Shreyasri R

7200405575 | ramkishreya@gmail.com | Chennai, Tamil Nadu, India 600059

SUMMARY

A passionate and proactive second-year Bachelor of Engineering student specializing in Computer Science and Engineering, with a strong foundational knowledge of programming languages including C, Java, and Python. Enthusiastic about deepening my expertise in software development, algorithms, and data structures while continuously improving my coding proficiency. Eager to contribute to real-world projects and leverage my problem-solving abilities to drive innovative solutions. Actively seeking internship opportunities to apply my technical skills, gain hands-on experience, and grow within the dynamic tech industry.

CAREER OBJECTIVES

- *Strengthen and refine programming skills while gaining hands-on experience in software development through practical projects and professional exposure.*
- *Contribute to innovative advancements in artificial intelligence, machine learning,and software engineering.*
- *Pursue a successful career in the technology sector, focusing on continuous growth in software design, system development, and emerging technologies.*
- *Thrive in dynamic and challenging environments, leveraging technical expertise tocreate impactful, forward-thinking solutions.*

WORK EXPERIENCE

Product Development Trainee January 2025 – February 2025

Transform Automations India P Limited, Bengaluru

- Collaborated on product development initiatives, contributing to analysis, design, and implementation of features with a focus on maintainability and reusability.
- Developed and tested source code adhering to established design and coding standards.
- Estimated task efforts proactively communicated challenges and adhered to project deadlines.
- Worked remotely, maintaining regular progress updates, and participating in feedback sessions with mentors and team members.
- Ensured confidentiality and data security in compliance with organizational guidelines.

EDUCATION

BACHELOR OF ENGINEERING (COMPUTER SCIENCE AND ENGINEERING)

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY, CHENNAI

2023 – 2027, CGPA: 8.9 (AS OF SEMESTER IV)

12TH GRADE (MATHEMATICS & SCIENCE)

SOPHIA HIGH SCHOOL, BANGALORE

MAY 2023, BOARD: ISC, PERCENTAGE: 88%

10TH GRADE

SRI CHAITANYA TECHNO SCHOOL, BANGALORE

MAY 2021, BOARD: CBSE, PERCENTAGE: 88%

PROGRAMMING/TECHNICAL SKILLS

- ***C:*** *Strong foundation in data types, loops, functions, and pointers. Used for problem-solving, algorithm development, and memory management in academic projects with GCC.*
- ***Python****: Proficient in OOP, exception handling, list comprehensions, and libraries like NumPy and Pandas. Experienced in automation, data manipulation, and basic machine learning with scikit-learn.*

- ***Java:*** *Knowledgeable in Java syntax, OOP principles, exception handling, and multithreading. Developed console and desktop applications (GUI with Swing) in academic projects.*
- ***Algorithms & Data Structures:*** *Skilled in sorting, searching, and implementing linear (arrays, linked lists) and non-linear (trees, graphs) data structures.*
- ***MySQL:*** *Experienced in database design, query writing, data manipulation, normalization, indexing, and performance optimization. Familiar with PHP and Python integration*
- ***Machine Learning:*** *Studied core ML concepts including supervised learning (Linear & Logistic Regression, Decision Trees, SVM, KNN) and unsupervised learning (K-Means, Hierarchical Clustering, EM algorithm). Explored ensemble methods like Bagging, Boosting, and fundamentals of Reinforcement Learning (Q-learning, SARSA). Also covered dimensionality reduction techniques (PCA, ICA), probabilistic models (Bayesian Networks, Hidden Markov Models), and model evaluation metrics (confusion matrix, precision, recall, F1-score, ROC).*
- ***Data and Information Science:*** *Focused on full-cycle data handling: data acquisition (APIs, web scraping), pre-processing (cleaning, wrangling, imputation), and visualization using tools like Tableau and Python (Matplotlib, Pandas). Worked with EDA, text mining (BoW, regular expressions, tokenization), feature engineering, and explored Big Data concepts (Volume, Velocity, Variety).*
- ***Data Warehousing****: Proficient in Snowflake for data warehousing, query optimization, pipeline automation, and secure data governance with BI tool integration. Used Snowflake Connector, Snowflake Drivers, Snowflake REST API, SnowSQL.*
- ***JavaScript:*** *Proficient in JavaScript, ES6+, and asynchronous programming.*
- ***Frameworks & Libraries:*** *Django, React.js, Redux, Bootstrap*
- ***Databases****: PostgreSQL, SQLite, MySQL*
- ***Version Control****: Git, GitHub*
- ***Mobile Development*** *Android Studio, Kotlin, REST API, and AWS Lamba and DynamoDB.*

SOFT SKILLS:

• Strong communication skills (verbal and written), with the ability to express ideas clearly and collaborate

• Excellent problem-solving and critical thinking abilities, applying creative solutions to complex challenges.

• Proven time management and organizational skills, balancing multiple tasks efficiently.

• Highly adaptable and resilient, able to adjust quickly to new situations and work effectively under pressure.

• Skilled in conflict resolution, building rapport, and maintaining positive relationships with colleague.

CERTIFICATIONS AND BADGES

- **Python Institute & Cisco Networking Academy –** *Issued Nov 2024*
  - *Statement of Achievement – Python Essentials 1*
- **Cisco Networking Academy – Networking Basics –** *Issued Nov 2024*
  - *Badge and certificate on the Networking Basics course and achieved this student level credential.*
- **Oracle Java Foundation Badge from Oracle -** *Issued Nov 2024*
  - *Badge and certificate on the Java programming and object-oriented capabilities, hands-on experience with the NetBeans IDE and Oracle Cloud.*
- **Cisco Networking Academy - Data Science -** *Issued Nov 2024*
  - *Badge and certificate on the Introduction to Data Science course and achieved this student level credential.*

- **Great Learning Academy– *JavaScript*** - *Issued Nov 2024*
    - *Introduction to JavaScript*
- **Oracle Cloud Infrastructure 2023 AI Certified Foundations Associate**
    - *Issued Aug 2024*
    - *AI services, cloud computing, and OCI for building AI-driven apps.*
- **Great Learning Academy–** *Frontend Development CSS*
        - *Issued Nov 2024*
- **Programming in Java from IIT Kharagpur- NPTEL–** *Issued Oct 2024*
    - *Data structures and algorithms and how the same can be implemented using* Java programming *language.*
- **Digital 101 – 30 Hours from NASSCOM -** *Issued June 2024*
    - *AI tools, digital communication, and other advanced professional skills*
- **Linear Algebra with MATLAB from MathWorks–**
    - *Issued April 2024*
- **Calculations with Vectors and Matrices with MATLAB from MathWorks** –
    - *Issued Oct 2024*
- **Certificate of Merit for heading BIONOVA -** *Issued 2022*
    - *Awarded by Sophia High School for leading & contributing to the project*

PROFESSIONAL PROJECTS

PROJECT: PRODUCT DEVELOPMENT - ASSUREAI

Transform Automations India P Limited

Role: Product Development Trainee

Duration: January 2025 – February 2025

Project Description: Contributed to the development of a product under the company's first project initiative. Focused on creating scalable and maintainable modules as part of a broader product development effort.

Responsibilities:

Requirement Analysis: Reviewed and understood module/feature specifications to ensure alignment with project goals.

Design and Development: Designed and developed features/modules emphasizing maintainability and reusability.

Code Standards and Testing: Wrote, reviewed, and tested source code, ensuring adherence to design and coding standards.

Collaboration: Worked closely with the mentor and team members in remote setups, participated in regular catch-ups for progress updates and feedback.

Project Management: Estimated effort for tasks, maintained timelines, and proactively communicated potential challenges.

Key Outcomes:

Delivered a functional module within the agreed timelines.

Ensured compliance with data security and confidentiality protocols.

Received continuous feedback from the mentor, leading to personal and professional growth.

TECHNICAL PERSONAL PROJECTS (Tools or Small Products)

PROJECT NAME: SUBSCRIPTION MANAGEMENT SYSTEM

*The Subscription Management System is a full-stack web application designed to manage customer subscriptions to various products. Built with **Django** for the backend and **React** for the frontend, this system allows users to add, extend, and end subscriptions while also generating a real-time revenue report.*

Key Features

Customer and Product Management

Add, update, and list customers and products.

Dropdowns dynamically load customers and products from the database.

Subscription Management

Add Subscription: Subscribe customers to products with specified start and end dates and the number of users.

Extend Subscription: Extend the subscription period by updating the end date.

End Subscription: Mark a subscription as ended by setting the end date to the current date.

Revenue Report: Calculate and display the total revenue based on active subscriptions, considering the number of users and product costs.

- *Technologies Used*

Backend: Django (Python)

Frontend: React (JavaScript)

Database: SQLite (development)

HTTP Client: Axios

Styling: CSS

PROJECT NAME: SHYNTAX - GST INVOICE & TAX MANAGEMENT DASHBOARD

***ShynTax** is an all-in-one GST invoice and tax management solution designed to simplify the way businesses handle invoices, clients, and taxes. Whether you're managing GST rates, generating invoices, or tracking tax payments, ShynTax provides a seamless experience with an intuitive and user-friendly interface.*

Key Features

Comprehensive Invoice Management

Create, manage, and track invoices with ease, streamlining the invoicing process for businesses.

Client Management

Add and manage clients, linking them directly to their respective invoices for efficient tracking.

GST Compliance

Automatically calculates CGST, SGST, and IGST based on the invoice details, ensuring full GST compliance for every transaction.

Analytics & Reporting

Provides detailed analytics to view invoice trends, unpaid invoices, and overdue payments to help businesses stay on top of their finances.

Easy Export Options

Download invoices and tax reports in PDF and Excel formats for quick reference, record-keeping, or filing.

Tax Payment Tracking

Track GST payments and manage tax payments against each client's invoice to ensure accurate records.

Customizable Design

A modern, professional dashboard with a minimalistic design focused on usability, making it easy for users to navigate and manage their GST tasks.

ShynTax is the perfect tool for businesses looking to streamline their GST and tax processes efficiently, with an emphasis on accuracy, compliance, and user convenience.

Technologies Used

Backend: Flask (Python), SQLAlchemy (ORM for database management), Jinja2 (templating engine)

Frontend: HTML5, CSS3, JavaScript, Bootstrap (responsive design)

Database: SQLite (development), SQLAlchemy (database management)

Analytics: Isolation Forest (for anomaly detection), Pandas (data processing)

Export: XHTML2PDF (for PDF export), Pandas (for Excel export)

Security: Flask session management, secret key for secure data handling

PROJECT NAME: SHYNARCH - FLOORPLAN GENERATOR PROJECT

Developed a dynamic and user-friendly FloorPlan Generator web application using Django (Backend) and React.js (Frontend). The tool automates the creation of customizable house floor plans based on user inputs such as area (sq. ft), dimensions, and room requirements, providing both 2D and 3D visualizations.

Key Features:

Automated Floor Plan Generation: Generates floor plans dynamically based on user-defined parameters.

Customizable Layouts: Drag-and-drop functionality to modify walls, doors, and windows.

3D Visualization: Integrated real-time 3D rendering for better spatial understanding.

Cost Estimation Tool: Provides approximate construction costs based on materials and specifications.

Room-Specific Configuration: Allows dimension adjustment and room-specific preferences.

Export & Print Options: Blueprint export in PDF format for easy sharing.

Technologies Used:

Backend: Django REST Framework for API development and data handling.

Frontend: React.js for building an interactive and responsive user interface.

Database: PostgreSQL for managing user data and project details.

3D Rendering: Integrated third-party libraries for real-time 3D visualization.

Impact: Reduced the time required to create preliminary architectural designs for users. Enhanced user experience with a highly interactive interface and visualization tools. Enabled non-technical users to create professional-grade floor plans effortlessly. This project demonstrates expertise in full-stack development, API integration, and user-centric design.

ACADEMIC PROJECTS

PROJECT NAME: PLAYLIST NAVIGATION SYSTEM IN PYTHON

This project simulates a playlist navigation system like Spotify, where users can navigate through songs in both forward and backward directions. The program implements a doubly linked list data structure to represent the playlist, allowing for efficient insertion, deletion, and traversal of songs.

Key Features: *Doubly Linked List Structure:*

The playlist is represented as a doubly linked list, where each song is a node that contains Song details (title, artist, duration.) Pointers to the next and previous songs in the playlist

- *Navigation: Users can Play the next song in the playlist (move forward).*

  Play the previous song (move backward).

Insertion & Deletion Operations:

Insert Song: Add new songs to the playlist at the beginning, end, or any specific position. Delete Song: Remove songs from the playlist based on their position or title.

Real-Time Application of Data Structures:

This project demonstrates the use of doubly linked lists in managing dynamic data (songs) where users can perform operations like insertion, deletion, and navigation in both directions.

- *Technologies Used: Python - Basic File I/O (Optional for Saving Playlist Data)*
- *Use Case: Ideal for applications like music players where playlists need to be navigated in both directions (next and previous). Allows easy addition/removal of songs, and efficient traversal of the playlist.*

PROJECT NAME: TICKET COUNTER QUEUE MANAGEMENT SYSTEM FOR SKY CINEMAS

This project simulates a queue management system for handling customer ticket requests at the SKY Cinemas ticket counter. The system is implemented using the queue data structure, where customers are served in a first-come, first-served (FCFS) manner. The queue ensures efficient management of the ticket counter and smooth processing of customer requests.

- ***Key Features:*** *Queue Data Structure:*

  The system is built using a queue, where:

  Enqueue operation adds customers to the end of the queue (i.e., new customers waiting to buy tickets).

  Dequeue operation serves customers from the front of the queue (i.e., ticket counter serves customers in the order they arrived).

  Peek operation allows checking the customer currently at the front of the queue without removing them.

- *Real-Time Application of Queue Operations:*

  This project effectively models a real-world scenario where customers are queued up for services at a cinema counter. The queue ensures that the system processes customer requests in the order they arrive, avoiding confusion or disorder.

- *Technologies Used: Python*

  Queue Data Structure (Implemented using list or custom class)

  Basic File I/O (Optional, for storing customer details or ticket information)

- *Use Case: This system can be used in ticket counters at cinemas, banks, or any service-oriented businesses where customers need to be served in the order of their arrival.*

  It models the First-Come, First-Served (FCFS) principle, which is commonly used in queues to ensure fairness.

ACADEMIC PROJECT: SALES DATA ANALYTICS PLATFORM

This project designs a scalable data pipeline and analytics platform using Snowflake for processing and analyzing large sales datasets.

Key features:

Developed a data warehousing solution using Snowflake to handle structured and semi-structured data (CSV and JSON formats).

Configured and optimized Snowflake's virtual warehouses for efficient querying and storage.

Built a data ingestion pipeline using Python to automate data loading from local and cloud sources into Snowflake.

Wrote SQL queries to analyze sales performance, customer behavior, and product trends.

Implemented role-based access control in Snowflake to ensure data security and privacy.

Integrated Snowflake with Tableau to create interactive dashboards for real-time business insights.

Enhanced query performance by leveraging Snowflake features like clustering and query caching, reducing analysis time by 30%.

- *Key Technologies Used: Snowflake, Python, SQL, Tableau, AWS S3*
- *Outcome: Delivered a scalable and efficient data analytics platform, enabling comprehensive insights into sales trends and decision-making processes.*

Project Name: Complex Expression Conversion to Machine-Readable Form

This project focuses on taking a complex mathematical expression from the user, formatted in a human readable BODMAS (Brackets, Order, Division, Multiplication, Addition, Subtraction) notation, and converting it into a machine-readable form that can be processed by a computer for evaluation. The goal is to parse and convert an infix expression (common mathematical notation) into either a postfix (Reverse Polish Notation - RPN) or prefix notation, which can be efficiently interpreted and calculated by a machine.

Key Features:

Human-Readable to Machine-Readable Conversion:

*The program accepts a complex mathematical expression as input in standard BODMAS format, such as 3 + (2 * 5) - 7. It then converts it into postfix (or prefix) notation, which is a more machine-friendly format without the need for parentheses to enforce operator precedence.*

- *Infix to Postfix Conversion: The core of the project involves converting an infix expression (like 3 + (2 * 5)) to postfix notation (like 3 2 5 * +). This is done using the Shunting Yard algorithm.*
- *Error Handling: The program includes basic error handling to check for invalid inputs, such as mismatched parentheses or invalid characters, and ensures the expression is valid before processing it.*
- *Operator Precedence & Associativity: The program accounts for operator precedence (multiplication before addition) and associativity (left-to-right for most operators, right-to-left for exponentiation), ensuring accurate conversion and evaluation.*
- *Technologies Used: Python*

  Stacks for managing operators and operands during conversion BODMAS rules for respecting the order of operations. Shunting Yard Algorithm for converting infix to postfix/prefix notation.
- *Use Case: This project demonstrates how to handle and process complex mathematical expressions in a form that a computer can evaluate efficiently.*

  It's useful in scenarios where expressions need to be evaluated programmatically, such as in scientific calculators, compilers, or automated mathematical tools.

Project Name: Sorting Phone Contacts by First Name Using Insertion Sort

This project implements an insertion sort algorithm to sort a list of phone contact names in ascending order based on the first name. The contacts are represented as a list of dictionaries or tuples, where each entry contains a first name and last name. The program uses the insertion sort algorithm to rearrange the contacts in lexicographical order (alphabetical order) based on the first name.

Key Features: *Contact List Representation:*

The contacts are stored as a list of dictionaries (or tuples), where each dictionary contains keys like first_name and last_name.

Insertion Sort Algorithm: The insertion sort algorithm is used to sort the contacts by their first name. This algorithm works by iterating over the list, comparing each element with the previous ones, and inserting it into the correct position in the sorted portion of the list.

Sorting Process: The program sorts the list in ascending order based on the first name. If two contacts have the same first name, they remain in the order they appeared in the original list.

Display Sorted Contacts: After sorting, the program displays the contacts in the sorted order.

- *Technologies Used: Python*

  Insertion Sort Algorithm for sorting data Basic Data Structures (List, Dictionary)

- *Use Case: This project simulates the process of sorting contact names in a phone book or address book application. Sorting contacts by first name can be useful for quick lookups and better organization of contacts. The insertion sort algorithm was chosen for its simplicity and efficiency in this scenario, where the contact list is not excessively large.*