

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

| | |
|---------------|--|
| Date | 20 October 2022 |
| Team ID | PNT2022TMID09848 |
| Project Name | Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation |
| Maximum Marks | 4 Marks |

Technical Architecture:

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

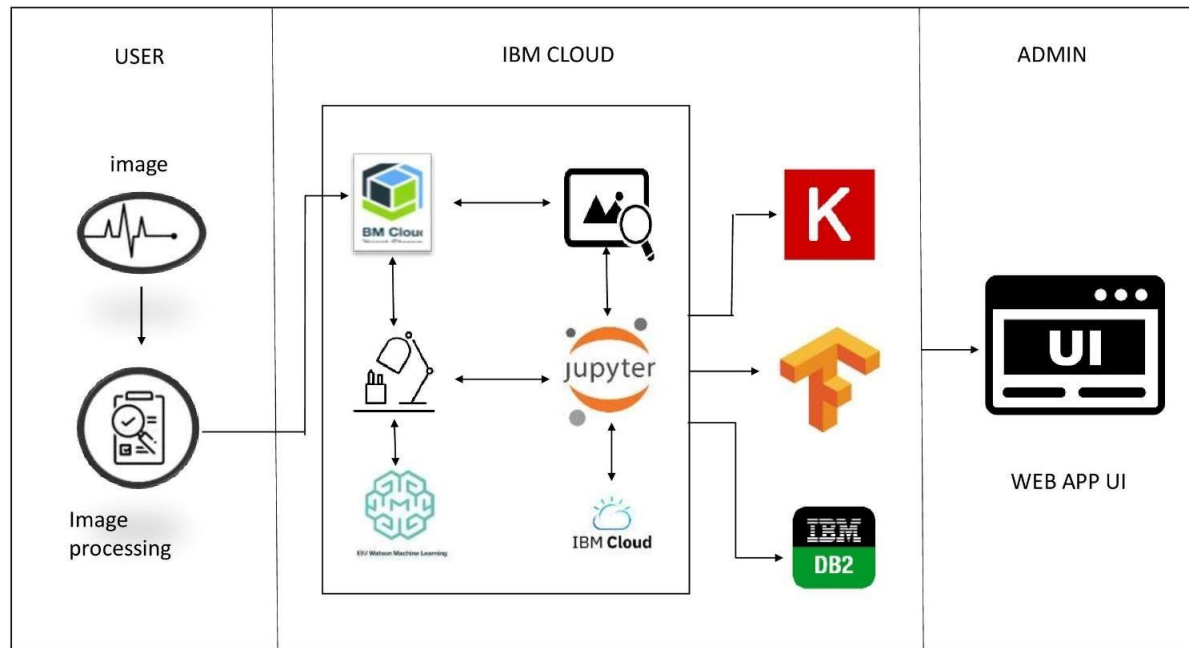


Table-1 : Components & Technologies:

| S.No | Component | Description | Technology |
|------|--------------------|---|--|
| 1. | User Interface | A web application using Flask | HTML, CSS, JavaScript / Angular Js / React Js etc. |
| 2. | Image augmentation | Creating training examples from the existing ones | Python |

| | | | |
|-----|-------------------------|---|--|
| 3. | Building the model | Building the model by adding various layers and training the model | IBM Watson Cloud storage, IBM cloud Pak Data |
| 4. | Testing the model | Testing the model | IBM Watson Machine Learning |
| 5. | Database | Database contains the ECG signals of various arrhythmia patients | IBM cloud storage |
| 6. | Cloud Database | Database Service on Cloud | IBM DB2 |
| 7. | File Storage | File storage requirements | IBM Block Storage |
| 8. | Anaconda Navigator | The web application is deployed and run on the local host with the help of anaconda navigator | Anaconda Navigator |
| 9. | Tensorflow | For numerical computation that makes machine learning and developing neural networks | Tensorflow library, Jupyter Notebook |
| 10. | Machine Learning Model | To classify the type of arrhythmia with Images uploaded by the users. | Image Classification Model |
| 11. | Infrastructure (Server) | Application Deployment on Local System | Local |

Table-2: Application Characteristics:

| S.No | Characteristics | Description | Technology |
|------|--------------------------|-------------------------|------------------------------------|
| 1. | Open-Source Frameworks | Flask, Jupyter Notebook | Flask, Jupyter Notebook |
| 2. | Security Implementations | use of API key | IAM Controls, wml control(API Key) |
| 3. | Scalable Architecture | Micro-services | API gateway |

| | | | |
|----|--------------|-----------------------------|-------------------------------|
| 4. | Availability | Based on availability zones | IBM Watson studio |
| 5. | Performance | Using CDNs | IBM Content Delivery Network. |