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

Date	10/MAY/2023
Team ID	IBM--18527-1682584903
Project Name	AI Enabled Car Parking Using OpenCv
Maximum Marks	4 Marks

Technical Architecture:

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

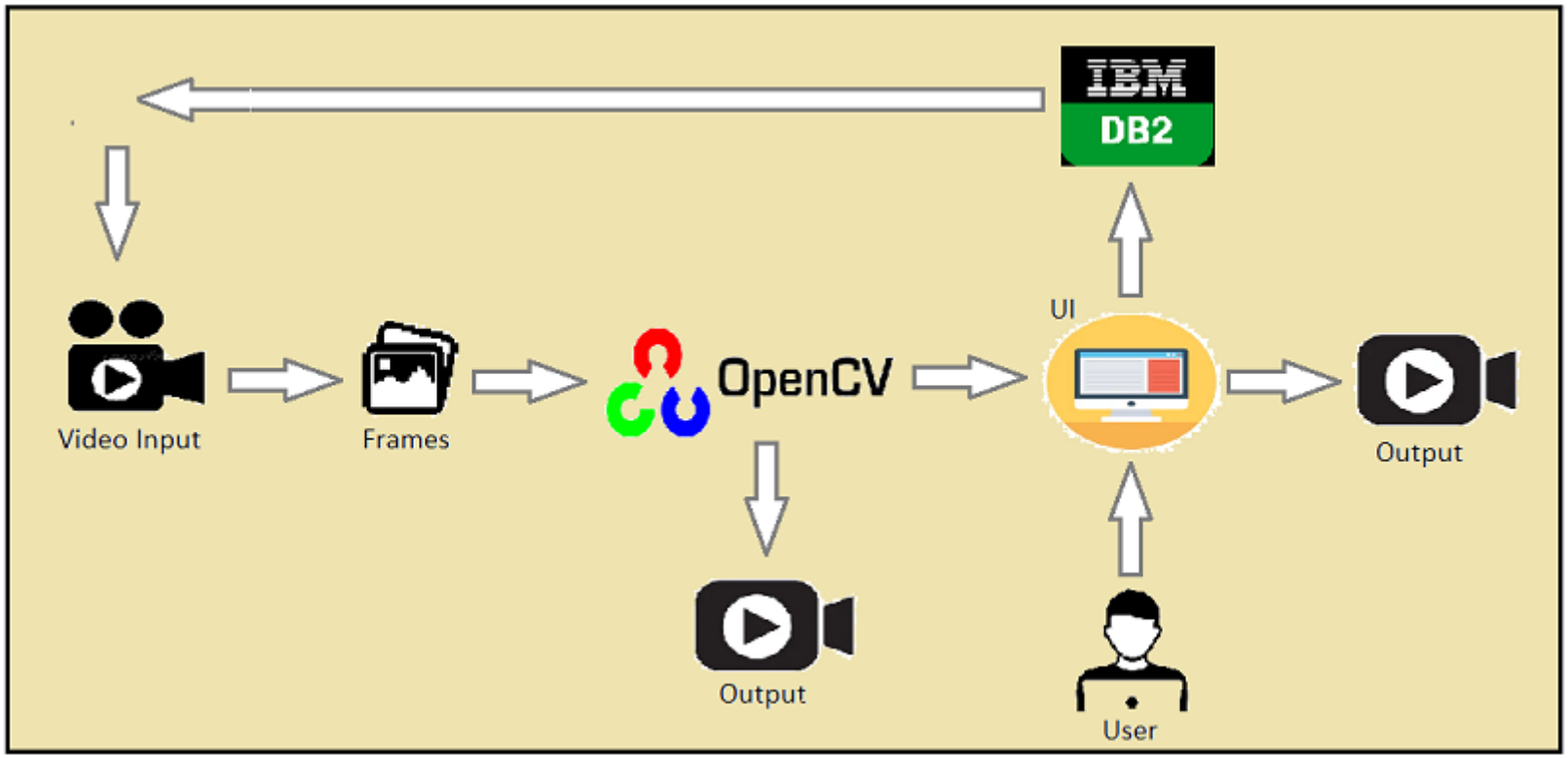
IBM
DB2
Video Input
Frames
OpenCV
UI
Output
Output
User

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	Cameras	Capture video input from parking lot	CCTV cameras
2.	OpenCV	Analyze frames and detect parking space occupancy and license plates	Python using OpenCV library
3.	IBM DB2	Store parking lot occupancy data and configurations	IBM DB2 database
4.	Flask	Develop server-side web application for user interface	Python Flask or Django web framework
5.	Bootstrap/Materialize	Develop responsive and mobile-friendly user interface	HTML, CSS, JavaScript, Bootstrap or Materialize front-end library
6.	TensorFlow	Develop deep learning algorithms for vehicle detection and license plate recognition	Python TensorFlow
7.	IBM Cloud	Host cloud-based server and services	Cloud hosting platforms
8.	Git	Manage source code and collaboration	Version control system
9.	Docker	Containerize and deploy the application	Containerization technology
10.	Kubernetes	Orchestrate and manage containerized application	Container orchestration platform

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Ability to use open-source frameworks for efficient and cost-effective development	OpenCV, TensorFlow, PyTorch, Flask
2.	Security Implementations	Ability to implement strong security measures to protect user data and parking lot information	SSL encryption, secure cloud hosting, user authentication, and firewalls
3.	Scalable Architecture	Ability to handle an increasing number of parking spaces and users	Cloud hosting platforms, containerization technology, and container orchestration platform
4.	Availability	Ability to ensure high availability of the system with minimal downtime	Load balancing, redundancy, and failover mechanisms
5.	Performance	Ability to detect parking space occupancy and recognize license plates with high accuracy and speed	OpenCV, TensorFlow or PyTorch, and optimized algorithms

References:

Build a simple smart parking project using python and OpenCV | by Razvan Vilceanu | The Startup | Medium

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

Written and submit by.

AJAYKUMAR.A(TEAM LEADER)

REGISTER NUMBER:6BD654E34A81AD6895846B94CBCB1BE6

EMAIL :ajaykumar75025@gmail.com

MOBILE NUMBER :7502522887

DATE OF BIRTH :31/01/2000

DEGREE :Bachelor of Engineering/Technology

BRANCH :B.Tech. Information Technology

COLLEGE :ULTRA College of Engineering& Technology