**Software Architecture Document**

**1.****Introduction**

**1.1****Purpose**

This document provides a comprehensive architectural overview of healthcare mobile based application system module1 named as Parent Registration, using a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions which have been made on the system.

**1.2****Scope**

This Software Architecture Document provides an architectural overview of the Parent-Registration System. The Parent-Registration System is being developed by different technical to support online parent community for free parenting and pregnancy care application. It is a well thought out and executed app that aims to make parent's lives easier and less stressful. This Document has been generated directly from the Parent-Registration Analysis & Design Model implemented in GitHub.

**2.****Architectural Representation**

This document presents the architecture as a series of views; use case view, logical view, process view and deployment view. There is no separate implementation view described in this document. These are views on an underlying Unified Modelling Language (UML) model.

**3.** **Use-Case View**

A description of the use-case view of the software architecture. The Use Case View is important input to the selection of the set of scenarios and/or use cases that are the focus of an iteration. It describes the set of scenarios and/or use cases that represent some significant, central functionality. It also describes the set of scenarios and/or use cases that have a substantial architectural coverage (that exercise many architectural elements) or that stress or illustrate a specific, delicate point of the architecture.

The Parent-Registration use cases are:

- Registration for Parent

-Login

- Child Details Management

- Check Child Status

-Vaccination Management

- Hospital Management Report

- BMI Analyzer

- Child Feeding Control

- First Aid

- Add Reminder

- Add Memos

- Maintain Parent Information

- Maintain Child Information

- Select Category from the list.

- Submit button.

These use cases are initiated by the user after downloading the application on mobile. In addition, interaction with external actors; Parent details and child details occur.

**3.1 Architecturally-Significant Use Cases**

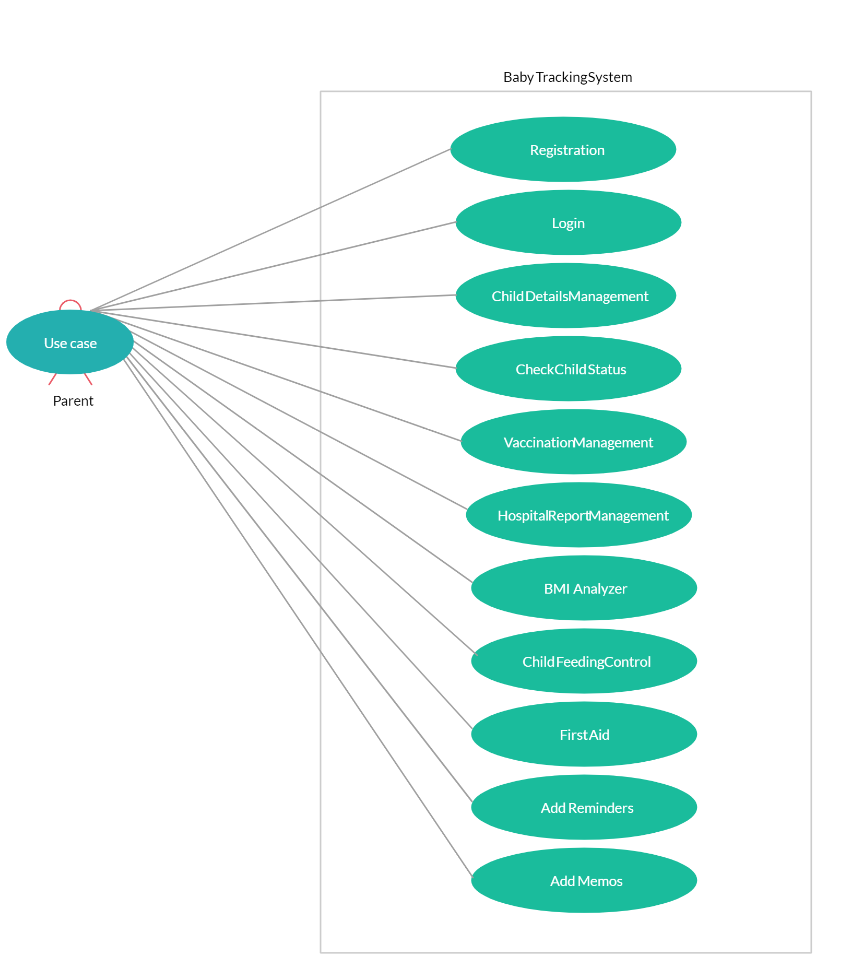


Diagram Name: Architecturally Significant Use-Cases

*4.1.2 Registration*

Brief Description: This use case describes how a Parent logs into the child Registration. The actors starting this use case are child, child name, child age, child gender etc.

*4.1.3 Child Details Management*

Brief Description: This use case allows the register to maintain child information in the registration system. This includes adding, modifying, and deleting professors from the system. The actor of this use case is the Registrar.

*4.1.4 Select child status*

Brief Description: This use case allows a child to select the child status offerings (DOB specific will be given) from the date calendar for that he/she is eligible. The actor starting this use case is the child. The DOB catalogue is an actor within the use case.

*4.1.5 Vaccination Management Module*

Brief Description: This use case allows a child get reminder for the vaccination according to the date of birth set up in the system. The parent can also modify or delete Date of birth DOB. The DOB Catalogue provides a list of all the vaccination will be given to the child. The main actor of this use case is the child DOB.

*4.1.6 View Hospital Report Management*

Brief Description: This use case allows a child to view his/her report card for the previously completed months. The child growth date is the actor of this use case. It manages the smooth healthcare performance along with administrative, medical, legal and financial control. That is a cornerstone for the successful operation of the healthcare facility.

*4.1.7 BMI Analyzer*

Brief Description: This use case allows user BMI report submit child. BMI is also known as **Body Mass Index (BMI)** is a common method used to assess the health of an individual by comparing the amount of weight they carry to the height of the individual. BMI is calculated simply by dividing a person’s weight in kilograms by their height: BMI = kg/m2 . The actor in this use case is the child weight.

*4.1.8 Child Feeding Control Module*

Brief Description: This use case allows mother to include food eating habit and restriction and also to maintain mother information in the registration system The actor for this use case is the Mother food eating detail. It is a tool that is widely used to measure parental **controlling feeding** practices, including pressure to eat and restriction. Pressure to eat denotes parental enforcement of practices that aim to increase a **child's** consumption of food.

*4.1.9* First Aid *Module*

Brief Description: This use case allows mother to include food eating habit and restriction and also to maintain mother information in the registration system is the Mother food eating detail. It is a tool that is widely used to measure parental controllingfeeding practices, including pressure to eat and restriction. Pressure to eat denotes parental enforcement of practices that aim to increase a child's consumption of food.

*4.1.9 Child Feeding Control Module*

Brief Description: This use case allows mother to include food eating habit and restriction and also to maintain mother information in the registration system The actor for this use case is the Mother food eating detail.

*4.1.9 Add Reminder Module*

Brief Description: This use case helps parent to remind the vaccine chart reminder for their chart. It also helps to maintain child vaccination and also other reminder information in the registration system.

*4.1.9 Add Memos Module*

Brief Description: This use case helps parent to check the disease history, test results, as well as new data to the child name file.

**5.****Logical View**

A description of the logical view of the architecture. Describes the most important classes, their organization in service packages and subsystems, and the organization of these subsystems into layers. Also describes the most important use-case realizations, for example, the dynamic aspects of the architecture. Class diagrams may be included to illustrate the relationships between architecturally significant classes, subsystems, packages and layers.

The logical view of the course registration system is comprised of the 3 main packages: User Interface, Business Services, and Business Objects.

The User Interface Package contains classes for each of the forms that the actors use to communicate with the System. Boundary classes exist to support login, maintaining of schedules, maintaining of child info, Login details, submitting child details, maintaining child and parent info etc.

The Business Services Package contains control classes for interfacing with the billing system, controlling child registration, and managing the child details.

The Business Objects Package includes entity classes for the module-based artifacts and boundary classes for the interface with the Child details Catalogue System.