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

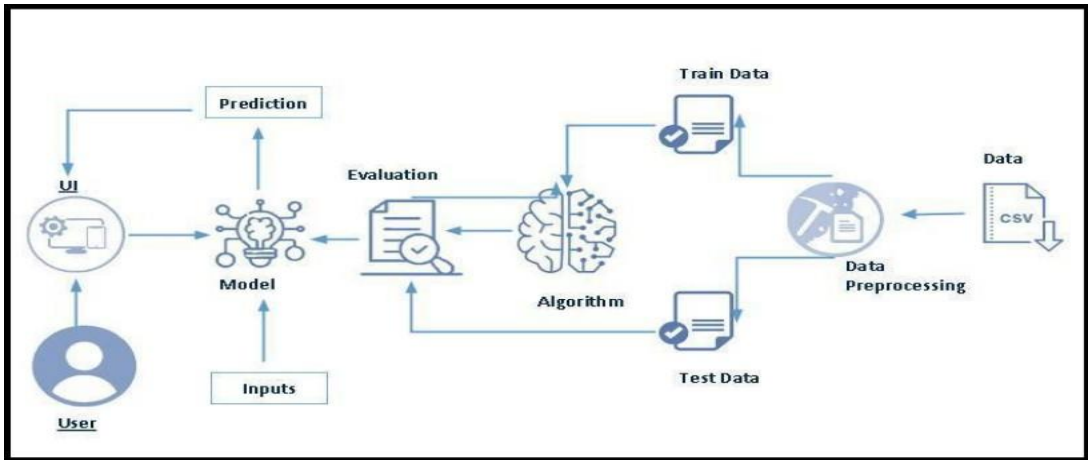
Date	03 October 2022
Team ID	PNT2022TMIDxxxxxx
Project Name	Project - xxx
Maximum Marks	4 Marks

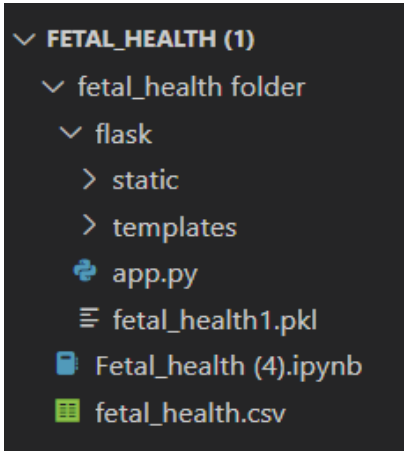
Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

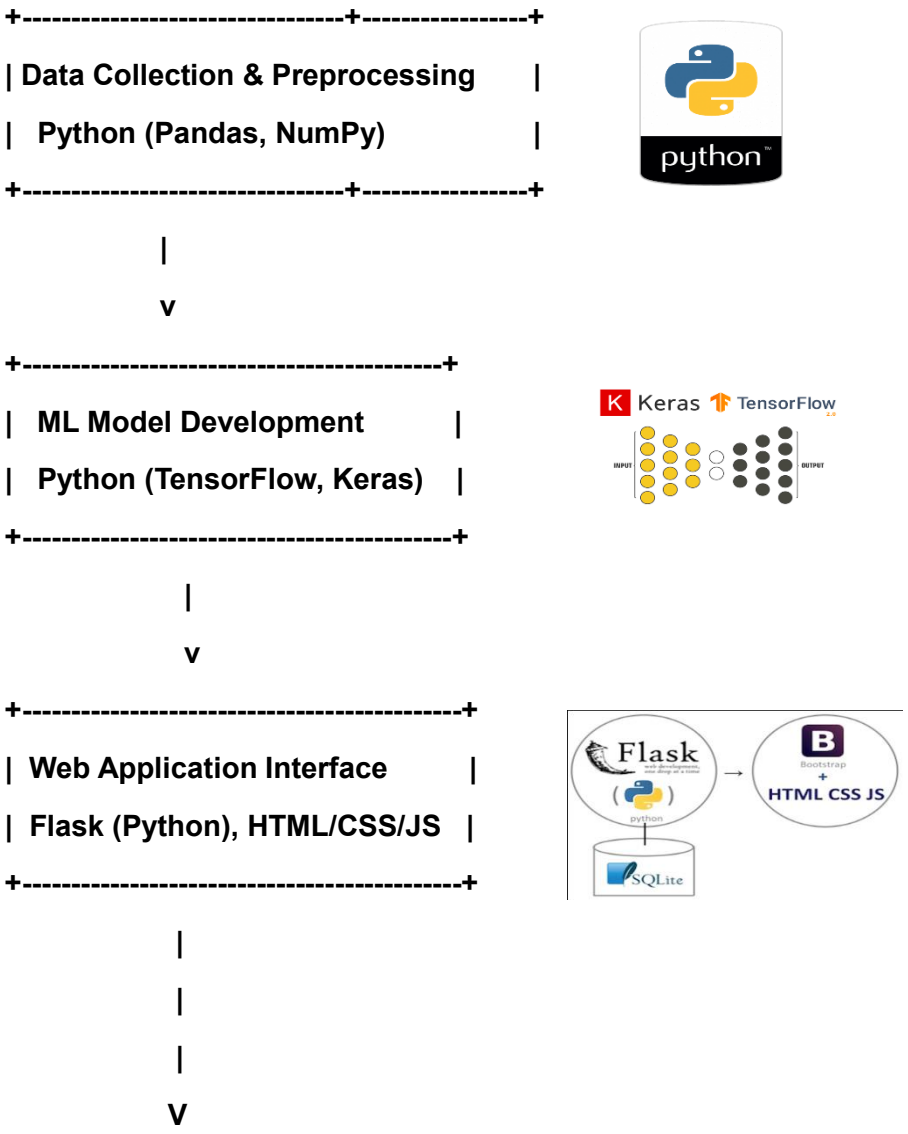
Example: Order processing during pandemics for offline mode

Reference: https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/





Technical Architecture:



+---+

| Cloud-based Storage |

| Amazon S3, Google Cloud Storage |

+---+

|

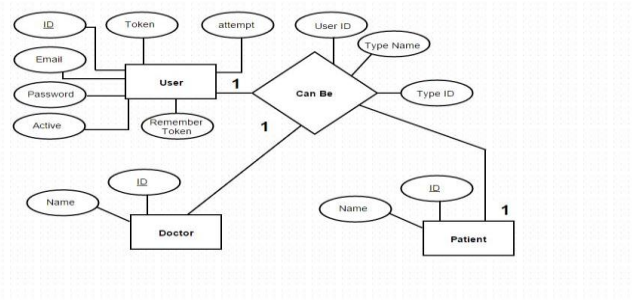
v

+---+

| External Healthcare Data Sources |

| Hospital Databases, APIs |

+---+



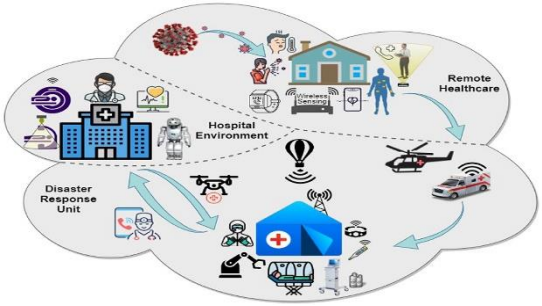


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, Flask
2.	Application Logic-1	Logic for a process in the application	Python
3.	Application Logic-2	Logic for a process in the application	**Python Libraries**:Pandas and NumPy,Scikit-learn **Machine Learning Frameworks**: TensorFlow or PyTorch
4.	Application Logic-3	Logic for a process in the application	**Python Libraries**:Pandas and NumPy,Scikit-learn, Flask or Django, AWS (Amazon Web Services) or Google Cloud Platform:
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL,MangoDB etc.
6.	Cloud Database	Database Service on Cloud	IBM AWS cloud architecture, Google cloud services etc.
7.	File Storage	File storage requirements	IBM Block Storage, Amazon cloud storage(S3) or Other Storage Service or Local Filesystem
8.	External API-1	Purpose of External API used in the application	Medical Database API:
9.	External API-2	Purpose of External API used in the application	Medical Imaging API:
10.	Machine Learning Model	Purpose of Machine Learning Model	Classification models ,CNN, Random Forest etc
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local **Server Configuration**: Utilizing services from cloud providers like Amazon Web Services (AWS), Google Cloud Platform (GCP), or Microsoft Azure. **Deployment Environment**: Deploying the application and machine learning models on cloud servers or platforms (e.g., AWS EC2, Google Compute Engine, Azure App Service).	Local, Cloud Foundry, Kubernetes. AWS EC2, Google Compute Engine, Azure App Service,etc

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Technology of Opensource framework TensorFlow, Keras, OpenAI, Scikit learn
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	e.g. SHA-256, AES-256 Role-Based Access Control (RBAC), HTTPS/TLS),Encryptions, IAM Controls, OWASP, etc.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Spring Boot (Java), Flask (Python), or Node.js for developing microservices. cloud services such as AWS ECS, Google Kubernetes Engine (GKE), or Azure Kubernetes Service (AKS) for scalable infrastructure provisioning and management.
4.	Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	Utilize technologies like NGINX, HAProxy, or cloud-based load balancers (AWS Elastic Load Balancing, Google Cloud Load Balancing). AWS EC2, Google Cloud VM) AWS CloudFront.
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	**Caching Mechanisms:** Redis, Memcached **Load Balancing:** NGINX, HAProxy, AWS Elastic Load Balancing; **Browser Cache Control**: HTTP headers; **CDNs**: Cloudflare, AWS CloudFront, Akamai

References:

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

Project Planning Phase

Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)

Date	18 October 2022
Team ID	PNT2022TMIDxxxxxx
Project Name	Project - xxx
Maximum Marks	8 Marks

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	Phase 1;Ideation phase: Emapthy Map; Brainstorming Map	20	High	Ruwan Aryan; Keerthi krishna.s
Sprint-2		USN-2	Phase 2: project design phase Proposed Solution , Solution Architecture, Data Flow Diagram (DFD	20	Medium	Sreelasya Changalsetty
Sprint-3		USN-3	Phase 3: Project Planning. Technology Stack, Sprint Planning	10	Medium	Gregory.T.R
Sprint-4		USN-4	Phase 4: project development Model building ,UI implementation, Flask, HTML	30	High	RuwanAryan, Gregory.T.R, Keerthikrishna.s
Sprint-5	Login Dashboard	USN-5	Phase 5: Performance and final Submission Phase: Solution Performance, Project Documentation,	20	Medium	Sreelasya Changalsetty, RuwanAryan, Gregory.T.R, Keerthikrishna.s

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2023	29 Oct 2023	20	29 Oct 20223
Sprint-2	20	6 Days	30 Oct 2023	04 Nov 2023	20	04 Nov 2023
Sprint-3	10	6 Days	05 Nov 2023	11 Nov 2023	10	11 Nov 2023
Sprint-4	30	6 Days	12 Nov 2023	18 Nov 2023	30	18 Nov 2023
Sprint-5	20	6 Days	19 Nov 2023	25 Nov 2023	20	25 Nov 2023

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

AV= avg velocity/sprint duration=

20/6=3.33

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.