55226828)

[1.3 Documentation Convention 4](#_Toc55226829)

[1.4 Product Scope 4](#_Toc55226830)

[1.5 Overview 5](#_Toc55226831)

[2. User Classes & Characteristics 5](#_Toc55226832)

[2.1 Owner 5](#_Toc55226833)

[2.2 Parents 5](#_Toc55226834)

[2.2.1 New Parents 5](#_Toc55226835)

[2.2.2 Registered Parents 5](#_Toc55226836)

[2.3 Manager 6](#_Toc55226837)

[2.4 Employee 6](#_Toc55226838)

[2.4.1 New Employee 6](#_Toc55226839)

[2.4.2 Assigned Employee 6](#_Toc55226840)

[3. Implementation Constrain & Design 6](#_Toc55226841)

[3.1 Operating Environment: 6](#_Toc55226842)

[3.2 Software Language Used: 6](#_Toc55226843)

[3.3 Development Tools 6](#_Toc55226844)

[3.4 Database Support 7](#_Toc55226845)

[4. Users Requirements 7](#_Toc55226846)

[4.1 Functional Requirements 7](#_Toc55226847)

[4.1.1 User Management 7](#_Toc55226848)

[4.1.2 Package Management 8](#_Toc55226849)

[4.1.3 Payment Directory 8](#_Toc55226850)

[4.1.4 Performance Handling 8](#_Toc55226851)

[4.1.5 Attendance Management 8](#_Toc55226852)

[4.2 Non-Functional Requirement 9](#_Toc55226853)

[4.2.1 Security 9](#_Toc55226854)

[4.2.2 Maintenance 9](#_Toc55226855)

[4.2.3 Performance 9](#_Toc55226856)

[4.2.4 Reliability 9](#_Toc55226857)

[5. Requirement Engineering process 9](#_Toc55226858)

[5.1 Feasibility study 9](#_Toc55226859)

[5.1.1 Technical Feasibility 9](#_Toc55226860)

[5.1.2 Financially Feasibility: 10](#_Toc55226861)

[5.1.3 Scheduling Feasibility 10](#_Toc55226862)

[5.1.4 Legal Feasibility 10](#_Toc55226863)

[5.2 Cost Analysis 10](#_Toc55226864)

[5.3 Schedule Analysis 11](#_Toc55226865)

[6. Uses Case Diagram 12](#_Toc55226866)

[6.1 Use case for Admin 12](#_Toc55226867)

[6.2 Use case for Manager 14](#_Toc55226868)

[6.3 Use case for Employee 16](#_Toc55226869)

[6.4 Use case for Parents 18](#_Toc55226870)

[7. Activity Diagram 19](#_Toc55226871)

[7.1 Activity Diagram for Admin 19](#_Toc55226872)

[7.2 Activity Diagram for Manager 20](#_Toc55226873)

[7.3 Activity Diagram for Employee 21](#_Toc55226874)

[7.4 Activity Diagram for Parent 22](#_Toc55226875)

[8. Sequence Diagram 23](#_Toc55226876)

[8.1 Sequence Diagram for Admin 23](#_Toc55226877)

[8.2 Sequence Diagram for Manager 24](#_Toc55226878)

[8.3 Sequence Diagram for employee 25](#_Toc55226879)

[8.4 Sequence Diagram for parent 26](#_Toc55226880)

[9. Class Diagram 27](#_Toc55226881)

# Introduction

## Purpose

The purpose of this document is to describe the overall description of the project. In this report, we mention all the functional and non-functional requirements of the system. We also describe the interfaces of the project and also write our constraints, goals, etc. In this report, there are also diagrams such as the use case diagram, activity diagram.

## Intended Audience and Reading Suggestions

This document is made for whole project team. User also can read this. This report is helpful for developers as well as project manager as they can easily know the entire system’s functions and requirements. By reading this report developers can identify their main concern easily.

## Documentation Convention

This SRS is divided up into sections detailing an overall description, the external interface requirements, system features, and other non-functional requirements. As this is the final draft, any future modifications of this document would involve adapting the product to changing systems and uses. We hope to have the product evolve to changing times as to ensure continued use and success. The Document and Specification team have prepared the overall information in this document to the best of their ability. Once read, it is evident that each section is important to the overall SRS and significant to the project in its own right.

## Product Scope

The proposed software product is Baby care. This system can be use babysitter finder. The project is related to babysitter & parents. For this we will have a database server. Any parents can place an appointment of their desire babysitter through using our system. They can also know nearby baby care’s location.

## Overview

This Software Requirements Specification (SRS) specifies all the requirements for Baby care. The objective of this document therefore is to formally describe the system’s high-level requirements including functional requirements, non-functional requirements, and constraints. The detailed structure of this document is organized as follows:

Section 2 of this document provides an overview of the domain that the proposed software Baby care will support. These include a general description of the product, user characteristics, general constraints, and assumptions for this system. To contrast, in section 3 all the design and Implementation constraints are given as well the Assumptions and Dependencies are described concisely in the Section 5. Section 6 includes the system features having all the functional requirements along with their rationale. Section 7 presents the details of the external interface requirements. Finally, section 9 contains the other non-functional requirements.

# User Classes & Characteristics

## Owner

The owner will have all kinds of powers. He/she can appointment a manager at any time and dismiss from the manager if he/she wants. Again, any employee can be hired and dismissed. He can control everything. Managers, employees, parents and kids can see the information. The owner can determine what package is running here. If a package has to be changed, he/she can do it.

## Parents

### New Parents

They can see the location of day care center. He/she can see all facilities and packages. Could contact the manager. And if they like, they can register and join.

### Registered Parents

Register parents will need to login. They can change their packages at any time. They can find information about their child. If they want, they can also take-home maid. Can communicate with the manager and the owner through this system.

## Manager

Manager must be login. Manager can add employee, view employee or remove employee. Manager also can add & view baby’s information. Baby care’s entire package are customized by manager. He can just view all payment information & billing system. Set baby care’s discount package. Most of the features of this system will handle by manager. Manager have the power of control maximum information.

## Employee

### New Employee

New employee can search job & apply

### Assigned Employee

Assigned employee must login. Take baby’s attendance. Give feedback about baby & upload feedback our system. Contact baby’s parents. Employee can see baby’s list. Check complains against his/her.

# Implementation Constrain & Design

## Operating Environment:

The Baby care center will be web-based system. Thus, anyone having a browser can hit the specific link and can get access to it. Thus, it will ensure its best usage and will ease the means of getting access to the system. Moreover, it will remove the complexities of running the system in multiple platforms, as it will be deployed in a web server.

## Software Language Used:

The application will develop using Laravel 6.

## Development Tools

For the development, purpose Microsoft Visual Studio 2019 Professional edition will use. For handling different database operations MySQL will use.

## Database Support

The database that will be used is MySQL**.**

# Users Requirements

## Functional Requirements

When start the project we needed the comprehensive engagement and lighting quick coordination with the stakeholder. After that based on stakeholder and user need, we study about the requirement. And gather all the requirements. After that we make a list with all the requirements. Here is the requirement for our stakeholder who will use the system.

### User Management

Owner will have the main power who will add without registration. Owner will add manager and manager will add employee.

|  |  |
| --- | --- |
| Requirement No | Requirement |
| UM-001 | Owner will login with his given password and username. |
| UM-002 | Manager will add by owner. And manager also have username and password |
| UM-003 | Password will be encrypted. It can be MD5 or encrypt any hash function |
| UM-004 | Manager will add all employee and will provide his username and password. |
| UM-005 | Parents will be registered after confirmation and can login with his username and password |
| UM-006 | Used should be approved by manager |

### Package Management

In this system, there will be different type of packages.

|  |  |
| --- | --- |
| Requirement No | Requirement |
| PM-001 | Manager will define all package |
| PM-002 | Package will have also discount scheme |
| PM-003 | Package can be updated by manager |
| PM-004 | All package will visible to user |
| PM-005 | Admin can give his feedback about packages |
| PM-006 | Which baby included which package there will be a list |
| PM-007 | User can update his package about his need |

### Payment Directory

All payment system will be integrated in this system

|  |  |
| --- | --- |
| Requirement No | Requirement |
| PD-001 | Baby Care center’s all payment system |
| PD-002 | Parents can take a package to admit their baby. And the admit payment system |
| PD-003 | Employee monthly payment list down |
| PD-004 | All information of payment for every baby. Due or advance payment system also include |
| PD-005 | All package’s payment also attaches |

### Performance Handling

Employee performance and all baby’s performance will monitor. Based on this many things are depended

|  |  |
| --- | --- |
| Requirement No | Requirement |
| PD-001 | Employee performance measure based on parents’ feedback |
| PD-002 | Parents feedback will list down in a table base on employee |
| PD-003 | Babies progress report will be created by employee |
| PD-004 | Parents can check his/her baby’s performance |
| PD-005 | Manager performance will be monitored by owner |

### Attendance Management

Employee will define a schedule for baby’s and take attendance. In addition, manager will be scheduled time for employee

|  |  |
| --- | --- |
| Requirement No | Requirement |
| AM-001 | Employee will count baby’s attendance |
| AM -002 | Manager will count employee’s attendance |
| AM -003 | Manager can view baby’s and employee’s attendance |
| AM -004 | Owner can see all attendance (Employee, Manager, Baby) |
| AM -005 | Babies attendance will divide based on package |

## Non-Functional Requirement

### Security

|  |  |
| --- | --- |
| **NR-01** | This system is safe and secure for human. |
| **Description** | Nobody will harm with this system. And everyone’s data will be safe and password will always be encrypted. |
| **Stakeholder** | Owner, Parents, Manager, Employee |

### Maintenance

|  |  |
| --- | --- |
| 1. **NR-02** | Update will generate in a schedule and will fix all |
| **Description** | This system will always up to date and try to solve all bug in a short time |
| **Stakeholder** | Owner of center |

### Performance

|  |  |
| --- | --- |
| **NR-03** | No internal legacy will face. |
| **Description** | All option and feature will be in a smooth view. No delay more than 5 second. |
| **Stakeholder** | Owner, Parents, Manager, Employee |

### Reliability

|  |  |
| --- | --- |
| **NR-03** | All time service available |
| **Description** | This system must be performing well every day. Except further issues or update system |
| **Stakeholder** | Anonymous |

# Requirement Engineering process

## Feasibility study

### Technical Feasibility

We are a team of two person to do this project. We are working in ASP .Net framework using c#. As we all have personal computer to do our project and we have all software for this project. Also, we need a database to store data for our project. We manage a server for database. So, we think we are technically feasible for this project.

### Financially Feasibility:

We estimate our all cost for this project such as cost for software, hardware, project manager, project tester, and we need to pay for our database server. Our cost estimate is given in figure (). Initially we think we are also feasible for financial support.

### Scheduling Feasibility

We make a time estimation in figure 2 for a four-month duration project. This project starts with requirement gathering. All work such as development, testing the software is scheduled. Hope this also feasible.

### Legal Feasibility

Our project is based on babies and their parent. This is related to country’s welfare. And here is no part which is against our country’s law. All user can use this with their comfort. For more we can take a legal license from government.

## Cost Analysis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Items | Unit | Cost/unit | Total | % of total |
| 1. Hardware | **-** | **-** | $300 | 5.26% |
| 1. Software | **-** | - | $250 | 4.39% |
| 1. Visual studio |  | $100 | $100 | 1.75% |
| 1. Windows 10 |  | $150 | $150 | 2.63% |
| 1. Developer | 5 | $500 | $2500 | 43.86% |
| 1. Testing | 2 | $400 | $800 | 14.04% |
| 1. Designer | 2 | $300 | $600 | 10.53% |
| 1. Project Manager | 1 | $1000 | $1000 | 17.54% |
| **Total: $5700** | |

## Schedule Analysis

We will estimate the whole project time required by using PERT time analysis methods. In this method, to = minimum time required for that activity, tp = maximum time required for that activity, tm = neither optimistic nor pessimistic time required & te = PERT’s average time.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Activity | to | Number of Days  tm | tp | to+4tm+tp  te=--------------  6 |
| Requirement Collection | 2 | 4 | 6 | 4 |
| Feasibility Study | 3 | 5 | 8 | 5.1667 |
| Drawing Diagrams | 2 | 4 | 8 | 4.3334 |
| UI/UX design | 6 | 8 | 12 | 8.3333 |
| MVC Architecture Design | 8 | 12 | 15 | 11.8333 |
| Programming | 15 | 19 | 24 | 19.1667 |
| Testing | 4 | 6 | 9 | 6.1667 |

# Uses Case Diagram

## Use case for Admin



|  |  |  |
| --- | --- | --- |
| **Use case** | Center Owner | |
| **Goal** | See all information about his day care center. He can manage his manager, employee and other user. He will add his manager to manage whole system. See feedback | |
| **Actor** | Owner | |
| **Pre-condition** | Need to login in the system with username and password | |
| **Post condition** | After his confirmation manager can login. | |
| **Trigger** | Owner features will come after his login. At first the username and password will serve by developer after he can change those. | |
| **Description** | Step | Action |
| 1.1  1.2  1.3 | Add manager  Set username password for manager  Monitor users |
| **Alternative Flows** | No alternative if username and password is forgotten. Meet to developer | |
| **Quality Requirement** | Set an attractive interface and less load time | |

## Use case for Manager



|  |  |  |
| --- | --- | --- |
| **Use case** | Manager | |
| **Goal** | To handle all employee and parents. Can add employee | |
| **Actor** | 1.Primary actor: Manager  2.Secondary actor: Employee, Parents | |
| **Pre-condition** | Login with owners given username password | |
| **Post condition** | No employee can do his/her work without manager confirmation. Also, no parents are available to take a package without manager advice | |
| **Trigger** | This feature will work after a manager assigned by owner | |
| **Description** | Step | Action |
| 1.1  1.2  1.3  1.4  1.5 | Add employee  Give feedback for employee  Help parent to take a package  Set up package  Update all package when needed |
| **Alternative Flows** | Employee and parents can contact with owner. Owner can handle | |
| **Quality Requirement** | Many works for manager. So not more time for do any work. An easy and comfort interface require. | |
|  |  | |

## Use case for Employee



|  |  |  |
| --- | --- | --- |
| **Use case** | Employee | |
| **Goal** | Observe and serve all baby and give feedback of pre schooling | |
| **Actor** | 1.Primary actor: Employee  2.Secondary actor: Manager, Owner | |
| **Pre-condition** | Must be assigned by manager. And understand all responsibility | |
| **Post condition** | Give babies feedback about their per schoolwork though parent can monitor their baby. | |
| **Trigger** | After manager add and give the username password | |
| **Description** | Step | Action |
| 1.1  1.2  1.3  1.4 | Observe all baby  Monitor baby’s pre-school performance  Contact with parents  Contact with owner for manager feedback/complain |
| **Alternative Flows** | Employee can meet parents and say their thought by contacting with them | |
| **Quality Requirement** | Feasible to work with the system. Easy interface. In addition, private chat with owner for further compliance. | |

## Use case for Parents



|  |  |  |
| --- | --- | --- |
| **Use case** | Parent | |
| **Goal** | Find day care center, choose package, Monitor babies | |
| **Actor** | 1.Primary actor: Registered parents  2. Secondary actor: Anonymous | |
| **Pre-condition** | Choose a package and take admission for baby | |
| **Post condition** | Monitor baby and give feedback about employee | |
| **Trigger** | Must be registered with package fee | |
| **Description** | Step | Action |
| 1.1  1.2  1.3  1.4  1.6  1.7 | Find a center  Choose package  Get registered  Send feedback  Monitor baby  Suggest better package |
| **Alternative Flows** | N/A | |
| **Quality Requirement** | Make easy to find center and package. User-friendly interface. | |

# Activity Diagram

## Activity Diagram for Admin



## Activity Diagram for Manager



## Activity Diagram for Employee



## Activity Diagram for Parent



# Sequence Diagram

## Sequence Diagram for Admin



## Sequence Diagram for Manager



## Sequence Diagram for employee



## Sequence Diagram for parent



# Class Diagram

