ARANI SHANMUKH ABHIRAM

Email: shanmukhabhiram7@gmail.com

OBJECTIVE

A engineer with curiosity and interest in the field of technology in both software and electronics and fast learner with good discipline and clear ideology.

PROFILE

- Had Two Years Experience in Backend Development at Trace Technologies
- Hands-on python
- Known Technologies are #c, c++, java, python, html, css, javascript, sql
- Basic knowledge in docker and apache server
- knowledge in FLASK, React JS, React Native framework
- The ability to analyse complex technical information
- knowledge in working with Arduino uno, Esp32, and Raspberry Pi
- Can analyse, design and implement database and code structures
- Detail oriented
- Excellent problem solver
- Possess good leadership skills
- Good and fast learner

WORK EXPERIENCE

- **Backend Developer**

  Trace Technologies

  February 2022 – March 2025

  Technologies Used: Flask, MySQL, PostgreSQL, MongoDB, Excel Macros

  - Developed and maintained backend systems and APIs, with a focus on database-driven applications.
  - Built admin systems from scratch, including modules for CRUD operations and bulk data processing via Excel templates.
  - Utilized Macros for automated data entry and validation within Excel templates.
  - Worked across multiple databases including MySQL, PostgreSQL, and MongoDB to support scalable, high-performance applications.

- **Freelance Backend & Frontend Developer**

  ByteNode

  December 2023 – Present

  Technologies Used: Flask, React.js, PostgreSQL

  - Worked independently as a full-stack developer, building and maintaining web applications for client projects.

- Developed RESTful APIs using Flask for backend logic and database operations.
- Built dynamic and responsive user interfaces using React.js.
- Integrated PostgreSQL for efficient and secure data storage and retrieval.
- Collaborated closely with clients to gather requirements and deliver custom solutions on time.

EDUCATIONAL BACKGROUND

- 10th SSC, Sri Chaitanya EM School, Suchitra Branch, 2015.
- Inter, Sri Chaitanya Junior College, Kukatpally, Stream: MPC, Graduated Class of 2017
- Sree Venkateswara College Of Engineering, Stream: ECE, Bachelor of technology, 2017-2021

COURSES AND INTERNSHIPS

- Python specialist course in iiec-rise
- Data science using python internship in dhaaps
- AI master class using python course in pantech solutions
- Matlab master class internship in pantech solutions
- Blender course from IIT Bombay
- Android app development course from IIT Bombay
- Master class on embedded systems and IOT using ESP32

SKILL SET

- **Programming Languages**: python, C, basics in java and c++
- **Frameworks Know** : Flask, React js, React native
- **Databases**: MySQL, PostgreSQL
- **Web Technologies** : HTML, CSS, Javascript
- **O/S**: Windows, Linux
- **IDEs Tools**: Visual studio, PyCharm, Jupiter, Collab
- **Webservers** : Apache
- **Cloud Environment** : basics in AWS, Google Cloud

Projects Related To The Work

Project #1 : Admin System

Description:

This was the main project I worked on at Trace Technologies. I built and developed the admin system from the ground up and created many modules for database-related operations. These included APIs for creating, editing, listing, and deleting records. I also implemented bulk APIs that handled data from Excel sheets.

Additionally, I developed the admin system for another company under Trace Technologies, called Safe Air Technologies. For creating templates in Excel sheets, I used macros in some cases to enable bulk data entry. These bulk APIs were designed to first validate the data in the provided Excel templates, and only if the data was correct, they proceeded with the bulk data operations.

I also wrote several Python scripts to automate tasks, manage the admin system, and process raw data efficiently.

Technologies: Flask, Macros, Python

Project #2 : Reports generation Apis

Description:

These Are The Apis That I Had Done At the company, I developed APIs for generating reports related to worker performance and various other key metrics. These APIs also supported functionality for assigning stars and badges to workers based on their performance, enabling a structured and automated evaluation and recognition system.

Technologies: Flask, Python

Project #3 : Alerts System For Sending Emails

Description:

This project was developed to continuously monitor specific database conditions and generate alerts when defined thresholds were exceeded. The alerts were stored in the database, and a separate Python program checked for any unsent alerts and dispatched them via email to designated users.

The alert messages were sent using a well-formatted HTML email template to ensure clarity and professionalism. In Trace Technologies, this system was used to notify users about offline gateways or inactive location tags. In SafeAir Technologies, alerts were triggered when certain pollutant parameter values exceeded predefined thresholds, ensuring timely notifications for critical environmental conditions.

Technologies: Python, SQL

Project #4 : Fullstack Development In Bytenode

Description:

This During my freelancing tenure at ByteNode, I worked on a variety of full-stack development projects, contributing to both backend and frontend components. One of the key projects I handled was the backend development for an electric vehicle (EV) dashboard. Using the Flask framework, I developed APIs that fetched data from the database and provided real-time responses. These APIs delivered parameters such as vehicle speed, battery percentage, last known location, and other telemetry details essential for monitoring EV performance.

Another major project I worked on was a comprehensive water meter management system, where I was responsible for both backend and frontend development. On the backend, I created robust APIs for managing water meter data, including endpoints for creating, editing, listing, and deleting information related to meters and their associated tenants. I also implemented APIs for generating reports and calculating water consumption, either over a custom date range or for a specific month. The system included functionality for generating billing reports in PDF format and returning them as part of the API response.

On the frontend, I developed the user interface using React.js to ensure that all backend data could be easily viewed and managed. This included dashboards, forms for data entry and updates, and interactive views for reports and billing details. The system was designed to provide a seamless experience for users managing water meters and consumption data across multiple modules.

Technologies: Flask, Python, React Js

Project #5 : Iot Project For The Irriagation Motors In Bytenode

Description:

This As part of my freelancing work with ByteNode, I developed an IoT-based irrigation motor management system for agriculture. In this project, I was responsible for both the hardware and software components. I used an ESP32 microcontroller to manage connectivity and control logic. Initially, if Wi-Fi credentials were not pre-configured, the ESP32 created its own Wi-Fi hotspot. Users could connect to this hotspot, access a local IP configuration page, and input their Wi-Fi name and password. Upon submission, the ESP32 would disconnect from the hotspot and connect to the specified Wi-Fi network.

After establishing the Wi-Fi connection, the ESP32 connected to an MQTT server and subscribed to relevant topics. Based on the messages received such as "ON" or "OFF" the system would control the irrigation motor accordingly. To switch the motor on and off, I integrated a Solid State Relay (SSR) with appropriate heat dissipation using a heat sink and thermal paste. The relay's control pins were connected to the ESP32, and a physical switch was added to reset the ESP32 and clear saved Wi-Fi credentials, allowing reconfiguration when needed.

For the hardware enclosure, I designed and assembled a power socket box with a built-in switch, connected to the SSR's output side. This allowed for safe and reliable motor control, making the system a robust and user-friendly solution for remote irrigation management.

Technologies: Micro Python

Personal Projects

Project #1 : Voice assistant

Description:

This was a prototype for a voice assistant, similar to Google Assistant, developed using Python. The modules used included Google Text-to-Speech, Google Translator, os, playsound, time, BeautifulSoup, requests, and Selenium. The system was capable of recognizing various languages available in Google Translate and converting them to English for processing.

The prototype could execute several voice-based instructions, such as providing weather information through web scraping using Selenium. It could also fetch and deliver detailed or brief information from Wikipedia, which it would then read aloud using the Google Text-to-Speech module.

Technologies: Python

Project #2 : Object following and obstacle avoiding robot by using image processing

Description:

The Object Following and Obstacle Avoiding Robot was our final year university project, developed using Python. The hardware components used included a robot chassis, Raspberry Pi B+, two lithium ion rechargeable batteries, a motor driver, PiCamera, ultrasonic sensor, and a power bank. The Python modules used in the software were picamera, OpenCV, time, numpy, and gpiozero.

The robot captures frames using the camera, converts them into a masked view based on a specific color in our case, yellow and then calculates the position of the object using its X and Y coordinates relative to the center of the frame. Based on these coordinates, the robot adjusts the power to its left or right wheels to follow the object. If the object goes out of frame or its detected area falls below a certain threshold, the robot rotates in place to search for it.

To handle obstacles, the robot uses an ultrasonic sensor to detect and avoid up to three obstacles that may block its path just before executing obstacle-avoidance commands. If an obstacle appears during these commands, the robot stops until the obstacle is cleared. A VNC server was used to remotely control the Raspberry Pi from our device.

My role:

- My role was to write the algorithm and program for the obstacle avoidance and setting up the raspberry pi
- And I was also involved in the object following code.

Technologies: Python, raspbian os, vnc server

Project #3 : Chatting Web app by using FLASK Framework and SQL

Description:

This project was developed using the Flask framework with Python, MySQL, and SQL. In this application, a user must first log in with their name, after which they are redirected to a chat page where they can interact with other users. All chat messages are stored in a MySQL database.

To ensure session security, the application always redirects unauthenticated users back to the login page if they try to access the chat page directly. A logout option is available in the navigation bar to allow users to end their session manually. I used HTML and CSS to design the front-end interface.

Once logged in, the session remains active for up to 25 days, even if the browser is closed. This project represents a basic yet functional implementation of a Flask-based web application.

Technologies: FLASK Framework, MYSQL, HTML, CSS, Python, SQL

Project #4 : Gender Detection by using AI

Description:

This project uses Artificial Intelligence algorithms to detect a person's gender based on facial recognition. It was developed using Python, with the following modules: TensorFlow, scikit-learn, NumPy, Matplotlib, random, OpenCV, and os.

The codebase is divided into two main files: one for training the model and another for using the trained model to perform gender detection via a webcam. The dataset consists of 2,300 images of men and women, organized into two separate folders. The training process is conducted over 100 epochs using batch training to handle the large dataset efficiently. Once trained, the model is saved in a dedicated folder.

The training script also generates accuracy and loss charts using Matplotlib, allowing visualization of the model's performance. The second file loads the trained model and uses OpenCV to detect a face from the webcam feed. The detected face is cropped and then passed to the model, which predicts the gender based on the trained data.

Technologies: Python

Project #5 : Encrypted connection to send server or computer details by using sockets

Description:

This project was developed to remotely monitor and transmit system information such as RAM usage, CPU usage, GPU usage, system uptime, available storage, operating system, and system architecture. The data transmission is secured through encryption to ensure safe communication between systems.

The project utilizes several Python libraries, including platform, time, datetime, psutil, subprocess, shlex, requests, ctypes, threading, and socket. The program runs continuously, collecting and sending system data every five minutes to a client application running on a separate device. On the client side, the received data is decrypted and displayed, providing real-time system monitoring capabilities.

Technologies: Python

Extra-Curricular Activities / Achievements:

- Runners in school Kho-Kho tournament.
- 1st Prize in school essay writing Competition.

I hereby declare that the above written particulars are the best of my knowledge and belief.

Place : Hyderabad **ARANI SHANMUKH ABHIRAM**