

# **Architecture Design**

## **Credit Card Default Prediction**

Revision Number – 1.1

Last Date of Revision: 22-06-2023

Nanthini. S

Nithish. R

## Document Version Control

Date	Version	Description	Author
16-05-23	1.0	Abstract Introduction Architecture	Nanthini. S Nithish.R
16-05-23	1.1	Architecture Design	Nanthini. S Nithish. R

## Contents

<b>Document Version Control .....</b>	<b>2</b>
<b>Abstract .....</b>	<b>4</b>
<b>1. Introduction .....</b>	<b>5</b>
<b>1.1 Why this Architecture Design Document ? .....</b>	<b>5</b>
<b>2. Architecture .....</b>	<b>5</b>
<b>3. Architecture Design .....</b>	<b>6</b>
3.1 Data Collection .....	6
3.2 Data Description .....	6
3.3 Importing data into database .....	7
3.4 Exporting data from database .....	7
3.5 Data preprocessing .....	7
3.6 Modelling process .....	7
3.7 UI Integration .....	7
3.8 Data from User .....	7
3.9 Data Validation .....	8
3.10 Rendering the Results .....	8
3.11 Depolyment .....	8

## **Abstract**

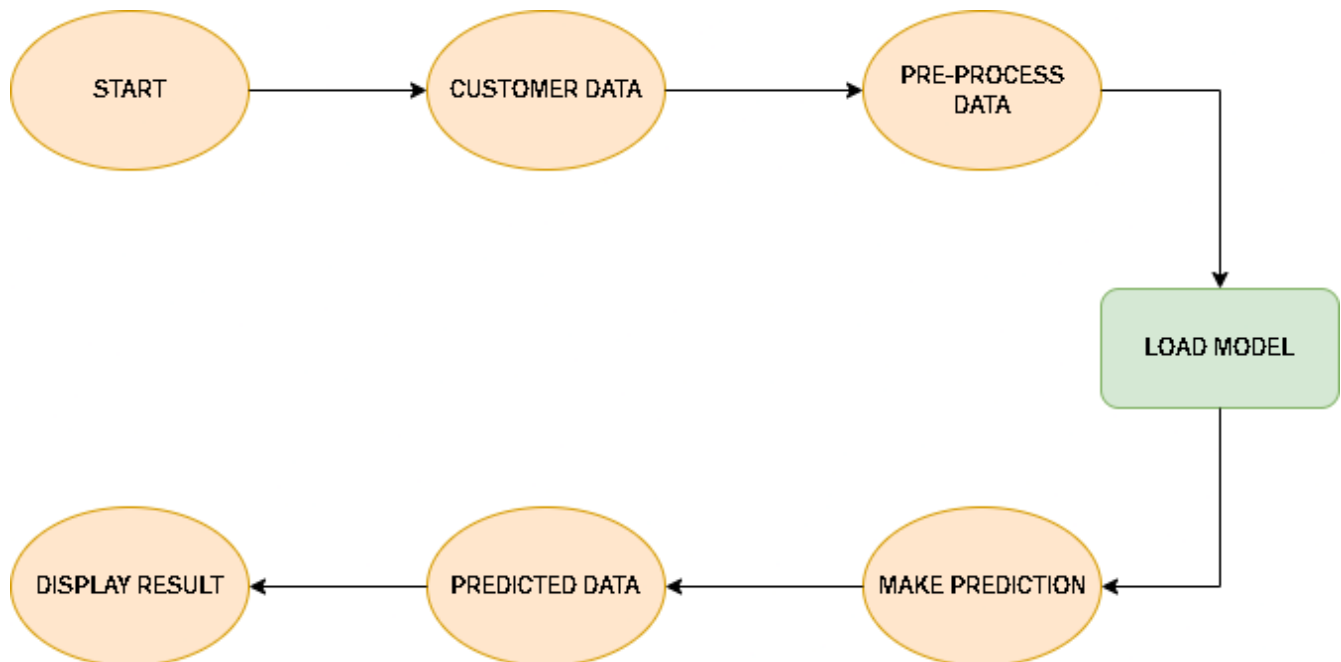
Recent trends are to build tall buildings in big cities as a way out of the current housing overpopulation problem. These new structures unveil problems that if not addressed in time could cause catastrophes of unimaginable impact. Some of those problems is the incidence of a fire threat happening upstairs in one of those buildings, medical emergencies due to any road accidents or mob that may cause threat to the human kind. This work discusses the implementation of the unmanned ground vehicles to spot the real location of the medical emergencies due to road mishap, mob or illegal activities such as hooliganism, snatching, robbery and the fire emergency and accordingly channelize or route them to the concerned helpline for quick mitigation and avoid disaster.

## 1. Introduction

### 1.1 Why this Architecture Design Document ?

The main objective of the Architecture design documentation is to provide the internal logic understanding of the Credit card defaults code. The Architecture design documentation is designed in such a way that the programmer can directly code after reading each module description in the documentation.

## 2. Architecture



### 3. Architecture Design

#### 3.1 Data collection

Data Export from Database - The data in a stored database is exported as a CSV file to be used for Data Pre-processing and Model Training.

#### 3.2 Data discription

Credit card default is dataset publicly available on the Kaggle. The information in the dataset is present in two separated excel files named as train.xlsx and test.xlsx. Dataset contains 10683 rows which shows the information such Date of Journey, Source, Destination, Arrival Time, Departure Time, Total stops, Airlines, Additional Info and Price. The glance of the Dataset is :

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	
1		ID	AGE	BILL_AMT1	BILL_AMT2	BILL_AMT3	BILL_AMT4	BILL_AMT5	BILL_AMT6	EDUCATIO	LIMIT_BAL	MARRIAGE	PAY_0	PAY_2	PAY_3	PAY_4	PAY_5	PAY_6	PAY_AMT1	PAY_AMT2	PAY_AMT3	PAY_AMT4	PAY_AMT5	PAY_AMT6	SEX	default_payment_next_month			
2	0	4317	24	4050	5426	6743	8544	9858	1303	3	10000	2	0	0	0	0	0	0	2000	2000	2000	2000	1000	0	1	0			
3	0	25269	25	52109	43007	33481	28563	24065	2802	2	120000	1	0	0	0	0	0	0	1565	3194	1200	6206	0	310	2	0			
4	0	3372	36	45849	46909	47928	48731	49793	50812	2	140000	1	0	0	0	0	0	0	2100	2100	1900	2000	2000	2100	2	0			
5	0	14340	36	0	250	79	79	290	106	1	390000	1	1	-1	-1	-1	-1	-1	250	79	0	290	106	66	1	1			
6	0	18417	24	30767	28818	30411	28805	26895	27162	2	30000	2	0	0	0	0	0	0	1600	2517	4006	2000	1000	1500	2	0			
7	0	1584	70	49546	48480	48621	17793	18224	18612	1	50000	2	2	2	0	0	0	0	0	2200	700	700	674	608	2	0			
8	0	7034	24	28329	28980	29564	29710	29700	0	2	90000	1	0	0	0	0	0	0	1800	1774	920	1102	4014	0	2	0			
9	0	24299	37	9608	12127	11647	13138	14903	17664	2	40000	1	0	0	2	0	0	0	3000	0	2000	2000	3000	2000	1	0			
10	0	13909	28	158245	161695	163620	165604	169075	172401	1	180000	2	2	2	2	2	2	2	7400	6000	6100	6200	6200	6300	2	0			
11	0	28386	35	0	0	0	0	249	317	3	100000	1	1	-2	-2	-2	-2	-1	-1	0	0	0	249	317	0	2	0		
12	0	9892	32	50669	48942	27836	40346	38668	20016	2	50000	1	0	0	0	0	0	0	1613	1484	3675	698	1000	706	1	0			
13	0	16096	36	59379	63007	61459	59798	61287	8383	2	140000	1	1	2	3	2	0	0	5200	0	0	3009	1000	94000	2	0			
14	0	26713	26	14598	13998	15467	14913	17692	16302	3	30000	2	2	2	2	2	2	2	2000	2000	0	3009	0	2000	1	0			
15	0	19221	47	25893	30628	29368	27769	25999	26924	2	30000	1	2	2	2	0	0	0	5501	22	2000	1000	2000	2039	2	1			
16	0	28459	42	0	1473	1863	0	112978	115357	3	160000	1	-2	-2	-2	-2	-2	-1	0	1473	1863	0	119001	5000	4500	2	0		
17	0	9640	29	128208	100939	92549	90038	88772	111287	1	260000	2	0	0	0	0	0	0	17000	6000	5000	6500	25000	4000	1	0			
18	0	23912	23	31348	50778	53407	19443	16597	18423	1	60000	2	0	0	0	0	0	0	20109	5000	2000	1000	2000	1000	1	0			
19	0	9067	27	3682	4714	2336	0	5471	5415	3	80000	2	2	2	2	2	2	-1	-1	3193	0	0	5471	5415	5124	2	0		
20	0	24066	23	14697	11983	4098	14899	10170	7151	1	20000	2	-1	-1	-1	-1	-1	-1	12483	4098	14899	10170	427	0	1	0			
21	0	18410	25	18696	20086	20285	16530	0	0	2	20000	2	2	0	0	0	0	0	-2	1705	1418	2011	0	0	0	2	1		
22	0	17062	23	50653	49348	47995	40226	27828	28411	3	50000	2	2	0	0	0	0	0	0	2190	2027	2204	996	1031	1047	2	0		
23	0	14059	46	48885	48793	34440	30889	30182	29912	2	50000	1	0	0	0	0	0	0	0	2000	2500	2000	1500	5000	1500	1	0		
24	0	4830	48	7065	8148	9199	9558	9136	9392	3	10000	3	0	0	0	0	0	0	0	1200	1200	500	500	500	0	2	0		
25	0	2731	30	31217	33423	32600	31777	33966	34759	3	50000	2	1	2	3	2	2	2	2	3000	0	0	2700	1500	0	1	0		
26	0	27002	41	8988	6849	7471	8043	19526	7564	2	50000	1	0	0	0	0	0	0	0	1200	2000	2000	3000	1000	10640	1	0		
27	0	28069	40	92393	85345	76109	77068	77868	75954	2	100000	2	0	0	0	0	0	0	0	3500	3500	3300	3000	3000	3000	2	0		
28	0	10113	31	9514	2067	-6	-409	793	2385	2	50000	2	0	-1	-1	-2	-1	-1	-1	2100	405	0	2385	2385	3000	2	0		
29	0	5056	25	232	815	631	1237	1529	357	1	80000	2	-1	-1	-1	-1	-1	-1	-1	815	631	1237	1529	357	2957	2	0		
30	0	11194	25	140876	48005	13650	12371	12511	10697	2	140000	1	0	0	0	0	0	0	0	3600	4015	600	500	800	119	2	0		
31	0	22591	41	8554	1375	2310	0	0	351	2	160000	0	0	0	0	0	0	-2	-1	1000	1000	0	0	351	0	2	1		
32	0	19124	32	1249	142	5965	991	-9	1974	1	450000	2	-2	-2	-2	-2	-2	-2	-2	142	6001	996	0	1983	7228	2	0		
33	0	28572	29	96686	77817	29166	7403	3095	9641	2	120000	2	0	0	0	0	-1	-1	-1	3508	2003	2772	3095	9641	2803	2	0		
34	0	6428	23	8002	7954	9777	10026	9657	9279	2	100000	2	0	0	0	0	0	0	0	2000	2000	1000	504	1000	2000	2	0		
35	0	11929	39	44574	0	0	0	0	0	3	50000	1	0	0	-2	-2	-2	-2	-2	0	0	0	0	0	0	1	0		
36	0	18464	25	69816	71257	72812	74305	75872	77554	2	80000	2	0	0	0	0	0	0	0	2600	2700	2700	2800	2900	4300	2	0		
37	0	22607	38	25080	26460	27042	27617	28546	13487	1	30000	1	2	0	0	0	0	0	0	1800	2000	1400	1300	700	1000	2	0		
38	0	17728	51	26106	8999	0	0	10961	11349	1	440000	2	0	-1	-1	-2	-1	0	8999	0	0	10961	503	448	2	0			
39	0	2713	39	56045	59524	58563	56831	53628	34490	1	60000	1	0	0	2	0	0	0	0	5000	2000	1000	3000	8000	0	2	0		

### 3.3 Importing 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 cardInfo.
- cqlsh command is written for making the info table with needed parameters.
- And finally, a cqlsh command is written for uploading the Knowledge Set into data table by bulk insertion.

### 3.4 Exporting data from database

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

### 3.5 Data preprocessing

- Checked for info of the Dataset, to verify the correct datatype of the Columns.
- Checked for Null values, because the null values can affect the accuracy of the model.
- Converted all the desired columns into Datetime format.
- Performed One – Hot encoding on the desired columns.
- Checking the distribution of the columns to interpret its importance.
- Now, the info is prepared to train a Machine Learning Model.

### 3.6 Modeling process

After preprocessing the data, We visualize our data to gain insights and then these insights are randomly spread and split into two parts, train and test data. After splitting the data, we use Random Forest Regressor to model our data to predict the Credit card default prediction.

### 3.7 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.

### 3.8 Data from User

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

### **3.9 Data Validation**

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

### **3.10 Rendering the Results**

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

### **3.11 Deployment**

The tested model is then deployed to local Flask app

.