Architecture

Credit Card Default Detection Application

by

Abhijit Paul

Gouthami K

Revision Number: 1.0
Last date of revision: 17/12/2023

Document Version Control

Date Issued	Version	Description	Author
17/12/2023	1	Initial AD - V1.0	Gouthami K

Contents

Abstract

The architecture of the Credit Card Default Detection machine learning project is outlined in this document, serving as a comprehensive guide to the underlying structure and implementation intricacies of the system. This architectural overview delves into key design decisions and specifications crucial for the successful development and deployment of the Credit Card Default Detection application.

The document provides insights into the system architecture, elucidating the relationships and interactions among various components. It details the selected classification algorithm, its architectural nuances, and the essential input features. Furthermore, the document offers a comprehensive understanding of model training methodologies and validation techniques employed to ensure the predictive accuracy of the system.

API design considerations form a crucial aspect of this architectural documentation, encompassing endpoint definitions, input validation procedures, error handling mechanisms, and robust security measures, including token authentication. Deployment strategies, along with environment specifications and containerization details, are meticulously outlined to facilitate a seamless and efficient implementation process.

The document also addresses testing methodologies, both at the unit and integration levels, ensuring the reliability of both the machine learning model and the API functionalities. Logging mechanisms and monitoring metrics are defined to enable effective debugging and continuous assessment of the system's health.

A strong emphasis is placed on documentation planning, covering code documentation with a focus on docstrings and comments. Additionally, a comprehensive user guide is provided to assist both developers and end-users in navigating the system, fostering a smooth and user-friendly interaction.

Security measures are integrated to address data privacy concerns, outlining strategies to safeguard sensitive financial data. Model explainability techniques are incorporated to enhance transparency in predictions, providing stakeholders with a clear understanding of the Credit Card Default Detection system.

This abstract serves as a succinct reference, encapsulating the essential features of the architectural design document for stakeholders involved in the evolution of the Credit Card Default Detection system.

1 Introduction

1.1 Purpose and Significance of Architecture Document?

This architectural document aims to provide a comprehensive overview of the design and structure underlying the Credit Card Default Detection system. It outlines the purpose and significance of the architecture, emphasizing its role in guiding the development team and stakeholders through the intricate details of the system's construction. By detailing key architectural decisions, this document ensures a shared understanding of the system's blueprint, fostering efficient collaboration and informed decision-making.

1.2 Scope

The scope of this architecture document extends to elucidating the fundamental design elements, relationships between components, and the overall structure of the Credit Card Default Detection system. It is intended for a diverse audience, including stakeholders and developers, providing them with a detailed insight into the architectural decisions that define the system. This document, proposed for approval by higher management, serves as a guiding resource throughout the development lifecycle, facilitating a cohesive and well-informed approach to system architecture.

2 Technical specifications

2.1 Dataset

The dataset includes information on each client's personal details, credit limit, demographic data, repayment history, bill statements, previous payments, and whether they defaulted on the next month's payment. This comprehensive dataset can be utilized for developing predictive models to assess credit risk and predict the likelihood of default for each client. The information is spread across multiple tables, mirroring the structure of a relational database, providing a rich source for analyzing and modeling credit card default detection.
There are a total of 2,2500 records in the training set and 7,500 records in the testing set, and 3,5018 records in the training set and 11,710 records in the testing set after resampling

	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
1	ID	LIMIT_BAL	SEX	EDUCATIC	MARRIAG	AGE	PAY_0	PAY_2	PAY_3	PAY_4	PAY_5	PAY_6	BILL_AMT	BILL_AMT	BILL_AMT	BILL_AMT	BILL_AMT	BILL_AMT	PAY_AMT	PAY_AMT	PAY_AMT	PAY_AMT	PAY_AMT	PAY_AMT	default.payment.next.month		
2	1	20000	2	2	1	24	2	2	-1	-1	-2	-2	3913	3102	689	0	0	0	0	689	0	0	0	0	1		
3	2	120000	2	2	2	26	-1	2	0	0	0	2	2682	1725	2682	3272	3455	3261	0	1000	1000	1000	0	2000	1		
4	3	90000	2	2	2	34	0	0	0	0	0	0	29239	14027	13559	14331	14948	15549	1518	1500	1000	1000	1000	5000	0		
5	4	50000	2	2	1	37	0	0	0	0	0	0	46990	48233	49291	28314	28959	29547	2000	2019	1200	1100	1069	1000	0		
6	5	50000	1	2	1	57	-1	0	-1	0	0	0	8617	5670	35835	20940	19146	19131	2000	36681	10000	9000	689	679	0		
7	6	50000	1	1	2	37	0	0	0	0	0	0	64400	57069	57608	19394	19619	20024	2500	1815	657	1000	1000	800	0		
8	7	5.00E+05	1	1	2	29	0	0	0	0	0	0	367965	412023	445007	542653	483003	473944	55000	40000	38000	20239	13750	13770	0		
9	8	1.00E+05	2	2	2	23	0	-1	-1	0	0	-1	11876	380	601	221	-159	567	380	601	0	581	1687	1542	0		
10	9	140000	2	3	1	28	0	0	2	0	0	0	11285	14096	12108	12211	11793	3719	3329	0	432	1000	1000	1000	0		
11	10	20000	1	3	2	35	-2	-2	-2	-2	-1	-1	0	0	0	0	13007	13912	0	0	0	13007	1122	0	0		
12	11	2.00E+05	2	3	2	34	0	0	2	0	0	-1	11073	9787	5535	2513	1828	3731	2306	12	50	300	3738	66	0		
13	12	260000	2	1	2	51	-1	-1	-1	-1	-1	2	12261	21670	9966	8517	22287	13668	21818	9966	8583	22301	0	3640	0		
14	13	630000	2	2	2	41	-1	0	-1	-1	-1	-1	12137	6500	6500	6500	6500	2870	1000	6500	6500	6500	2870	0	0		
15	14	70000	1	2	2	30	1	2	2	0	0	2	65802	67369	65701	66782	36137	36894	3200	0	3000	3000	1500	0	1		
16	15	250000	1	1	2	29	0	0	0	0	0	0	70887	67060	63561	59696	56875	55512	3000	3000	3000	3000	3000	3000	0		
17	16	50000	2	3	3	23	1	2	0	0	0	0	50614	29173	28116	28771	29531	30211	0	1500	1100	1200	1300	1100	0		
18	17	20000	1	1	2	24	0	0	2	2	2	2	15376	18010	17428	18338	17905	19104	3200	0	1500	0	1650	0	1		
19	18	320000	1	1	1	49	0	0	0	-1	-1	-1	253286	246536	194663	70074	5856	195599	10358	10000	75940	20000	195599	50000	0		
20	19	360000	2	1	1	49	1	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0		
21	20	180000	2	1	2	29	1	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0		
22	21	130000	2	3	2	39	0	0	0	0	0	-1	38358	27688	24489	20616	11802	930	3000	1537	1000	2000	930	33764	0		
23	22	120000	2	2	1	39	-1	-1	-1	-1	-1	-1	316	316	316	0	632	316	316	316	0	632	316	0	1		
24	23	70000	2	2	2	26	2	0	0	2	2	2	41087	42445	45020	44006	46905	46012	2007	3582	0	3601	0	1820	1		
25	24	450000	2	1	1	40	-2	-2	-2	-2	-2	-2	5512	19420	1473	560	0	0	19428	1473	560	0	0	1128	1		
26	25	90000	1	1	2	23	0	0	0	-1	0	0	4744	7070	0	5398	6360	8292	5757	0	5398	1200	2045	2000	0		
27	26	50000	1	3	2	23	0	0	0	0	0	0	47620	41810	36023	28967	29829	30046	1973	1426	1001	1432	1062	997	0		
28	27	60000	1	1	2	27	1	-2	-1	-1	-1	-1	-109	-425	259	-57	127	-189	0	1000	0	500	0	1000	1		
29	28	50000	2	3	2	30	0	0	0	0	0	0	22541	16138	17163	17878	18931	19617	1300	1300	1000	1500	1000	1012	0		
30	29	50000	2	3	1	47	-1	-1	-1	-1	-1	-1	650	3415	3416	2040	30430	257	3415	3421	2044	30430	257	0	0		
31	30	50000	1	1	2	26	0	0	0	0	0	0	15329	16575	17496	17907	18375	11400	1500	1500	1000	1000	1600	0	0		
32	31	230000	2	1	2	27	-1	-1	-1	-1	-1	-1	16646	17265	13266	15339	14307	36923	17270	13281	15339	14307	37292	0	0		
33	32	50000	1	2	2	33	2	0	0	0	0	0	30518	29618	22102	22734	23217	23680	1718	1500	1000	1000	1000	716	1		
34	33	1.00E+05	1	1	2	32	0	0	0	0	0	0	93036	84071	82880	80958	78703	75589	3023	3511	3302	3204	3200	2504	0		

2.2 Predicting Credit Card Default

- The system presents a form with fields representing the client's information, similar to historical data attributes (e.g., credit limit, gender, education, marital status, age, repayment history, bill amounts, payment amounts).
- The user fills in the required information for the client in question.
- Upon confirmation, the system processes the input data through the selected prediction model.
- The prediction output is displayed, indicating whether the model predicts a default or non-default for the given client.

2.3 Logging

1. User Activity Logging:
Ensure comprehensive logging of all user activities within the system.

2. Dynamic Logging Identification:
The system intelligently determines the appropriate steps in the workflow where logging is necessary.

3. System Flow Logging:
Enable logging for every step and aspect of the system's flow to capture detailed information.

4. Flexible Logging Methods:
Developers have the option to employ various logging methods, including database logging or file logging.

5. Optimized System Performance:The system should maintain optimal performance even when employing extensive logging methods, such as database or file logging.

6. Logging for Debugging:
Emphasize the importance of logging for debugging purposes; it is a mandatory practice to facilitate issue identification and resolution.

2.4 Database

1. Data Storage:
The system must store every credit card default detection request in the database, ensuring a format that facilitates easy model retraining.

2. User Interaction:
The user selects the credit card default detection option and provides the necessary information.

3. Comprehensive Data Storage:
The system systematically stores all user-provided data or received information related to credit card default detection in the chosen database, allowing flexibility between MongoDB and MySQL.

2.5 Deployment

Deployment, in the context of software development, refers to the process of making a software application available and operational in a specific environment. It involves transferring a developed and tested software system from a staging or development environment to a production environment, where it becomes accessible to end-users. In the case of this particular project, AWS (Amazon Web Services) was utilized for the deployment, leveraging its cloud infrastructure and services to host, scale, and manage the application in the production environment.

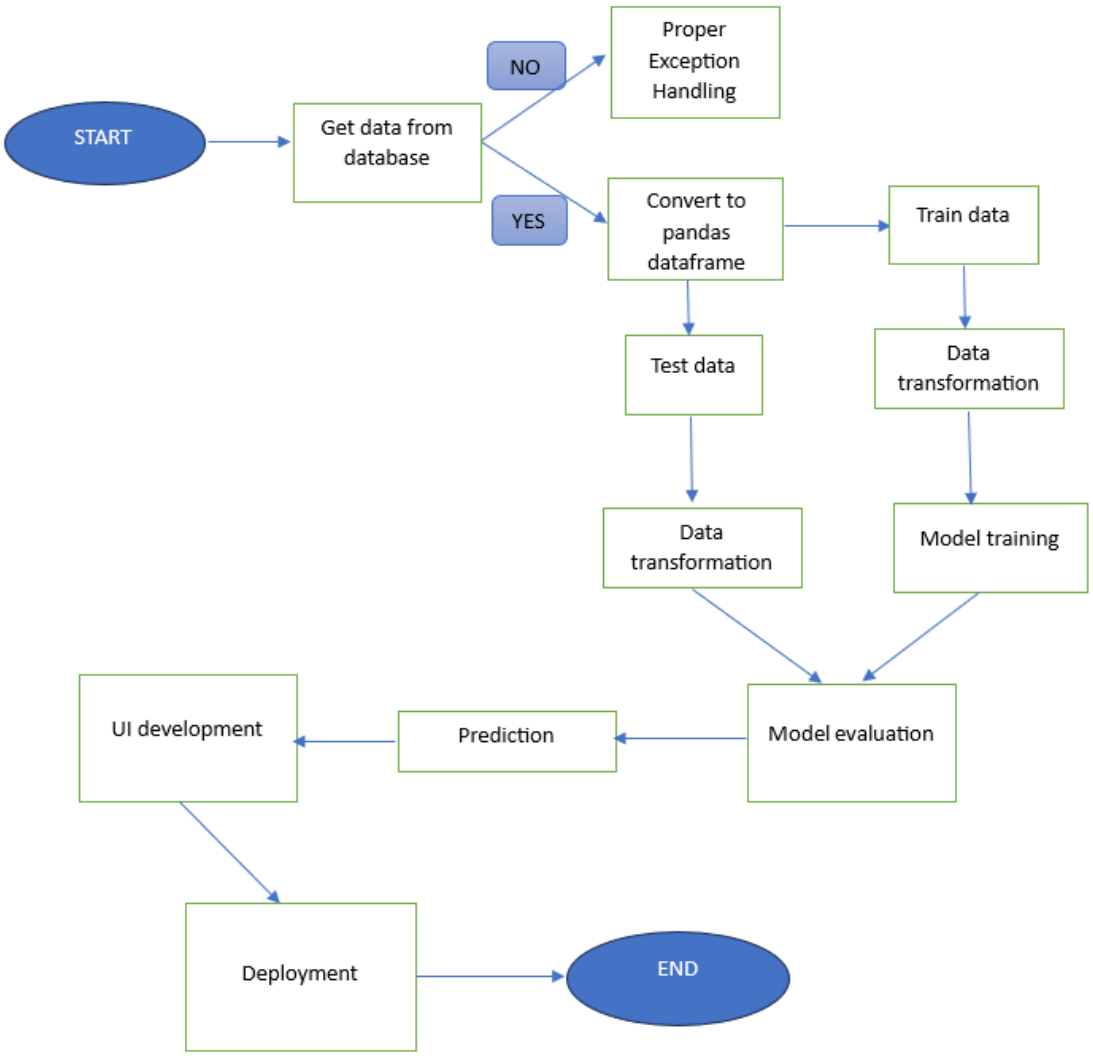
3 Technology stack

The technology stack comprises a dynamic blend of HTML, and CSS for an engaging front-end user experience. Python powers the back end, handling logic and data processing seamlessly. MongoDB, a NoSQL database, offers flexibility in managing evolving data structures. Amazon Web Services (AWS) for deployment ensures a scalable, reliable, and secure cloud infrastructure. This stack is not only versatile but also prioritizes user-centric design, emphasizing a visually appealing and interactive interface. The integration of these technologies streamlines development and deployment workflows, creating a robust foundation for credit card default detection system.

Front End	HTML/CSS
Backend	Python
Database	MongoDB
Deployment	AWS

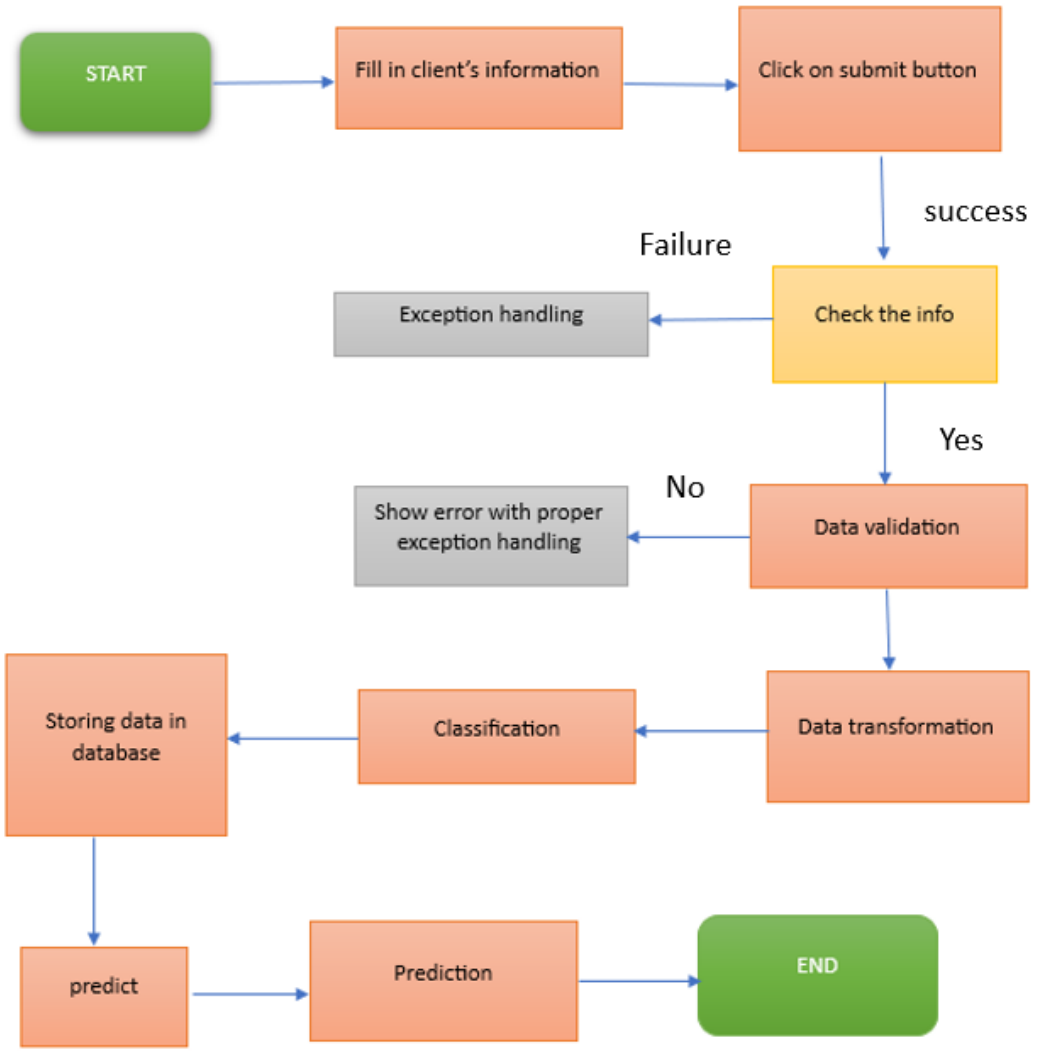
4 Model training/ validation workflow

The model training and validation workflow involves several key steps to ensure the effectiveness of the credit card default detection system. Initially, a dataset is collected and preprocessed to prepare it for training. Features such as credit limit, gender, education, and repayment history are selected. The data is then split into training and test sets. A supervised learning classification model, built using Python and popular machine learning libraries, is trained on the labeled data. The model's performance is assessed using test data, employing metrics such as accuracy, precision, recall, and F1 score. The final trained model is then ready for deployment in the credit card default detection system, providing accurate and reliable predictions based on user inputs.



5 User I/O workflow

The User I/O workflow for the credit card default detection system involves a user-friendly interface designed with HTML, and CSS. Users initiate the process, select the credit card default detection option, and provide relevant information such as credit limit, gender, education, and repayment history. The system incorporates exception handling to address errors or incomplete information, guiding users to correct input. Upon successful validation, the system uses a pre-trained supervised learning classification model, built with Python, to predict the likelihood of credit card defaults. The results, including prediction and additional insights, are presented to users through a clear and visually appealing interface, enhancing the overall user experience.



6 Conclusion

The architecture document for the Credit Card Default Detection system provides a thorough blueprint for the application's design and implementation. It covers critical aspects such as system architecture, classification algorithm details, API design considerations, deployment strategies, testing methodologies, and security measures. This document serves as a vital reference for stakeholders, offering insights into the system's robustness, reliability, and user-friendliness.
As the system evolves, the architecture outlined herein will guide its development, ensuring a well-informed and cohesive approach to the Credit Card Default Detection application.