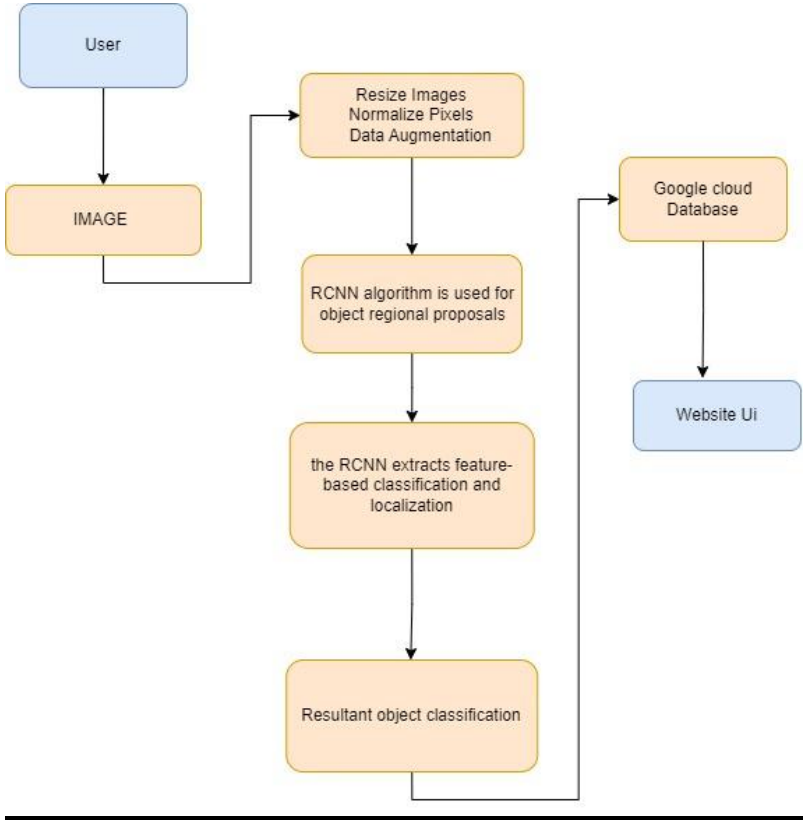
Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	03 October 2022
Team ID	Team-593089
Project Name	Deep Learning Model for Detecting Diseases in Tea Leaves
Maximum Marks	4 Marks

Technical Architecture:



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

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI	HTML, CSS, JavaScript
2.	Application Logic-1	Creating a deep learning model for image classification	Python
3.	Application Logic-2	Allowing user to upload image for tea leave disease classification	python
4.	Database	Image	NoSQL
5.	Cloud Database	Google cloud	Google cloud.
6.	File Storage	File storage requirements	Local Filesystem
7.	External API-1	To use the dataset	Kaggle API
8.	Machine Learning Model	RCNN (Region-based Convolutional Neural Network)	Image Recognition Model
9.	Infrastructure (Server / Cloud)	Application Deployment on Local System	Local.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Security Implementations	• Ensure that uploaded file names do not contain potentially harmful characters or escape sequences. • implement proper access controls to prevent unauthorized access to the uploaded images.	File Name Sanitization, Access Controls
2.	Scalable Architecture	• Improving the deep learning model by increasing the dataset • Improving the architecture of the model	Python, real time data
3.	Availability	• load balancing technology distributes incoming web traffic across multiple servers. • Cloud hosting platforms Google Cloud, • Cache website content on both the server and client sides to reduce server load and improve page loading times, contributing to better availability.	Load Balancing, Cloud Hosting, Content Caching
4.	Performance	• CDNs use a network of distributed servers to deliver website content to users from the nearest server location, reducing latency and improving load times, • Caching stores frequently accessed content, such as images and web pages, in temporary storage, reducing the need to fetch them from the server on every request.	Caching, Content Delivery Networks (CDNs)