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

|  |  |
| --- | --- |
| Date | 15 October 2022 |
| Team ID | PNT2022TMID24961 |
| Project Name | Web Phishing Detection |
| Maximum Marks | 4 Marks |

Technical Architecture:

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

Web Phishing Detection Diagram Reference: What is phishing? | IBM

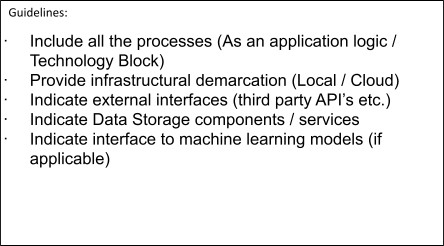
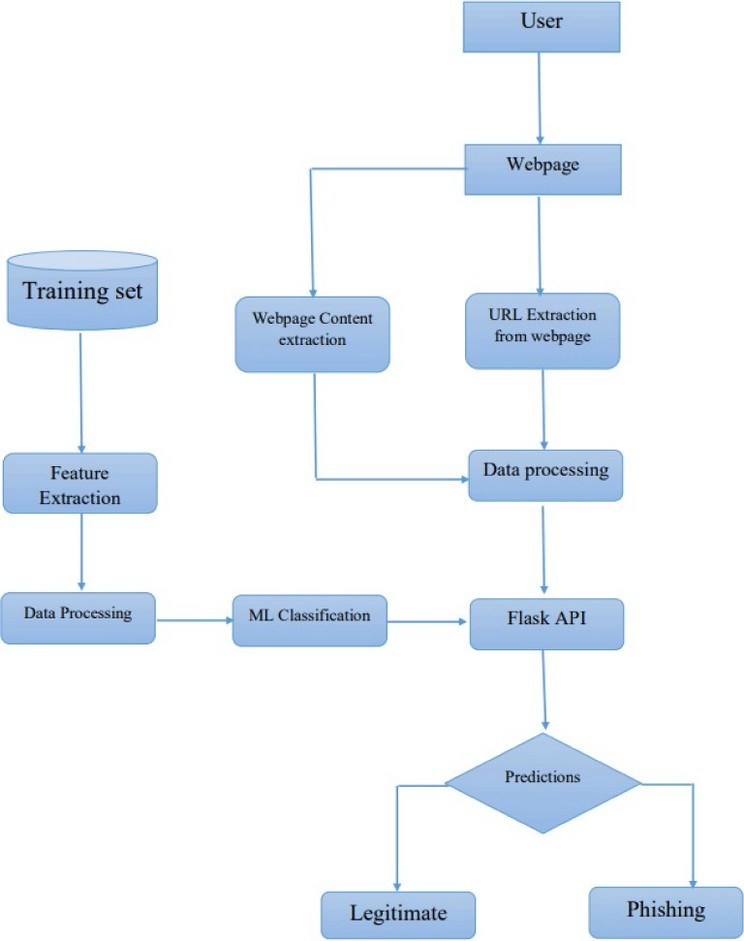


Table-1 : Components & Technologies:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1. | User Interface | Dynamic Web UI | HTML, CSS, JavaScript, Bootstrap |
| 2. | Application Logic-1 | User Registration/Login | IBM API Connect Service, Gmail API, LinkedIn API |
| 3. | Application Logic-2 | Web app that predicts if the link is a phishing site or  not | Flask API, Python |
| 4. | Database | Store user input links in the database | MongoDB |
| 5. | Cloud Database | Database Service for storing user profile | IBM DB2, IBM Cloudant etc. |
| 6. | File Storage | Store the datasets used for prediction | Local Filesystem |
| 7. | External API-1 | User Registration/Login using email and password | IBM API Connect |
| 8. | External API-2 | User Registration/Login using external apps | Gmail API, LinkedIn API |
| 9. | Machine Learning Model | Machine Learning Model for web phishing  detection | Logistic Regression Model |
| 10. | Infrastructure (Server / Cloud) | Application Deployment on Local System / Cloud | Local, Render, IBM Cloud |

Table-2: Application Characteristics:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | High-level open-source frameworks | Docker, Flask, Bootstrap |

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 2. | Security Implementations | It is the security discipline that makes it possible for the right entities (people or things) to use the right resources (applications or data) when they need to, without interference, using the devices they want to use. | IAM Controls of IBM |
| 3. | Scalable Architecture | Compose is a tool for defining and running multi-container Docker applications. With a single command, can create and start all the services from the configuration. | Docker, Docker Compose |
| 4. | Availability | It can balance the load traffic among the servers to  help improve uptime. Can scale applications by adding or removing servers,  with minimal disruption to traffic flows. | IBM Cloud load balancers |
| 5. | Performance | It provides performance feedback such as page size and how long it takes to load a page, and can show the impact new features have on the performance of the site. | IBM’s Speed Curve and Delivery Pipeline |

References:

What is identity and access management? IAM, SSO, MFA and IDaaS definitions | IBM SpeedCurve: Visually monitor an app across platforms - IBM Garage Practices

Load Balancer | IBM