**PROJECT DESIGN PHASE-II**

**TECHNOLOGY STACK (ARCHITECTURE & STACK)**

|  |  |
| --- | --- |
| DATE | 19 OCTOBER 2022 |
| TEAM ID | PNT2022TMID23524 |
| PROJECT NAME | SMART SOLUTION FOR RAILWAYS |

**SUBMITTED BY**

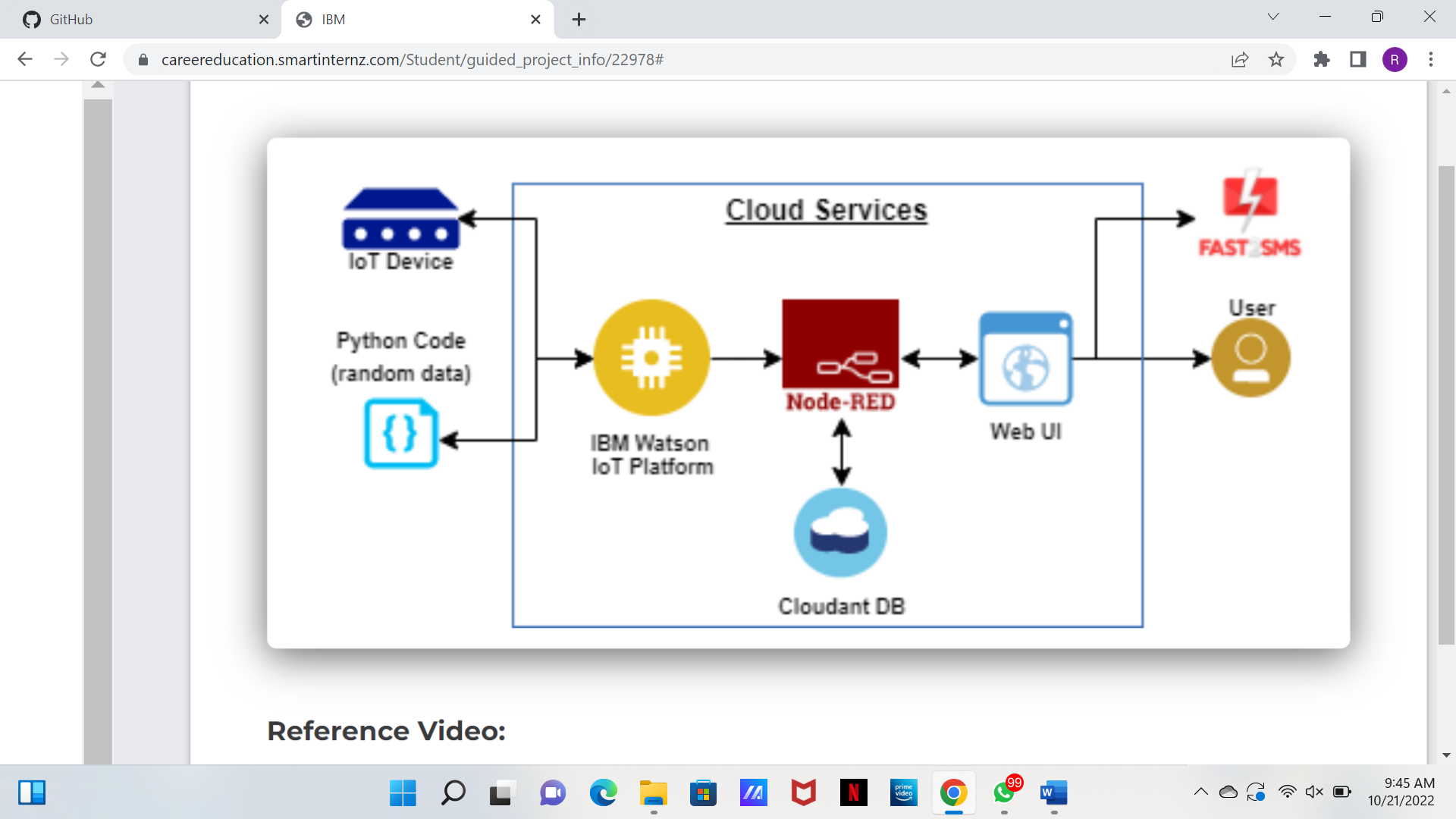
**SWETHA .G (113219041121)**

**MEENA.C (113219041066)**

**RASIKA.J (113219041094)**

**MALAVIKA.R (113219041062)**

**TECHNICAL ARCHITECTURE:**

****

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

**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, JavaScript, React Js,  Flutter |
| 2. | Application Logic-1 | Logic for a process in the application | MIT app inventor, |
| 3. | Application Logic-2 | Logic for a process in the application | IBM Watson STT service |
| 4. | Application Logic-3 | Logic for a process in the application | IBM Watson Assistant |
| 5. | Database | Data Type, Configurations etc. | MySQL |
| 6. | Cloud Database | Database Service on Cloud | IBM DB2, Firebase |
| 7. | File Storage | File storage requirements | IBM Block Storage |
| 8. | External API-1 | Purpose of External API used in the application | Stripe Payment API |
| 9. | Machine Learning Model | Purpose of Machine Learning Model | Recommendations system |
| 10. | Infrastructure (Server / Cloud) | Application Deployment on Cloud system | Kubernetes, Heroku |

**Table-2: Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | List the open-source frameworks used | React JS, Flutter |
| 2. | Security Implementations | List all the security / access controls implemented, use of firewalls etc. | SHA-256, Encryptions, IAM Controls, OWASP |
| 3. | Scalable Architecture | Justify the scalability of architecture (3 – tier, Microservices) | 3 – Tier architecture |
| 4. | Availability | Justify the availability of application (e.g. use of load balancers, distributed servers etc.) | Nginx Load balancers, Fault tolerant systems |
| 5. | Performance | Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN’s) etc. | Multicore processors for servers |