

## Low Level Design (LLD)

# THYROID DISEASE DETECTION

Revision Number: 1.6

Last date of revision: 5<sup>th</sup> November 2021

Amber Gupta

## Document Version Control

Date Issued	Version	Description	Author
4 <sup>th</sup> Aug 2021	1.1	First Draft	Amber Gupta
4 <sup>th</sup> Aug 2021	1.2	Added Workflow chart	Amber Gupta
4 <sup>th</sup> Aug 2021	1.3	Added Exception Scenarios Overall, Constraints	Amber Gupta
5 <sup>st</sup> Aug 2021	1.4	Added user I/O flowchart	Amber Gupta
5 <sup>st</sup> Aug 2021	1.5	Added dataset overview and updated user I/O flowchart.	Amber Gupta
5 <sup>st</sup> Aug 2021	1.6	Restructure and reformat LLD	Amber Gupta

## Contents

Document Version Control	2
<b>Abstract</b>	4
1 Introduction	5
1.1 Why this Low-Level Design Document?	5
1.2 Scope	6
1.3 Constraints	6
1.4 1.3 Risks	6
1.5 1.4 Out of Scope	7
2 Technical specifications	7
2.1 Predicting Disease	7
2.2 Logging	7
2.3 Database	7
3 Technology stack	8
4 Proposed Solution	8
5 Model training/validation workflow	9
6 User I/O workflow	10
7 Test Cases	11

## Abstract

Thyroid disease a very common problem in India, more than one crore people are suffering with the disease every year. Especially it is more common in female. Hyperthyroidism and hypothyroidism are the most two common diseases caused by irregular function of thyroid gland. Thyroid disorder can speed up or slow down the metabolism of the body. In the world of rising new technology and innovation, healthcare industry is advancing with the role of Artificial Intelligence. Machine learning algorithms can help to early detection of the disease and to improve the quality of the life. This study demonstrates the how different classification algorithms can forecasts the presence of the disease. Different classification algorithms such as Logistic regression, Random Forest, Decision Tree, Naïve Bayes, Support Vector Machine have been tested and compared to predict the better outcome of the model.

## 1 Introduction

### 1.1 Why this Low-Level Design Document?

The purpose of this document is to present a detailed description of the Thyroid Disease Detection System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed to the higher management for its approval.

The main objective of the project is to predict if a person is having compensated hypothyroid, primary hypothyroid, secondary hypothyroid or negative(no thyroid).

Thyroid Disease Detection is a vital part of health IT and can:

- Contain a patient's medical history, diagnoses, medications, treatment plans, allergies and laboratory and test results.
- Allow access to evidence-based tools that providers can use to make decisions about a patient's care.
- Automate and streamline provider workflow

A Thyroid Disease Detection System contains patient health information, such as:

- Patient demographics
- Patient's age
- Vital signs
- Medical histories
- Diagnoses
- Medications
- Lab and test results

This project shall be delivered in two phases:

Phase 1: All the functionalities with PyPi packages.

Phase2: Integration of UI to all the functionalities.

## 1.2 Scope

This software system will be a Web application This system will be designed to detect the thyroid disease and whether it is a compensated, primary or secondary hypothyroid for better disease management, improved interventions, and more efficient health-care resource allocation. More specifically, Early detection of any preventable diseases is important for better disease management. This system is designed to predict the thyroid disease from patient information such as age, disease history, lab results, procedures and medications.

## 1.3 Constraints

The thyroid disease detection application must be user friendly, as automated as possible and users should not be required to know any of the workings.

## 1.4 Risks

Document specific risks that have been identified or that should be considered.

## 1.5 Out of Scope

Delineate specific activities, capabilities, and items that are out of scope for the project.

## 2 Technical specifications

### 2.1 Dataset

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
id	age	TSH	T3	TT4	T4U	FTI	sex	on_thyroxine	query_on_thyroxine	on_antithyroid_med_sick	pregnant	thyroid surgery	I131 treatment	query hypothyroid	query hyperthyroid	lithium	
0	41	1.3	2.5	125	1.14	109	F	f	f	f	f	f	f	f	f	f	
1	23	4.1	2	102	0.98	107	F	f	f	f	f	f	f	f	f	f	
2	46	0.98	2	109	0.91	120	M	f	f	f	f	f	f	f	f	f	
3	70	0.16	1.9	175	0.98	107	F	t	f	f	f	f	f	f	f	f	
4	70	0.72	1.2	61	0.87	70	F	f	f	f	f	f	f	f	f	f	
5	18	0.03	2	183	1.3	141	F	t	f	f	f	f	f	f	f	f	
6	59	1.4	2	72	0.92	78	F	f	f	f	f	f	f	f	f	f	
8	66	0.6	2.2	123	0.93	132	F	f	f	f	f	f	f	f	f	f	
9	68	2.4	1.6	83	0.89	93	M	f	f	f	f	f	f	f	f	f	
11	67	0.03	2	152	0.99	153	F	t	f	f	f	f	f	f	f	f	
12	71	0.03	3.8	171	1.13	151	F	f	f	f	t	f	f	f	t	f	
13	59	2.8	1.7	97	0.91	107	F	f	f	f	f	f	f	f	f	f	
14	28	3.3	1.8	109	0.91	119	M	f	f	f	f	f	f	f	f	f	
15	65	12	2	99	1.14	87	F	f	f	f	f	f	f	t	f	f	

#### 2.1.1 Thyroid dataset overview

The thyroid dataset consists of a table with 2800 records and 30 features. Features are distributed as 7 continuous features and 23 categorical features. There are a total 7852 patients in the training set and 1476 patients in the test set.

#### 2.1.2 Input schema

Feature name	Datatype	Size	Null/Required
Age	int	3	Required

### 2.2 Predicting Disease

- The system presents the set of inputs required from the user.

- The user gives required information.
- The system then predicts that the user is having thyroid or not. Also, it tells whether a user is having compensated, primary or secondary hypothyroid based on the lab test results provided by the user.

## 2.3 Logging

We should be able to log every activity done by the user.

- The System identifies at what step logging required
- The System should be able to log each and every system flow.
- Developers can choose logging methods. You can choose database logging/ File logging as well.
- System should not be hung even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

## 2.4 Database

System needs to store every request into the database and we need to store it in such a way that it is easy to retrain the model as well.

1. The User gives required information.
2. The system stores each and every data given by the user or receives on request to the database. Database which we have chosen is MongoDB.

## 2.5 Deployment

✓ HEROKU



### 3 Technology stack

<b>Front End</b>	HTML/CSS
<b>Backend</b>	Python Flask
<b>Database</b>	MongoDB
<b>Deployment</b>	Heroku/AWS

### 4 Proposed Solution

The proposed solution for this project is Machine learning algorithms can be implemented to predict the risk of thyroid disease. Considering various features like age, sex, level of thyroxine, anti-thyroid medication as inputs from the web app, the implemented classification model will predict the output as presence or absence of the thyroid.

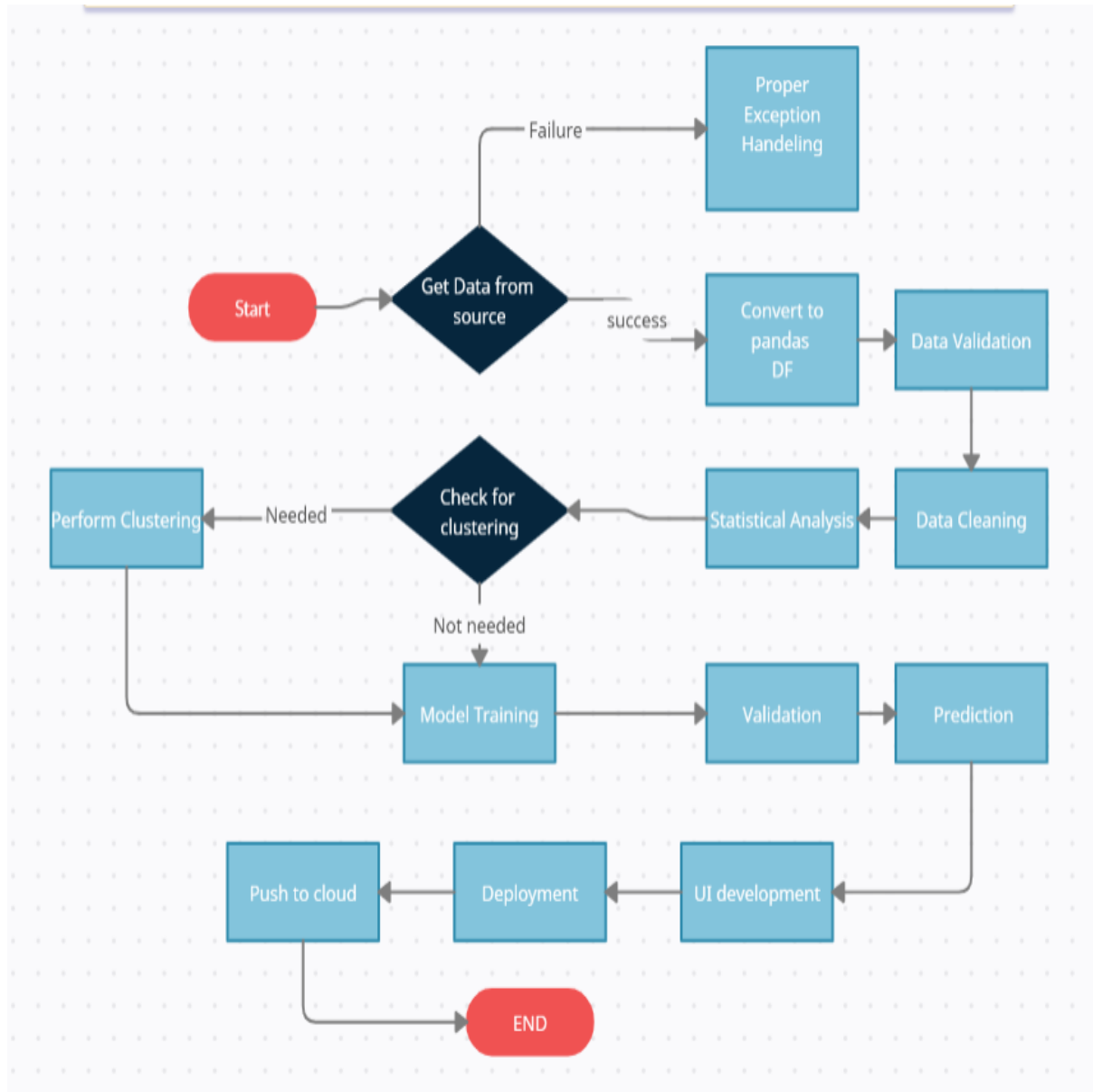
Here, we have used Random Forest Classifier to predict whether the patient is having Thyroid or not.

However, drawing a baseline model is important since it tells us how well other models have performed compared to base model. Here, the base model for Thyroid Disease Detection dataset is Logistic Regression.

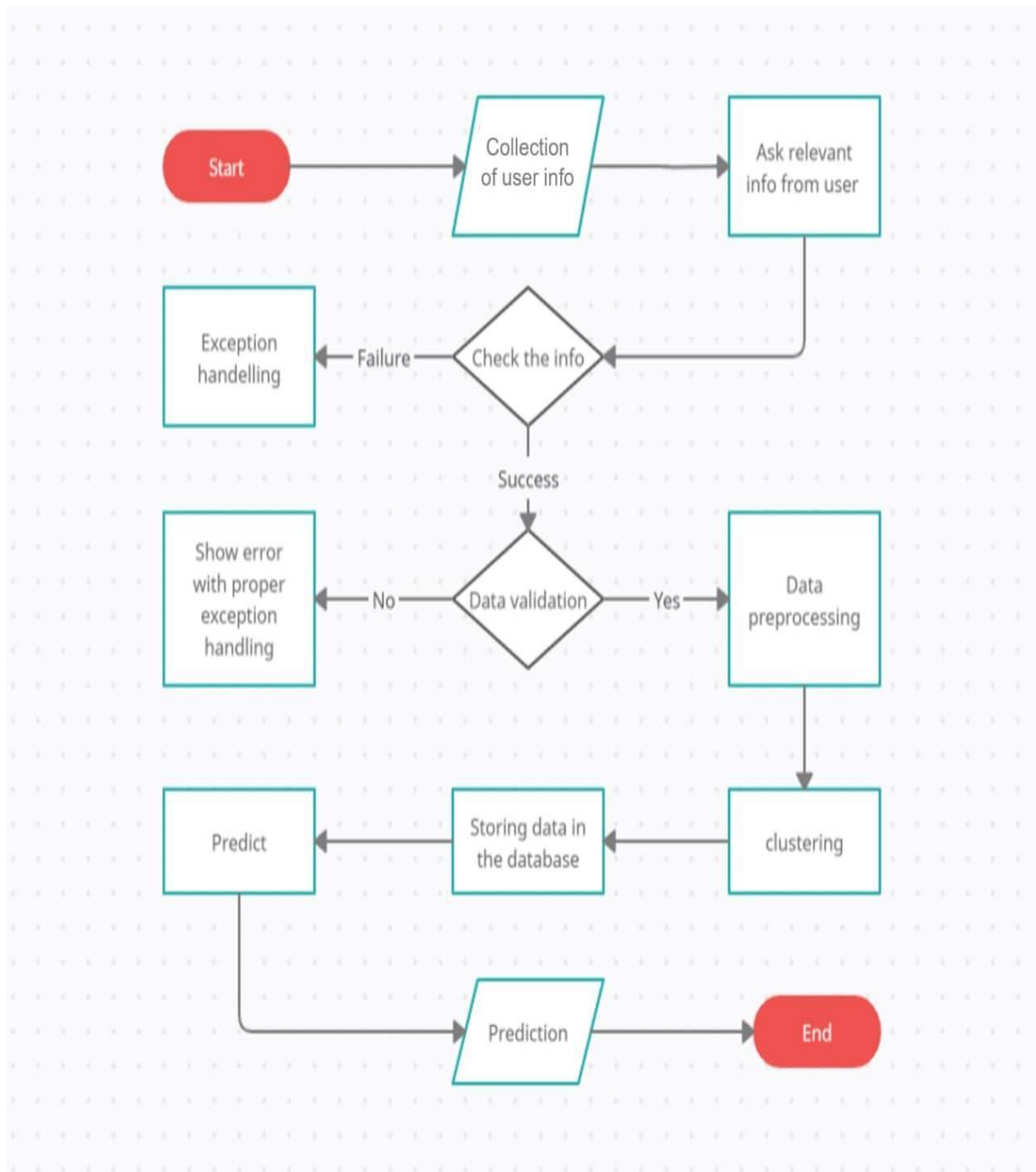
1. Baseline Model : Logistic Regression
2. Actual Model : Random Forest



## 5 Model training/validation workflow



## 6 User I/O workflow



## 7 Test cases

Test case	Steps to perform test case	Module	Pass/Fail

## 8 Performance

We can observe that the accuracy of the predicted output was seen at 97.97% using Random forest classifier. Other classification models such as logistic regression and decision tree have given good accuracy above 90%.