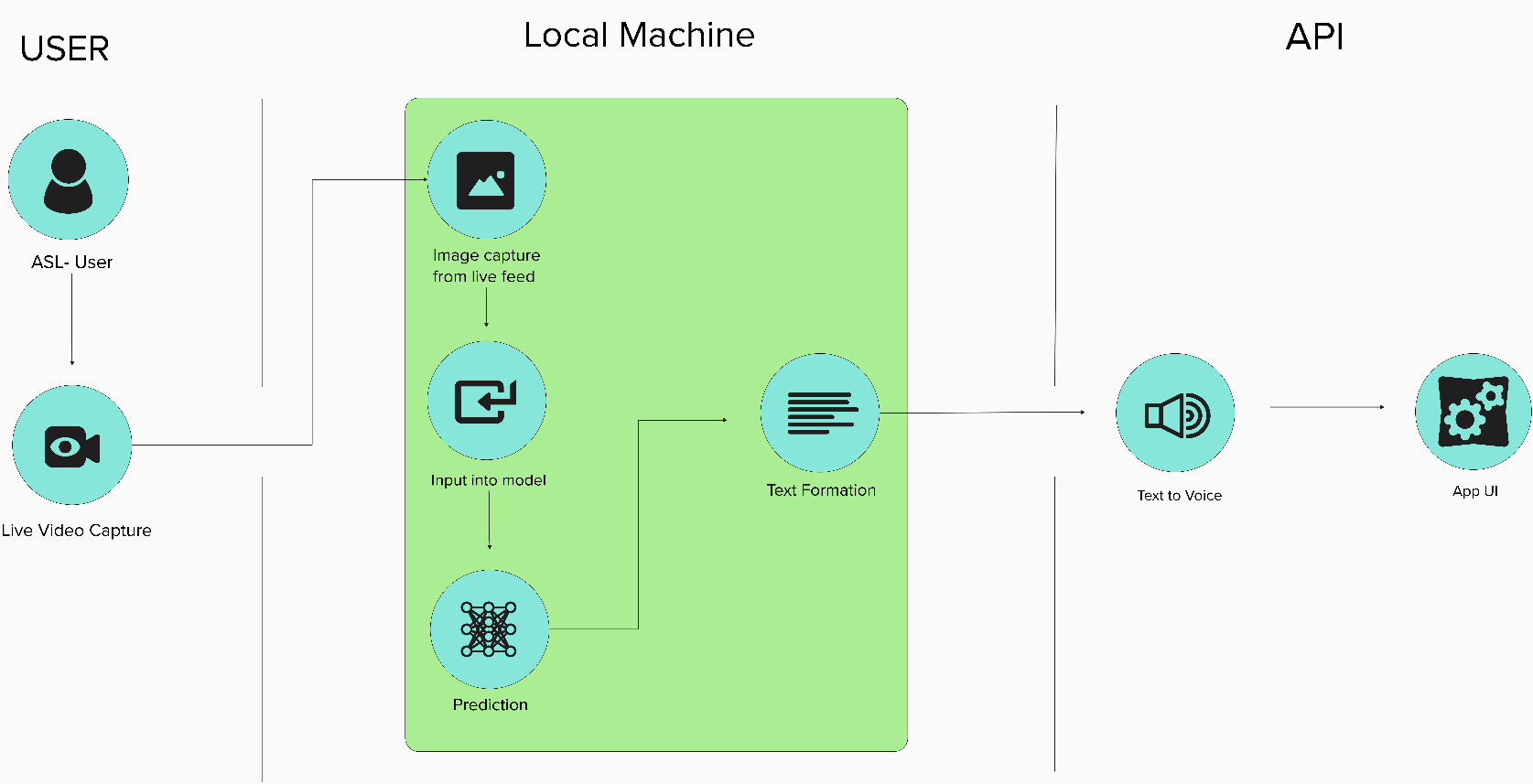
**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 27 October 2023 |
| Team ID | Team-592961 |
| Project Name | Project – ASL – Alphabet Image Recognition |
| Maximum Marks | 4 Marks |

**Architectural Diagram:**



**Table-1 : Components & Technologies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S No** | **Component** | **Description** | **Technology** |
| 1. | User Interface | Allows users to interact with the ASL recognition system. Includes elements for webcam access, video display, and interaction with the system | HTML, CSS, JavaScript , react |
| 2. | Application Logic-1 | It represents the core logic of the ASL recognition system. It involves handling user interactions, capturing video frames, and initiating the ASL recognition process | Python (Flask) |
| 3. | Application Logic-2 | This component of the application logic may include further processing of video frames and interaction with the ASL recognition model. It extends the core functionality. | Python, JavaScript |
| 4. | Application Logic-3 | additional processing or user-specific functionality, such as managing user profiles or maintaining session data. | Python, JavaScript |
| 5. | Database | To store user live feed for processing etc | MongoDB |
| 6. | File Storage | To store all the essential components | Local Filesystem |
| 7. | External API-1 | An external API used for converting recognized ASL signs into voice output (Text-to-Speech, TTS) to enhance user communication. | gTTS/ IBM Watson Text to Speech |
| 8. | Machine Learning Model | It is responsible for recognizing ASL signs from video frames | CNN(Convolutional Neural Network), Tensorflow |
| 9. | Infrastructure (Server / Cloud) | Local Server Configuration | Local |

**Table-2: Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | **OpenCV (Open Source Computer Vision Library)**  TensorFlow(deep learning) | C++ (core library), Python (bindings)  Python |
| 2. | Scalable Architecture | A microservice architecture is more aligned for this project as the API can be used as a standalone microservice which can be integrated into web applications. | Corresponding Technologies used |
| 3. | Performance | The accuracy score has been tested between the training data and testing data. | sklearn |