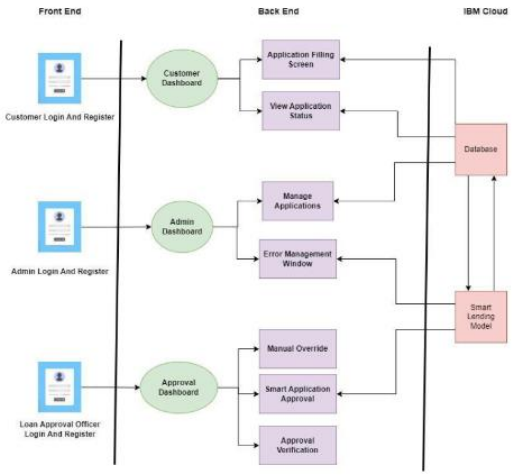
Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	03 October 2022
Team ID	PNT2022TMID52712
Project Name	Smart Lender- Applicant Credibility Prediction for Loan Approval
Maximum Marks	4 Marks

Technical Architecture:



Guidelines:

1. Include all the processes (As an application logic / Technology Block)
2. Provide infrastructural demarcation (Local / Cloud)
3. Indicate external interfaces (third party API's etc.)
4. Indicate Data Storage components / services
5. Indicate interface to machine learning models (if applicable)

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Users interact with the application with the help of a web UI	HTML, CSS etc.
2.	Building application	Getting user information from UI and feeding it to ML model	Python Flask
3.	Visualizing and analysing data	Reading and understanding the data properly with the help of visualization and analyzing techniques.	Python pandas, numpy, pickle, matplotlib, seaborn
4.	Pre-processing or cleaning data	Handling missing values, Handling categorical data, Handling outliers, Scaling Techniques	Python pandas
5.	Database	Loan Approval dataset	.csv file
6.	Cloud Database	Deploying the model on cloud	IBM cloud
7.	Machine Learning Model	Using machine learning model for predicting loan approval	Model building using classification algorithms such as Decision tree, Random forest, KNN, and xgboost.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	IBMCloud
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	IBMcloud provides layered security controls across network and infrastructure.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Web3.0 IBMCoud
4.	Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	HTML,CSS,JavaScript
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	Python

References:

https://c4model.com/

https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/

https://www.ibm.com/cloud/architecture

https://aws.amazon.com/architecture

https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d