**General Technical Approach:**

Create a web application which user can use as an online tool to detect an object from images. Image after detection will return a list of objects. The application will be designed to leverage controller and view in a Flask framework. A Python application uses a TensorFlow Object Detection model to provides an API to the web application to detect images’ objects. A MySQL database is used store object detection results.

### Key Technical Design Decisions:

Embedding: The application is required TensorFlow Object Detection API. A Python application using TensorFlow will be running on the server to create a TensorFlow Object Detection API. This application needed a high-performance machine to perform training and testing Object Detection model. In this project scope, a computer with GPU is used to host the Flask server and run the TensorFlow Object Detection API.

Back End Service: The TensorFlow Object Detection application is required to process and detect objects in the image; a back-end service using Flask will consume TensorFlow Object Detection API and get the detail of the objects to belong to the image to store in the database. The MySQL database is used for the project.

Front End Web Application: A front-end web application using HTML, JavaScript to implement a detect image’s object application. The application will leverage some common functions and display the object detection result, as well as display the images which contain specific objects when user look for it

TensorFlow Object Detection API: TensorFlow Object Detection API is an open source framework built on top of TensorFlow that makes it easy to construct, train and deploy object detection models. TensorFlow provides several object detection models (pre-trained classifiers with specific neural network architectures) . In this project, a MobileNet SSD model is used to train the dataset and export the trained model that can use to detect the image’s object.

|  |  |
| --- | --- |
| Proof of Concepts | |
| **Description** | **Rational** |
| 1. TensorFlow | To create object detection application |
| 2. PASCAL VOC 2012 | To provide training set of labelled images. There are twenty object classes that have been selected in the dataset are:   * *Person:* person * *Animal:* bird, cat, cow, dog, horse, sheep * *Vehicle:* aero plane, bicycle, boat, bus, car, motorbike, train * *Indoor:* bottle, chair, dining table, potted plant, sofa, tv/monitor |
| 3. MobileNet SSD (Single Shot Detection) | The algorithm to detect object from image |
| 4. Flask | to run the application. In this project, Flask server is deployed on Azure Virtual Machine |
| 5. Python application to detect objects from image and return object detail in JSON format | In this project scope, python application is deployed on local computer |
| 6. MySQL database | To store user information, Image and it’s object detection |
| 7. Host application and database | User localhost. |
| Hardware and Software Technologies | | |
| 1 – MySQL 5.6: open-source relational database management system. It I GPL(version 2) license | | |
| 2 – Python 3.6: developed y Python Software Foundation | | |
| 3 – Flask 1.0.2: microframework for Python. It is BSD licensed. | | |
| 4 – TensorFlow 1.10: open source software library for high performance numerical computation. Originally developed by researchers and engineers from the Google Brain team within Google’s AI organization | | |
| 5 – PASCAL VOC 2012 dataset from PASCAL VOC project. | | |
| 6 – Windows 10 2.2 GHZ x 16GB RAM x 8GBGPU x 250 GB SSD | | |