# **SOFTWARE PROJECT PLAN**

**0.1.1 Definition**

The Insulin pump, the patient can inject insulin/glucagon to his body on a regular basis

which is called as the Basal rate and also at certain times it is called as Bolus rate. Special

conditions like sleeping or exercising can be specially configured to deliver appropriate

hormone for regular functioning of the body. Bolus is considered when an extra dosage of any

hormone is required. For example, if the patient has consumed extra food.

**0.1.2. Description**

**1. Introduction**

a. Project Scope

The goal of this Project is to develop an application that simulates the behaviour of an Insulin Pump. The simulation of the Insulin pump gives results based upon the behaviour of a human’s blood sugar

level under influences like insulin/glucagon injections, food and increased physical work.

b. Major software functions

Functional Requirements

There are three different modes in our software.

1. Patient mode

This mode is intended for the patient. Ideally, the blood sugar level of the patient is read from the sensor. But here, we simulate it as a manual entry from the user. The patient can view the insulin level, glucagon level and the battery level of the pump. He will also get warnings. The following are the different functions associated with this mode.

* 1. Analyse the Blood Sugar Level Reading and identify the band.
  2. Identify the hormone to be injected
  3. Calculate Dosage of Insulin/Glucagon
  4. Maintain the blood sugar level of the patient in the safe band

1. Doctor Mode

This mode is intended for the doctor. He can view the patient’s history, edit the patient’s details and also set the limit for the insulin or glucagon. The following are the different functions associated with this mode.

1. Patient details Entry
2. Set the daily dosage limit of Insulin / Glucagon
3. View the Insulin / Glucagon Intake of the patient
4. Battery Level Indication
5. Insulin/Glucagon Level Indication
6. Insulin or Glucagon level injection
7. Simulation Mode

This mode is for the tester. He will manually enter the blood sugar value and the Insulin and Glucagon value to see if the graph moves in the intended path.

Non- Functional Requirements

Safety Requirements

1. Warnings: Proper warnings must be given to the user.
   * Battery Low Warning
   * Insulin / Glucagon level Low Warning
2. Check for mathematical error.

If the dosage value results in some very large no (due to a mathematical error), the patient must not be given the large dose. Giving nothing or giving a large dose may endanger his life. Instead, he should be given a safe dose. If due to some reasons, the logic module fails to compute a dosage, the same must be done.

Safety Requirements

1. PIN protection

Unauthorized access to the doctor’s module can tamper the patient data. This can endanger the patient’s life. So, the doctor’s mode is PIN protected.

c. Performance/Behaviour issues (JITHU)

d. Safety and reliability targets

Following points are taken into account to ensure the safety and reliability for the pump.

1. If the calculation is done after a large time, then the system is not reliable and it is also not safe.
2. If the pump doesn’t alert the patient when the insulin or glucagon levels falls below a threshold, then it is not safe.
3. The system is reliable if after injecting insulin, graph (blood sugar level) goes down.
4. The system is reliable if after injecting Glucagon, graph (blood sugar level) goes up.

e. Management and technical constraints

Management Constraints:

We are following the Agile model. So, it requires daily stand-up meetings. But having a team meeting everyday is not possible and hence is a constraint. But we meet twice a week.

Technical Constraints:

1. For simulation, we have decided to use Matlab since it is the most powerful and popular tool. But it requires license.
2. Study of the feasibility of interfacing the simulation tool with the java program.

f. Process Model (SAMIR)

**2. Project Estimates**

a. Historical data used for estimates

* Insulin requirements are required to respond to the decrease or increase in sensitivity to insulin in an effort to maintain normoglycemia.
* Collected CGM (continuous glucose monitor) data serves as our historical data source. The historical input data (meals, insulin, exercise) is motivated by the fact that output data allows us to focus on modelling.

b. Estimation techniques applied and result (Functional point)

We are Functional points to estimate our project.

No of ILF (Internal logical files) = 5 =>5\*7=35

No of ELF (External logical Files) = 3 =>3\*7=21

No of EO (External Output) =3 =>3\*3=9

No of EI (External Input) = 3 =>3\*4=12

Calculation: -

1)UFP (Unadjusted Function Point) = (35+21+9+12) =77

2)TDI (Total Degree of Influence of the 14 General system characteristics)

1. Data Communication (2)

2. Distributed Data Processing (0)

3. Performance (4)

4. Heavily Used Configuration (4)

5. Transaction Role (0)

6. Online Data Entry (3)

7. End-User Efficiency (5)

8. Online Update (3)

9. Complex Processing (3)

10. Reusability (1)

11. Installation Ease (3)

12. Operational Ease (5)

13. Multiple Sites (0)

14. Facilitate Change (0)

(2+0+4+4+0+3+5+3+3+1+3+5+0+0) = 28

3)VAF (Value added Factor) = 0.65 + (0.01 \* TDI) =>0.93

4)FP (Functional Points) = UFP\*VAF

77\*0.93=71.61

5)Effort in Person Month = FP divided by no. of FP's per month

71.61/60=1.19

6)Schedule in Months = 3.0 \* person-month^1/3

3\*1.19^1/3=>3\*1.05

=3.1months

c. Safety Plan

* Daily dosage limit
* Dosage limit at time
* It alerts User when threshold is exceeded

**4. Risk Management**

(a) Project Risks

(b) Risk Table

(c) Overview of Risk Mitigation, Monitoring, Management

**5. Project Schedule** (SAMIR)

(a) Project task set

(b) Functional decomposition

(c) Task network

(d) Timeline chart

**6. Staff Organization** (NAMITHA)

(a) Team structure

(b) Management reporting and communication

**7. Tracking and Control Mechanisms** (JITHU)

(a) Quality assurance and control

(b) Change management and control