Low Level Design (LLD)

Low Level Design (LLD)

[**Insurance Price\_Prediction**](https://github.com/mohiteyashprogrammer/Adult_income_Prediction/tree/main)

Revision Number – 1.0

Last Date of Revision – 29-06-2023

**ILMA ARSHAD**

1

Low Level Design (LLD)

2

Low Level Design (LLD)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Contents** | | | | | | | | | | | | | | | |  |  |  |  |
| [**Abstract**](https://docs.google.com/document/d/1ZLlZEVdMuxYl3o0PziQjZcUbNmAcz-PR/edit#heading=h.gjdgxs) | |  | | | | | | | | | | | | | | [4](https://docs.google.com/document/d/1ZLlZEVdMuxYl3o0PziQjZcUbNmAcz-PR/edit#heading=h.gjdgxs) | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| [**INTRODUCTION**](https://docs.google.com/document/d/1ZLlZEVdMuxYl3o0PziQjZcUbNmAcz-PR/edit#heading=h.30j0zll) | | | | |  | | | | | | | | | | | [5](https://docs.google.com/document/d/1ZLlZEVdMuxYl3o0PziQjZcUbNmAcz-PR/edit#heading=h.30j0zll) | | | |
|  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | [**Why this LLD documentation?**](https://docs.google.com/document/d/1ZLlZEVdMuxYl3o0PziQjZcUbNmAcz-PR/edit#heading=h.1fob9te) | | | | | | | | | | | | | | | [5](https://docs.google.com/document/d/1ZLlZEVdMuxYl3o0PziQjZcUbNmAcz-PR/edit#heading=h.1fob9te) | | |  |
|  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| [**1 Architecture**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.3znysh7) | | | | | | | | | | | | | | | | [**3**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.3znysh7) | | | |
|  |  | |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |
| [**2 Architecture Design**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.3dy6vkm) | | | | | | | |  | | | | | | | | [**6**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.3dy6vkm) | | | |
|  |  | | |  |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |
|  | [**2.1 Data Gathering from Main Source**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.1t3h5sf) | | | | | | | | | | | | | | | [**6**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.1t3h5sf) | | |  |
|  |  | | |  |  |  |  | |  |  |  |  | |  |  |  |  |  |  |
|  | [**2.2 Tools Used**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.17dp8vu) | | | |  | | | | | | | | | | |  | | [**6**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.17dp8vu) | |
|  |  | | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |
|  | [**2.3 Data Description**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.3rdcrjn) | | | | | | | | | | | | | | | [**7**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.3rdcrjn) | | |  |
|  |  | | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |
|  | [**2.4 Import Data into Database**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.26in1rg) | | | | | | | | | | | | |  | | [**7**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.26in1rg) | | |  |
|  |  | | |  |  |  |  | | |  |  |  | |  |  |  |  | |  |
|  | [**2.5 Export Data from Database**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.lnxbz9) | | | | | | | | | | | | | |  | [**8**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.lnxbz9) | | |  |
|  |  | | |  |  |  |  | | |  |  |  | | |  |  |  | |  |
|  | [**2.6 Data Pre-Processing**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.35nkun2) | | | | | | | | | |  | | | |  | [**8**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.35nkun2) | | |  |
|  |  | | |  |  |  |  | | |  |  |  | | | |  |  | |  |
|  | [**2.7 Modelling**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.1ksv4uv) | | | | | | | | | |  | | | | | [**8**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.1ksv4uv) | | | |
|  |  | | |  |  |  |  | | |  | |  | | | |  |  | |  |
|  | [**2.8 UI Integration**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.44sinio) | | | | |  | | | | | | | | | | [**8**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.44sinio) | | | |
|  |  | | | |  |  |  | | |  | |  | | | |  |  | |  |
|  | [**2.9 Data From User**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.2jxsxqh) | | | | | |  | | | | | | | | | [**8**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.2jxsxqh) | | | |
|  |  | | | |  | |  | | |  | |  | | | |  |  | |  |
|  | [**2.10 Data Validation**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.z337ya) | | | | | | | | |  | | | | | | [**8**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.z337ya) | | |  |
|  |  | | | |  | |  | | |  | |  | | | |  |  | |  |
|  | [**2.11 Rendering the Results**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.3j2qqm3) | | | | | | | | | | |  | | | | [**8**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.3j2qqm3) | | |  |
|  |  | | | |  | |  | | | | |  | | | |  |  | |  |
|  | [**3 Deployment**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.1y810tw) | | | |  | | | | | | |  | | | |  | [**8**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.1y810tw) | |  |
|  | [**3.1 Unit Test Cases**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.4i7ojhp) | | | | | |  | | | | | | | | | [**9**](https://docs.google.com/document/d/1CawpaB-mO_T9ew2gcRJPxTYNEbUemWHc/edit#heading=h.4i7ojhp) | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

3

Low Level Design (LLD)

**Abstract**

This abstract summarizes a project aimed at providing individuals with personalized estimates of their insurance coverage needs based on their individual health situations. The project's objective is to assist individuals in making informed decisions by accurately predicting their future medical expenses. The prediction is based on factors such as age, sex, BMI, number of children, smoking status, and region of residence. By leveraging historical data and advanced data analytics techniques, the project aims to develop a reliable model for estimating future healthcare costs.

The project intends to empower individuals to work with any health insurance carrier and their plans, while keeping the projected cost from the study in mind. This approach allows individuals to focus on the health-related aspects of an insurance policy, rather than being burdened by ineffective or inadequate coverage.

The project entails data collection, preprocessing, feature engineering, and model development stages. Data collection involves acquiring and integrating relevant health and demographic data from various sources. Preprocessing techniques, such as data cleaning and normalization, ensure data quality and consistency. Feature engineering focuses on deriving meaningful variables from the available data to enhance prediction accuracy.

The model development stage utilizes machine learning algorithms to predict future medical expenses. Regression models, ensemble methods, and deep learning models are considered for this purpose. Model selection, evaluation, and hyperparameter tuning are crucial steps to optimize performance and ensure accurate predictions.

The project's anticipated outcome is a user-friendly system that provides individuals with personalized estimates of their insurance coverage needs. By incorporating accurate predictions of future medical expenses, the system aims to guide individuals towards suitable health insurance plans. Ultimately, the project aims to assist individuals in focusing on the health-related aspects of insurance policies, leading to more effective coverage and improved decision-making processes.

Top of Form

Bottom of Form

4

Low Level Design (LLD)

**1 Introduction**

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

The main purpose of this LLD documentation is to feature the required details of the project and supply the outline of the machine learning model and also the written code. This additionally provides the careful description on however the complete project has been designed end-to-end.

**1.2 Architecture**



**2. Architecture Design**

The proposed architecture follows a microservices-based approach with containerization for scalability and flexibility. It includes a front-end layer for user interaction and a back-end layer for data processing and prediction. The back-end consists of multiple microservices, such as data ingestion, preprocessing, feature engineering, and prediction. A message queue system ensures asynchronous communication between microservices for improved performance. The architecture utilizes a distributed database for efficient data storage and retrieval. Container orchestration using Kubernetes manages scalability and fault tolerance. RESTful APIs facilitate communication between the front-end and back-end components. Data security measures, such as encryption and access controls, are implemented at each layer. Continuous integration and deployment pipelines ensure seamless updates and releases. Monitoring and logging systems are integrated to track system performance and detect issues in real-time.

5

Low Level Design (LLD)

**2.1. Data Gathering**

The data for the current project is being gathered from Kaggle dataset, the link to the data is: [LINK](https://www.kaggle.com/datasets/noordeen/insurance-premium-prediction)

**2.2. Tool Used**

• Python 3.7 is employed because the programming language and frame works like numpy, pandas, sklearn and alternative modules for building the model.

* VScode is employed as IDE.
* For visualizations seaborn and components of matplotlib are getting used
* For information assortment prophetess info is getting used version

management.

• Netlify is employed for deployment

**2.3 Data Description**

The insurance premium prediction dataset consists of historical records of individuals' health and demographic information. The dataset includes attributes such as age, sex, BMI (Body Mass Index), number of children, smoking status, and region of residence. Each record corresponds to an individual and their respective insurance premium. The dataset encompasses a diverse range of individuals with varying characteristics, providing a comprehensive representation of the target population.

The dataset contains information from a wide geographic area, capturing regional differences in insurance pricing. It includes both categorical and numerical features, allowing for a holistic understanding of the factors influencing insurance premiums. The dataset is representative of a specific time period, ensuring temporal relevance in predicting future insurance premiums.

Quality checks and preprocessing have been performed to ensure data integrity, including handling missing values and outliers. The dataset is anonymized and follows privacy regulations to protect individuals' personal information. It is a well-curated and reliable dataset, suitable for training and evaluating machine learning models for insurance premium prediction.

The dataset serves as a valuable resource for developing a robust insurance premium prediction model, enabling individuals to estimate their insurance costs based on their health and demographic attributes.

Top of Form

Bottom of Form

6

Low Level Design (LLD)

**2.4 Import Data into Database**

* Created associate api for the transfer of the info into the Cassandra info, steps performed are:
* Connection is created with the info.
* Created a info with name Insurance.
* Cqlsh command is written for making the info table with needed parameters.
* And finally, a cqlsh command is written for uploading the knowledgeset into data table by bulk insertion.

**2.5 Export Data into Database**

In the above created api, the download url is also being created, which downloads the data into a csv file format.

**2.6 Data Preprocessing**

Steps performed in pre-processing are:

* First the info sorts square measure being checked and located solely the value column is of sort number.
* Checked for null values as there square measure few null values, those rows square measure born.
* Converted all the desired column into the date time format.
* Performed one-hot cryptography for the desired columns.
* Scaling is performed for needed information.
* And, the info is prepared for passing to the machine learning formula

7

Low Level Design (LLD)

**2.7 Modelling**

The pre-processed information is then envisioned and everywhere the specified insights are being drawn. Though from the drawn insights, the info is at random unfold however still modelling is performed with completely different machine learning algorithms to form positive we tend to cowl all the chances and eventually, for sure random forest regression performed well and any hyperparameter calibration is finished to extend the model’s accuracy.

**2.8 UI Integration**

Both CSS and HTML files are being created and are being integrated with the created machine learning model. All the required files are then integrated to the app.py file and tested locally. Note CSS and HTML is not done by me.

**2.3 Data from User**

The data from the user is retrieved from the created HTML web page.

**2.4 Data Validation**

The data provided by the user is then being processed by app.py file and validated. The validated data is then sent for the prediction.

**2.11 Rendering Result**

The data sent for the prediction is then rendered to the web page.

**3. Deployment**

The tested model is then deployed to AWS. So, users can access the project from any internet devices. Here is the link :

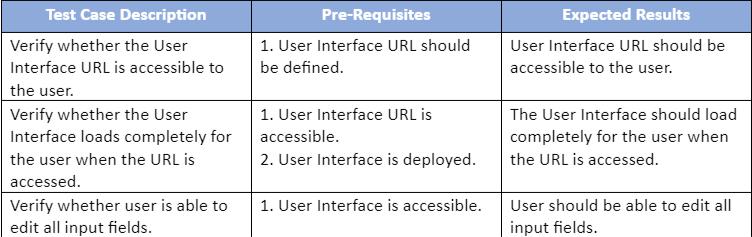
<http://insurancedeploy-env.eba-zvcbn3vk.us-east-1.elasticbeanstalk.com/>



8

Low Level Design (LLD)

**3.1 Unit Test**



9