**High Level Design**

**Thyroid Disease Detection**

## **Contents**

[Document Version Control. 2](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250021)

[Abstract.](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250020) 2

1. [Introduction](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250019) 3
   1. [Why this High-Level Design Document?.](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250018) 4
   2. [Scope.](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250017) 4
   3. [Definitions](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250016) 5
2. [General Description.](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250015) 5
   1. [Product Perspective](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250014) 5
   2. [Problem statement](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250013) 6
   3. [PROPOSED SOLUTION](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250012) 6
   4. [FURTHER IMPROVEMENTS](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250011) 6
   5. [Technical Requirements.](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250010) 7
   6. [Data Requirements](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250009) 7
   7. [Tools used.](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250008) 7
      1. Hardware Requirements. 8
      2. Thyroid Disease and Operational Data for Detection 8
   8. [Constraints](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250007) 8
   9. [Assumptions.](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250006) 9
3. [Design Details](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250005) 9
   1. [Process Flow.](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250004) 10
      1. [Model Training and Evaluation](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250003) 10
      2. [Deployment Process](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250002) 10
   2. [Event log](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250001) 10
   3. [Error Handling](file:///C:\Users\Sai\Downloads\UGV_HLD_2.0v.docx#_TOC_250000) 11
4. Performance. 12
   1. Reusability. 13

4.2 Application Compatibility 14

* 1. Resource Utilization 14

4.4 Deployment. 15

5 Conclusion 16

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date Issued | Author | Comment |
| 0.1 | 01/02/2022 | Sai Subhasish | Initial HLD |
|  |  |  |  |
|  |  |  |  |

Abstract

There are types of thyroids and we need to identify if the patient has thyroid or not. If it is a positive case then what type of thyroid the person is suffering from. We need to build a model which will be used by hospitals. In the first case of spatialization the model will predict if the person is suffering from thyroid or not. If the result will come positive, then the treatment will be on fast-track. The doctors will start treating the patients and identify the patient is suffering from hypo-thyroid or hyper-thyroid. If the result will come negative then the patient will be sent to a junior doctor and the junior doctors by using their own expertise they will decide that if the model has done correct prediction or not. Based the prediction comes true then the doctor release the patient. By seeing the readings if doctor analysed that there may be a chance of thyroid then patient sent to the senior doctors.

1. **Introduction**

###### Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

* + - Present all of the design aspects and define them in detail
    - Describe the user interface being implemented
    - Describe the hardware and software interfaces
    - Describe the performance requirements
    - Include design features and the architecture of the project
    - List and describe the non-functional attributes like:

o Security

* + - * Reliability
      * Maintainability
      * Portability
      * Reusability
      * Application compatibility
      * Resource utilization
      * Serviceability

##### 1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

**1.3 Definitions**

|  |  |
| --- | --- |
| Term | Description |
| IDE | Integrated Development Environment |
| Database | Collection of all information monitored by this system |
| AWS | Amazon Web Services |
|  |  |
|  |  |

**2. General Description**

**2.1 Product Prospective**

Thyroid Disease Detection is a machine learning-based disease prediction model which will help us to predict whether the patient has thyroid or not based on the readings taken.

**2.2 Problem statement**

Thyroid disease is one of the most common disease with endocrine disorder in the human population today. For example hyperthyroidism(over) and hypothyroidism(under), which are relate to release of amount of thyroid hormones the thyroid gland produces and whether it is over active trusted source (when thyroid gland makes too much thyroid hormone) or under active trusted source (when the thyroid gland doesn’t make enough thyroid hormone). We need to identify whether the patient has thyroid or not.

**2.3 Proposed solution**

We need to build a ML model which will be used by hospitals and help the hospital authority to identify if the patient has thyroid or not. If it is a positive case then medical will do further test to know what type of thyroid the person is suffering from and according to that the treatment will be on fast-track. The doctors will start treating the patients. If the result will come negative then the patient will be sent to a junior doctor and the junior doctors by using their own expertise they will decide that if the model has done correct prediction or not. If analysis comes true then the doctor release the patient. By seeing the readings if doctor analysed that there may be a chance of thyroid then patient sent to the senior doctors.

**2.4 Further improvements**

1. This ML model can be used to identify the type of thyroid based on test input given to the model.

2. Hashimoto **disease** can be identified from the initial stages, whichis the most common cause of hypothyroidism.

**2.5 Technical Requirements**

Medical testing equipment to perform tests and take readings.

**2.6 Data Requirements**

Data requirement completely depend on our problem statement.

* Patient statement need to recorded for age, sex etc.
* Readings from the different tests.

We require at least 50-60 readings from each label, where readings are nothing, but numerical values of tests performed in medical lab and patient statement.

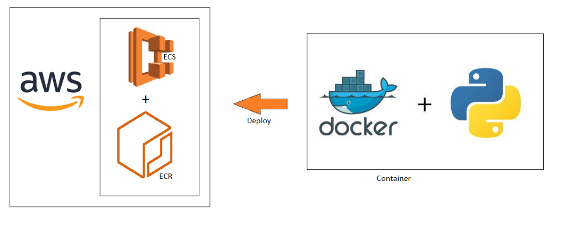
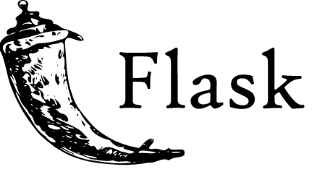
**2.7 Tools used**

Python programming language and it’s libraries & modules such as Numpy, Pandas, SciPy, Scikit-Learn, AWS, GitHub, mongo DB, docker, airflow.

A picture containing text, clipart

Description automatically generatedDiagram

Description automatically generated



* vsCode is used as IDE
* For visualization of the plots, Matplotlib, Seaborn, and Plotly are used.
* AWS is used for deployment of the model.
* Docker is used to create image by removing all the dependencies and deploy our code to virtual machines.
* MongoDB is used to retrieve, insert, delete, update database.
* Git is used as version control system.
* Airflow is used to schedule our job in CI/CD pipeline.
* Flask is used to create API for UI

**2.7.1 Hardware Requirements**

* Medical laboratories

**2.7.2 Thyroid Disease and Operational Data for Detection**

This program Thyroid Disease Detection data is From Garavan Institute

Documentation: as given by Ross Quinlan. It is an open-source software to which you can redistribute and/or modify.

**2.8 Constraints**

Each patient with positive outcomes need to be treated in fast track by senior doctors. And the patients with negative outcomes need to be verified by junior doctors.

**2.9 Assumptions**

The main objective of the project is to implement the use cases as previously mentioned (2.2 Problem statement) for new dataset that comes through medical lab test readings. Machine Learning based disease detection model is used for predicting the patient has thyroid or not. It is also assumed that all aspects of this project have the ability to work together in the way the designer is expecting.

**3. Design Details**

**3.1 Process flow**

To predict the patient has thyroid or not, we will use machine learning base model. Below is the process flow diagram is as shown below.

**Proposed methodology**

ML model on disease detection

Training/Validation on Dataset

Take the test readings and patient statement.

Prediction of problems (usecases) in the component

To take necessary action for treatment

**3.1.1 Model Training and Evaluation**

(3) Data

augmentation

(7)

Prediction

(5)

Training

New readings

Training set

(4)

Split

(1)

Result of the evaluation

Model

Annotated dataset

Dataset

(2)

annotation

(6)

Evaluation

Test set

Prediction

**3.1.2 Deployment Process**

**To run application :**

1. **Create python environment.**
2. **Install Dependencies**
3. **Run script.**

Virtual Machine

**Docker**

**Local**

**Docker**: Independent from local system

Linux Docker

1. Choose base machine.
2. Copy code from local to docker machine.
3. Install dependencies.
4. Run script.

Source Code

1. 2. 3.

Docker Image

Local: Test our code

Run your application.

GitHub Actions

Pull

Docker Image

Build Docker image and store to ECR

Continuous Delivery

**EC2**

Ubuntu Machine

Continuous Deployment

Run Docker Image

Access your Airflow.

Load Model

Start

Get readings from UI

Model

Pre-process readings

Make

predictions

Predicted decoded result.

* 1. **Event log**

The system should log every event so that the user will know what process is running internally.

Initial Step-By-Step Description:

1. The System identifies at what step logging required
2. The System should be able to log each and every system flow.
3. System should not hang even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do
   1. **Exception Handling**

Errors should be encountered, an explanation will be displayed as to what went wrong?, in which module error occurred and what is the line number. An error will be defined as anything that falls outside of the normal and intended usage.

**4. Performance**

Thyroid is most common disease with endocrine disorder. The given features along with test readings will relatively give us better result for disease prediction.

From the above problem statement we could observe that, we have to predict if the patient has thyroid or not using readings and patient statement. So, prediction should be accurate and model retraining is also very important.

**4.1 Reusability**

The code written and the components used should have ability to be reused with no problems.

**4.2 Application Compatibility**

The different components for this project will be using Python as an interface between them. Each component will have it’s own task to perform, and it is the job of the python to ensure proper transfer of information.

**4.3 Resource Utilization**

When any task is performed, it will likely use all the processing power available until that function is finished.

**4.4 Deployment**



**5. Conclusion**

The ML model will predict the patient has thyroid or not based on various pre-existing data, so we can help the medical to reduce human efforts.