Ammar Hany

Computer and Software Engineer

Projects

Company’s Farm-Api

Programming Languages: Python
Database: Firebase, Google Storage

- A robust RESTful API developed in Python Flask and deployed on Google Cloud Run.
- Utilizes Google Cloud Firestore and Google Storage Buckets for efficient database storage.
- Implements various security measures including Token-based and Basic Authentication.
- Designed for scalability and efficiency, with automated updates and deployment through Docker and GitHub integration.

Company Farm-Estrus-Id-Score api

Programming Languages: Python

- A RESTful API developed to process data and apply an ID Multi-Model.
- Utilizes Ultralytics YOLOv8 for detecting specific features within the data and incorporates a scoring function to evaluate the accuracy of detections.
- Optimized for deployment on Google Cloud Run, with meticulous handling of all aspects of ID Model training and testing.

Company Farm-estrus-Jetson Pipeline

Programming Languages: Python
Database: Firebase, Google Storage

- A Python-based application designed to run an AI Estrus Alert system on Jetson devices.
- Efficiently connects to multiple cameras via RTSP and initiates predictions for each camera.
- Utilizes hardware acceleration and optimized for enhanced efficiency in real-life deployment scenarios.

Company IoT Gate Pipeline

Programming Languages: Python
Database: Firebase, Google Storage

- An IoT app designed for deployment in farms, supporting EU RFID Tag Readers for automatically reading each cow's European ID.
- Compatible with RTSP and USB cameras, offering support for Linux and Windows platforms.
- Features multithreading for efficient connection to multiple cameras and serves as a data collection tool for AI and computer vision models.

Company Data Checker Pipeline

Programming Languages: Python

- Designed to complement the company’s IoT Gate, automating the processing of video frames and filtration using AI.
- Prepares datasets with specified IDs and streamlines the data collection process for further analysis.

Company’s All Computer Vision AI Scripts

Programming Languages: Python

- A comprehensive tool for training computer vision models using Ultralytics YoloV8.
- Encompasses various functionalities such as data augmentation, dataset preparations, YAML handling, and model testing.

Company’s AI Analysis App

Programming Languages: Python

- A Python application with a GUI for validating the accuracy of ID models through user testing.
- Fetches data from databases, processes it, and presents it to the user for validation with seamless integration with Microsoft Excel for further analysis.

Company’s Live Object Counter

Programming Language: Python

- Developed during an internship, the Live Object Counter is a computer vision project aimed at counting and annotating objects in real-time.
- Utilizes Python programming language along with OpenCV, Ultralytics, and Supervision libraries for efficient object detection and counting.
- Optimized for efficiency to ensure smooth operation on Jetson devices.

Teknofest 2023 Contest Computer Vision Project

Programming Language: Python
Libraries: Ultralyicts, Opencv

- Developed in collaboration with Toros University's Smart Lab for the Teknofest Contest, achieving the 25th rank, was viewed by 2.5 million visitors.
- Focuses on an Autonomous Harvesting Robot project, leveraging computer vision techniques for fruit and vegetable harvesting.
- Built a computer vision model using the YOLOv8 architecture, integrated with a Flask-based frontend for user interface interaction.
- Features automatic object cropping and color detection for fruits and vegetables, enhancing harvesting efficiency.

Self-Driving Car Using Behavioral Cloning

Programming Language: Python
Libraries: Tensorflow.Keras

- A machine learning project focused on training a self-driving car model using behavioral cloning.
- Utilizes manual driving to generate training data, which is then used to train a deep learning model based on convolutional neural networks (CNNs).
- Achieves autonomous driving capability without crashes, replicating human behavior using computer vision techniques.
- Completed as a graduation project, with model training conducted on a personal computer equipped with an RTX 2060 GPU.

Engineering Firm Website

Programming Language: HTML, CSS, JS and PHP
Database: MYSQL
Framework: Laravel 9 and Bootstrap

- Developed to showcase projects associated with an engineering firm, featuring an admin area for project management and user management.
- Utilizes HTML, CSS, JavaScript, and PHP within the Laravel 9 framework, offering role-based access control managed by the admin.
- SEO compatible, with dynamically generated meta tags from the admin area to enhance search engine visibility.

Real Estate Website

Programming Language: Javascript Wix Velo

- SEO Friendly Multilingual Website for showing newly built buildings in UAE With the objective of getting leads from buyers.
- Includes features like sending leads through email.
- Built with CMS, JavaScript, and WIX Velo.

IStone Company Software

Programming Language: C#, XAML
Framework: WPF
Database: MSSQL

- Developed using C# and XAML within the WPF framework, with MSSQL deployed as a server database.
- A role-based Windows application tailored for managing customers, tracking work status, inventory, and payments within the company.
- Features an authorization system for employees, managers, and owners, ensuring secure access to relevant functionalities.
- Custom-built to meet the specific requirements of the company, providing real-time project status updates accessible from any device on the local server.

Inventory System

Programming Language: C#, XAML
Database: SQLite
Framework: WPF

- A desktop application built using C# and XAML on the WPF framework.
- Designed for organizations to efficiently manage their inventory.
- Features a user-based login system, ensuring each user has access to their own inventory database.
- Users can add, remove, or edit items in their inventory and track the history of inventory changes, including quantity adjustments.
- Supports server deployment for multiple users to access simultaneously.
- Implements security measures such as DDOS protection and comprehensive logging of system activities.
- Supports automatic deployment and database building for a hassle-free setup and run experience. Specifically developed for Windows platforms.

Web Crawler

Programming Language: C#, XAML

Framework: WPF
Database: MSSQL

- Developed with C# and XAML within the WPF framework, utilizing MSSQL for data storage, with a focus on **Object Oriented programming** (OOB).
- A robust web crawler designed to traverse websites and extract all anchor links from the HTML structure.
- Implements multithreading to ensure smooth GUI operation without any lag.
- Capable of crawling an entire website by recursively following links within the provided domain.
- Features filtration capabilities to exclude websites outside the provided domain link.

Windows Notes App

Language: C#, XAML
Framework: WPF
Database: SQL

- A desktop application built using C# and XAML on the WPF framework.
- Designed for taking notes and storing them in a SQL database.
- Supports real-time note searching, editing, and deleting functionalities.
- Login-based, ensuring each user has their own set of notes.
- Built with object-oriented principles for fast performance and supports background threading.
- Implements extensive validation rules for a bug-free and fully functional experience.
- Specifically designed for Windows platforms.

School Library System

Programming Language: C#, XAML

Framework: WPF
Database: MSSQL

- Developed using C# and XAML within the WPF framework, with MSSQL utilized as the local database.
- A comprehensive library system designed for efficient management of books, featuring inventory and stock tracking functionalities.
- Implements roles-based authorization for Students, Lecturers, and Admins to ensure secure access.
- Includes time tracking for rented books and a notification system for important events.
- Admin privileges include role confirmation, stock management, and approval of returned books.
- Built-in rules and validation ensure smooth usage, with support for server deployment enabling multi-device access.
- Compatible with Windows operating systems.

School Management System

Programming Language: Java

Database: SQlite

- Developed in Java using SQLite as the local database for efficient data management.
- A versatile management system catering to course selection, exam scheduling, and weekly schedule management.
- Implements roles-based authorization for students and lecturers to access relevant functionalities.
- Lecturers have the ability to manage grades and courses, make announcements, and schedule exams and weekly activities.
- Compatible with Windows operating systems.

Cashier System
Programming Language: C#, XAML

Framework: WPF
Database: MSSQL

- Developed using C# and XAML within the WPF framework, with MSSQL serving as the local database.
- A robust cashier system equipped with order history and inventory management capabilities.
- Features an auto-stock tracking system to ensure inventory accuracy.
- Implements roles-based authorization for admin and cashiers, with admin privileges including order history management and inventory item management.
- Suitable for deployment on servers, allowing multiple users to access the system simultaneously, ideal for supermarket or POS locations.
- Compatible with Windows operating systems.

Notes And Tasks Android App

Programming Language: Java in Android Studio

Database: RoomsDB on SQLite

- Built as a university project using Android Studio and Java.
- Utilizes the RoomsDB library based on SQLite for database management.
- Designed as a versatile notes app allowing users to create, edit, delete, view, and pin notes.
- Features include note searching, task creation, editing, marking tasks as done, and deletion.