Python Developer

INDUSTRIAL TRAINING REPORT

Submitted in partial fulfillment of the requirements for the award of

degree of

Bachelor of Technology
In
Computer Science and Engineering

Submitted To: -
Er. Rajdavinder Singh Boparai

Submitted By: -
Pradeep Kumar
2121971

I.K. GUJRAL PUNJAB TECHNICAL UNIVERSITY,
JALANDHAR

October 2023

DECLARATION

I, Pradeep kumar, 2121971 student of B.Tech (CSE-Data Science) declare that I have undergone Industrial Training from duration 07/01/2023 to 31/07/2023. in “Asterisc Technocrat Pvt. Ltd” under the guidance of “Mr. Chandrakant ” on Technology “Python-developer”. It is for partial fulfillment of requirements for the Bachelor of Technology in Computer Science Engineering degree, Gulzar Group of Institutions, Khanna, Ludhiana.

Candidate Signature **Signature of HoD**

Date of Submission: October 9, 2023

ACKNOWLEDGEMENT

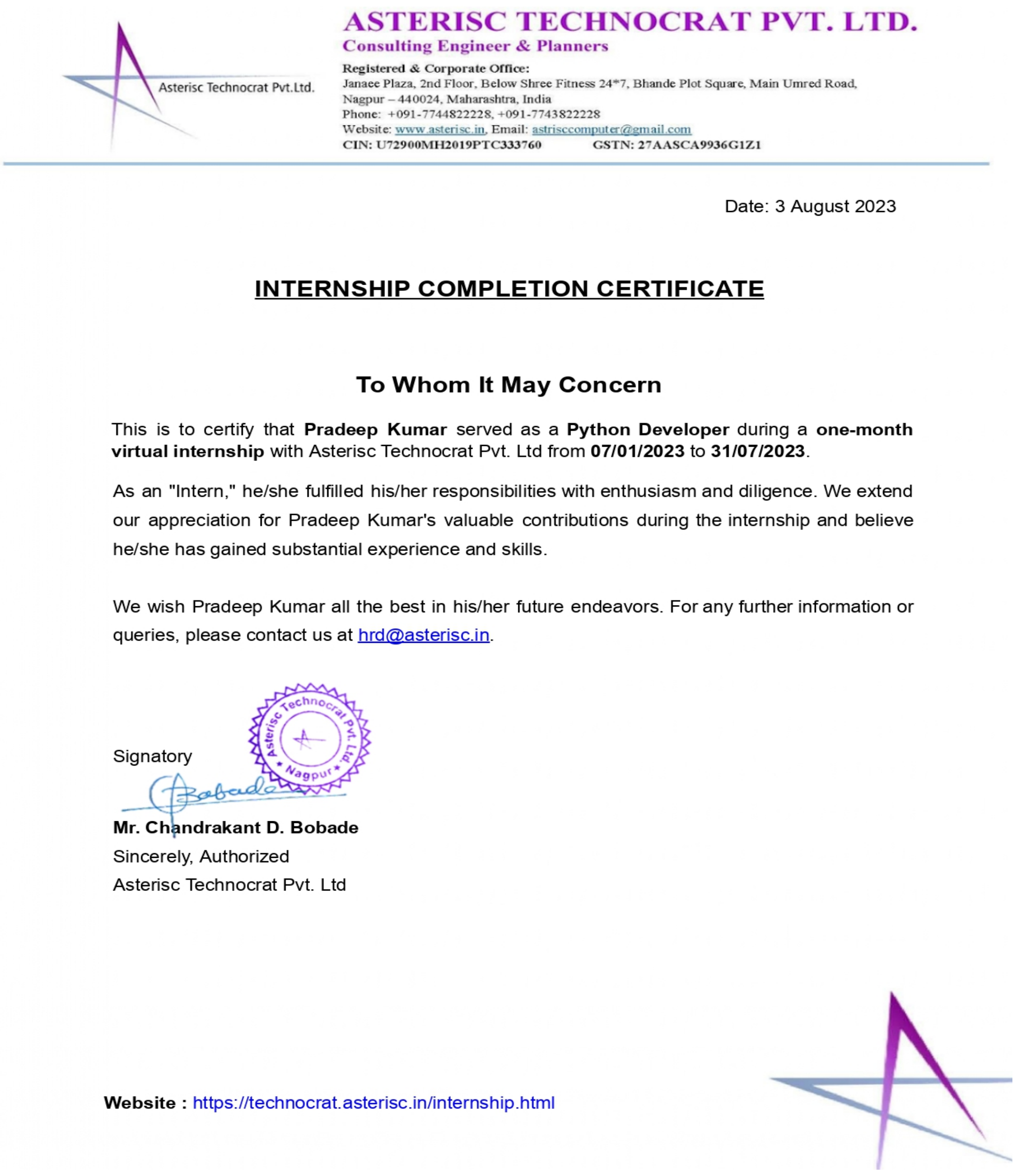
Asterisc Technocrat is a well-respected custom software development company in Nagpur, India. We are ISO 9001:2015 certified and have a high rating on Google reviews. We are committed to providing our customers with the best possible IT solutions and services.

Our team comprises dedicated and passionate individuals who are experts in their field. We have a deep understanding of business challenges and the latest technologies. This allows us to create custom software solutions that help our customers achieve their business goals.

We also care deeply about our employees and our clients. We believe that they are the most important part of our company. We foster a collaborative and supportive work environment where everyone has the opportunity to grow and succeed. We also value work-life balance and make every effort to create a positive and productive work environment for our employees.

We are grateful for the opportunity to serve our customers and to partner with them on their journey to digital transformation. We are committed to providing them with the best possible service and support.

CERTIFICATE



Asterisc Technocrat Pvt.Ltd.

ASTERISC TECHNOCRAT PVT. LTD.
Consulting Engineer & Planners
Registered & Corporate Office:
Janaee Plaza, 2nd Floor, Below Shree Fitness 24*7, Bhande Plot Square, Main Umred Road,
Nagpur – 440024, Maharashtra, India
Phone: +091-7744822228, +091-7743822228
Website: www.asterisc.in, Email: astrisccomputer@gmail.com
CIN: U72900MH2019PTC333760 **GSTN: 27AASCA9936G1Z1**

Date: 3 August 2023

INTERNSHIP COMPLETION CERTIFICATE

To Whom It May Concern

This is to certify that **Pradeep Kumar** served as a **Python Developer** during a **one-month virtual internship** with Asterisc Technocrat Pvt. Ltd from **07/01/2023** to **31/07/2023**.

As an "Intern," he/she fulfilled his/her responsibilities with enthusiasm and diligence. We extend our appreciation for Pradeep Kumar's valuable contributions during the internship and believe he/she has gained substantial experience and skills.

We wish Pradeep Kumar all the best in his/her future endeavors. For any further information or queries, please contact us at hrd@asterisc.in.

Asterisc Technocrat Pvt. Ltd. * Nagpur *

Signatory

Bobade

Mr. Chandrakant D. Bobade
Sincerely, Authorized
Asterisc Technocrat Pvt. Ltd

Website : https://technocrat.asterisc.in/internship.html

CONTENTS

1. INTRODUCTION

a) Organization Profile Overview:

Asterisc Technocrat an ISO 9001: 2015 Internationally Certified & Central India's high rating for Google reviews & most trusted company in Nagpur, We are a superb custom software development company in Nagpur delivering impressive business IT Solutions and related services to customers across the world. Our development services are led by our dedicated & passionate team members to provide the best industry executions combined with latest technology expertise and business domain knowledge to drive digital transformation. Our expertise, ability to understand business challenges and professional competence allows us to create a better experience for our customers. At Asterisc Technocrat, we care for our employees and our clients, they are the most important part in our company. We cuddle clan strategic culture where every individual gets an equal opportunity of growth & great exposure. We value our team and provide work-life balance and make their career progressive and admirable.
Mission and Values:

Asterisc Technocrat Pvt. Ltd is a company that helps young people learn and grow in their careers. They offer practical experience, mentorship, and a good environment to work in. The company's core values are:

- Nurturing skills: Helping people learn and grow their skills.
- Empowering young talents: Giving young people the opportunity to learn and grow in their careers.
- Providing a conducive environment: Creating a good place to work where people can learn and grow.
- Offering practical exposure: Giving people the opportunity to learn from real-world experience.
- Mentorship: Providing people with guidance and support from more experienced people.
- Learning and Development:

Asterisc Technocrat is a company that wants their employees to learn and grow in their careers. They offer their employees a variety of opportunities to learn new skills, such as training programs, mentorship, and self-study. Asterisc Technocrat also gives their employees the opportunity to work on challenging and innovative projects. This helps Asterisc Technocrat to attract and retain top talent, and to create a more productive and innovative workforce.

Collaboration:

Asterisc Technocrat is a company that values collaboration. They believe that their employees can achieve more by working together. Asterisc Technocrat fosters a collaborative environment by encouraging peer interaction, offering networking opportunities, and creating a sense of community.
Internship Programs:

Asterisc Technocrat offers internship programs for students and young professionals who want to learn more about being a Python developer. These programs give you the chance to get hands-on experience in the following areas:

Python developer: You will learn how to write Python code and use Python libraries to solve real-world problems.
Software development: You will learn about the software development process and how to build and test software applications.

- Web development: You will learn how to build and deploy web applications using Python frameworks such as Django and Flask.
- Data science: You will learn how to use Python to collect, analyze, and visualize data.

Asterisc Technocrats internship programs are a great way to learn about the field of Python development and to gain the skills you need to start a career in this field.
Impact and Recognition:

Asterisc Technocrat is a company that is dedicated to helping young people learn and grow in their careers. They believe that it is important to bridge the gap between academia and industry. Asterisc Technocrat has been recognized for its work in this area, and its alumni have gone on to be successful in their careers.

b) Introduction of Project

Quiz Application PROJECT

This project is about creating a software application that allows users to create and take quizzes. The application can be used for educational purposes, such as testing students' knowledge on a particular subject, or for entertainment purposes, such as creating quizzes for friends and family.

Daily Task Scheduler PROJECT

This project is about creating a software application that helps users organize and manage their daily tasks. Individuals or teams can use the application to track their progress on tasks, set deadlines, and receive reminders.

CAFE BILLING APPLICATION PROJECT

This project is about creating a software application that helps cafes and restaurants streamline their billing and order management process. The application can be used to take customer orders, calculate bills, and manage customer transactions.

All three of these projects are designed to solve real-world problems. The Quiz Application project can help students learn more effectively, the Daily Task Scheduler project can help people be more productive, and the CAFE BILLING APPLICATION project can help cafes and restaurants run more efficiently.

c) Problem Specification

Quiz Application

- Users need a way to create and take quizzes.
- Users need a way to assess their knowledge and receive instant feedback.
- Users need a way to track their quiz performance.

Daily Task Scheduler

- Users need a way to organize and manage their daily tasks efficiently.
- Users need a way to analyze data to identify productivity patterns, optimize time management, and improve task prioritization.
- Users need a way to view and manage their tasks alongside other appointments and commitments.

CAFE BILLING APPLICATION

- Cafe staff need a way to take customer orders efficiently and accurately.
- Cafe staff need a way to calculate bills and manage customer transactions.
- Cafe administrators need a way to manage the menu items offered by the cafe.

d) Problem Definition

Quiz Application project:

- Problem: Students need a way to effectively test their knowledge and receive instant feedback.
- Solution: Develop a quiz application that allows users to create and take quizzes on a variety of topics.

Daily Task Scheduler project:

- Problem: Individuals and teams need a way to organize and manage their daily tasks efficiently.
- Solution: Develop a daily task scheduler application that allows users to create, manage, and track their tasks.

CAFE BILLING APPLICATION PROJECT:

- Problem: Cafe staff and administrators need a way to streamline the billing and order management process.
- Solution: Develop a cafe billing application that allows cafe staff to take customer orders, calculate bills, and manage customer transactions.

The solutions are:

1. Quiz Application: Develop a quiz application that allows users to create and take quizzes.
2. Daily Task Scheduler: Develop a daily task scheduler application that allows users to create, manage, and track their tasks.
3. CAFE BILLING APPLICATION: Develop a cafe billing application that allows cafe staff to take customer orders, calculate bills, and manage customer transactions.

e) Objective of Project

Quiz Application:

- To develop a software application that allows users to create and take quizzes on a variety of topics.
- To provide a platform for users to assess their knowledge and receive instant feedback.
- To help students learn more effectively by providing them with a fun and engaging way to practice their skills.

Daily Task Scheduler:

- To develop a software application that helps individuals and teams organize and manage their daily tasks efficiently.
- To provide users with a way to analyze data to identify productivity patterns, optimize time management, and improve task prioritization.
- To help users reduce stress and anxiety by providing them with a way to stay on top of their tasks and avoid feeling overwhelmed.

CAFE BILLING APPLICATION:

- To develop a software application that helps cafes and restaurants streamline their billing and order management process.
- To improve the efficiency and accuracy of order-taking and billing.
- To reduce errors and improve the customer experience.
- To provide cafe administrators with insights into their business performance.

Concepts that are used in code

Quiz Application:

1. Programming Language: The entire program is written in Python, a high-level, interpreted programming language.

2. Object-Oriented Programming (OOP): The code follows the principles of Object-Oriented Programming, as evident from the definition of the User, Quiz, and Question classes.

3. Class Definition: Three classes are defined in the program: User, Quiz, and Question. Each class encapsulates related data and functionality.

4. Class Initialization: All classes (User, Quiz, and Question) have an __init__ method for initialization. Note that there is a typo in the code; the method should be named __init__ instead of init.

5. User Input Handling: The input function is used to get user input for various tasks, such as user registration, login, quiz creation, and answering quiz questions.

6. List Handling: Lists (questions in the Quiz class) are used to store collections of related data.

7. String Manipulation: Various string manipulation techniques are used, such as splitting a comma-separated list of options and answers.

8. User Registration and Authentication: The register function collects user information (name, email, password) and creates a User object. The login function takes user input for email and password, simulating an authentication process.

9. Quiz Creation: The create_quiz function allows the user to create a quiz by inputting the quiz title, duration, number of questions, and details for each question.

10. Quiz Taking: The take_quiz function presents each question to the user, collects their answers, and calculates the final score.

11. Looping Structure: The program includes a sequence of interactions within a script, rather than a persistent application with external data storage.

12. Type Conversion: Type conversion functions (int()) are used to convert user input into appropriate data types, such as converting the duration, the number of questions, and the user's answer choice to integers.

13. Conditional Statements: Conditional statements (if, else) are used to determine whether a user's answer is correct and to handle different scenarios based on user input.

14. User Interaction: The program interacts with the user by displaying prompts, collecting input, and providing feedback (e.g., registration successful, login successful, quiz created).

15. String Formatting: String formatting is used to present questions and options in a readable format.

16. Error Handling: The program lacks explicit error handling, making it susceptible to potential issues if users provide unexpected inputs.

Daily Task Scheduler:

1. Programming Language: The entire program is written in Python, a high-level, interpreted programming language.

2. Object-Oriented Programming (OOP): The code follows the principles of Object-Oriented Programming, as evident from the definition of the Task class.

3. Class Definition: The program defines two classes: Task and TaskScheduler. The Task class represents an individual task, and the task scheduler class manages a collection of tasks.

4. Class Initialization: Both classes (Task and TaskScheduler) have an __init__ method for initialization.

5. List Handling: The TaskScheduler class uses a list (tasks) to store instances of the Task class.

6. User Input Handling: The input function is used to get user input for various tasks, such as adding, updating, or deleting tasks.

7. Menu System: A menu system is implemented using a string (menu) to display options to the user. The user is prompted to enter a choice, and the program responds accordingly.

8. Looping Structure: The program uses a while True loop to continuously display the menu and handle user choices until the user decides to quit.

9. Task Operations: Methods in the TaskScheduler class (add_task, update_task, delete_task, view_tasks) provide functionality for adding, updating, deleting, and viewing tasks.

10. Helper Function: The program includes a helper function (get_user_input) to get user input by displaying a prompt and stripping leading/trailing spaces.

11. Instance Creation: An instance of the TaskScheduler class (scheduler) is created to manage tasks.

12. Input Validation: The program includes basic input validation to ensure that user inputs, such as task indices, are within appropriate ranges.

13. String Formatting: The enumerate function is used for string formatting when displaying tasks, providing a task number along with the task title.

14. User Interaction: The program interacts with the user by displaying a menu, receiving user input, and providing feedback on the actions performed (e.g., task added successfully).

15. Loop Exit Mechanism: The program provides an option for the user to exit the loop and terminate the program (choice == "5").

CAFE BILLING APPLICATION :

1. Programming Language: The entire program is written in Python, a high-level, interpreted programming language.

2. Object-Oriented Programming (OOP): The code follows the principles of Object-Oriented Programming, as evident from the definition of classes (MenuItem, OrderItem, and CafeBillingApplication).

3. Class Definition: Three classes are defined in the program: MenuItem, OrderItem, and CafeBillingApplication. Each class encapsulates related data and functionality.

4. Class Initialization: The classes (MenuItem, OrderItem, and CafeBillingApplication) have an __init__ method for initialization. Note that there is a typo in the code; the method should be named __init__ instead of init.

5. User Input Handling: The input function is used to get user input for menu item selection, quantity, discount percentage, tax percentage, and service charge percentage.

6. List Handling: Lists (menu and current_order) are used to store menu items and the current order, respectively.

7. Menu Operations: Methods like add_menu_item and display_menu are used to add menu items and display the menu, respectively.

8. Order Handling: The take_order method allows the user to input the menu item and quantity for the order, and the details are stored in the current_order list.

9. Billing Calculations: The calculate_bill method calculates the total bill based on the items and quantities in the current order.

10. Discount, Tax, and Service Charge: Separate methods (apply_discount, apply_tax, and
apply_service_charge) are defined to apply discount, tax, and service charge to the bill, respectively.

11. Example Usage: An instance of CafeBillingApplication is created, menu items are added, the menu is displayed, orders are taken, and the final bill is calculated and displayed with optional discount, tax, and service charge.

2. SYSTEM ANALYSIS

a) Feasible study

1. **Technical Feasibility:**

- Quiz Application : The Quiz Application is technically feasible. This means that the required technologies, such as a programming language .

- Daily Task Scheduler: The Daily Task Scheduler is also technically feasible. The required technologies, such as a programming language.

- Cafe BiLling Application: The Cafe BiLling Application is also technically feasible. The required technologies, such as a programming language.

2. **Functional Feasibility:**

- Quiz Application : The Quiz Application is functionally feasible. This means that the application can be developed to meet the needs of the users. For example, the application can be developed to include a variety of question types, such as multiple choice, true/false, and fill-in-the blank.

- Daily Task Scheduler : The Daily Task Scheduler is also functionally feasible. The application can be developed to include features such as task creation, management, and tracking.

- Cafe BiLling Application : Cafe BiLling Application is also functionally feasible. The application can be developed to include features such as order taking, bill calculation, and customer transaction management.

3. **Cost Feasibility:**

- Quiz Application project: The Quiz Application project is cost feasible. The development costs are relatively low, and there are a number of ways to reduce costs, such as using open source software and developing the application in-house.

- Daily Task Scheduler project: The Daily Task Scheduler project is also cost feasible. The development costs are relatively low, and there are a number of ways to reduce costs, such as using open source software and developing the application in-house.

- Cafe BiLling Application project: Cafe BiLling Application project is also cost feasible. The development costs are relatively low, and there are a number of ways to reduce costs, such as using open source software and developing the application in-house.

4. **Time Feasibility:**

- Quiz Application project: The Quiz Application project can be completed within a reasonable timeframe. The scope of the project is not too large, and the required resources are available.

- Daily Task Scheduler project: The Daily Task Scheduler project can also be completed within a reasonable timeframe. The scope of the project is not too large, and the required resources are available.

- Cafe BiLling Application project: Cafe BiLling Applicationproject can also be completed within a reasonable timeframe. The scope of the project is not too large, and the required resources are available.

5. **Risk Assessment:**

- Quiz Application project: The risks associated with the Quiz Application project are relatively low. The project is not too complex, and the required resources are available.

- Daily Task Scheduler project: The risks associated with the Daily Task Scheduler project are also relatively low. The project is not too complex, and the required resources are available.

- Cafe BiLling Application project: The risks associated with the Cafe BiLling Application project are also relatively low. The project is not too complex, and the required resources are available.

b) Summary and Constraint

- Quiz Application project: A software application that allows users to create and take quizzes on a variety of topics.

- Daily Task Scheduler project: A software application that helps individuals and teams organize and manage their daily tasks efficiently.

- Cafe BiLling Application project: A software application that helps cafes and restaurants streamline their billing and order management process.

Constraints:

- Budget: All three projects have a limited budget.

- Timeline: All three projects have a tight deadline.

- Resources: All three projects have a limited number of resources available.

c) Hardware and Software Requirements

Hardware Requirements:

Hardware requirements are the physical components of a computer system that are needed to run a particular software application. The hardware requirements for the three projects are relatively modest, and can be met by most modern computers.

Quiz Application project:

Processor: 1 GHz or higher
RAM: 1 GB or higher
Hard disk space: 100 MB or higher
Internet connection

Daily Task Scheduler project:

Daily Task Scheduler:

Processor: 1 GHz or higher
RAM: 1 GB or higher
Hard disk space: 100 MB or higher
Internet connection

Cafe Billing Application project:

Processor: 1 GHz or higher
RAM: 1 GB or higher
Hard disk space: 100 MB or higher
Internet connection

Software Requirements:

Python and Jupyter can be used to develop all three projects. Python is a popular general purpose programming language that is well-suited for a wide variety of tasks, including web development, data science, and machine learning. Jupyter is a web-based interactive environment that allows you to write Python code and execute it cell by cell.

Quiz Application :

You can use Python and Jupyter to develop the Quiz Application project by writing a Python script that creates a quiz database, generates quiz questions, and presents the questions to the user. You can use Jupyter to test your code and to debug any errors.

Daily Task Scheduler :

You can use Python and Jupyter to develop the Daily Task Scheduler project by writing a Python script that creates a task database, allows users to create and manage tasks, and generates reminders for upcoming tasks. You can use Jupyter to test your code and to debug any errors.

CAFE BILLING APPLICATION :

You can use Python and Jupyter to develop the CAFE BILLING APPLICATION project by writing a Python script that creates a menu database, allows users to place orders, calculates bills, and manages customer transactions. You can use Jupyter to test your code and to debug any errors.

3. DEVELOPMENT ENVIRONMENT

Python and Jupyter are a great development environment for the three projects that you have described. Python is a popular general-purpose programming language that is well-suited for a wide variety of tasks, including web development, data science, and machine learning. Jupyter is a web-based interactive environment that allows you to write Python code and execute it cell by cell.

Here are some of the advantages of using Python and Jupyter as a development environment:

Python is a popular language with a large community. This means that there are many resources available to help you learn Python and to find solutions to problems that you may encounter.

Python is relatively easy to learn. It has a simple syntax and a large number of libraries that can be used for a variety of tasks. It is a powerful tool for interactive development. It allows you to write and execute Python code cell by cell, and to view the results immediately. This makes it easy to experiment with your code and to debug any errors.

4. SYSTEM DESIGN

Modular Description

The system design for the three projects will vary depending on the specific requirements of each project. However, there are some general considerations that apply to all three projects.

One important consideration is the user interface. The user interface should be easy to use and navigate, and it should be designed to meet the needs of the intended users. For example, the user interface for the Quiz Application should be designed to make it easy for users to create and take quizzes. The user interface for the Daily Task Scheduler should be designed to make it easy for users to create and manage their tasks. And the user interface for the CAFE BILLING APPLICATION should be designed to make it easy for cafe staff to take customer orders and calculate bills.

Another important consideration is the system architecture. The system architecture should be designed to be scalable and reliable. It should also be designed to be easy to maintain and update. For example, the system architecture for the Quiz Application project should be designed to handle a large number of users taking quizzes at the same time. The system architecture for the Daily Task Scheduler project should be designed to be reliable and to handle a large number of tasks. And the system architecture for the CAFE BILLING APPLICATION project should be designed to be easy to use for cafe staff. Here are some specific system design considerations for each project:

Quiz Application :

- The Quiz Application should use a database to store the quiz questions and answers. The database should be designed to be efficient and scalable.
- The Quiz Application should use a web framework to generate the user interface and to handle user requests. The web framework should be designed to be secure and reliable.
- The Quiz Application should use a caching mechanism to improve performance. The caching mechanism should be designed to be efficient and scalable.

Daily Task Scheduler :

- The Daily Task Scheduler should use a database to store the user tasks. The database should be designed to be efficient and scalable.
- The Daily Task Scheduler should use a web framework to generate the user interface and to handle user requests. The web framework should be designed to be secure and reliable.
- The Daily Task Scheduler should use a notification mechanism to remind users about upcoming tasks. The notification mechanism should be designed to be reliable and scalable.

CAFE BILLING APPLICATION :

- The CAFE BILLING APPLICATION should use a database to store the menu items, customer orders, and bills. The database should be designed to be efficient and scalable.
- The CAFE BILLING APPLICATION should use a web framework to generate the user interface and to handle user requests. The web framework should be designed to be secure and reliable.
- The CAFE BILLING APPLICATION should use a payment processing gateway to process customer payments. The payment processing gateway should be designed to be secure and reliable.

5. TESTING

a) Testing

Testing is an important part of the software development process. It helps to ensure that the software meets the requirements, that it is free of defects, and that it performs as expected.

There are many different types of testing that can be performed on software. Some of the most common types of testing include:

- Unit testing: Unit testing involves testing individual units of code, such as functions or classes.
- Integration testing: Integration testing involves testing how different units of code work together.
- System testing: System testing involves testing the entire software system as a whole.
- Acceptance testing: Acceptance testing involves testing the software system to ensure that it meets the requirements of the users.

Each type of testing plays an important role in the overall quality assurance process. Unit testing helps to identify defects early on in the development process. Integration testing helps to ensure that different parts of the system work together as expected. System testing helps to ensure that the entire system meets the requirements. And acceptance testing helps to ensure that the system meets the needs of the users.

Quiz Application :

- The Quiz Application should be tested to ensure that the quiz questions are correct and that the answers are scored correctly.
- The Quiz Application should be tested to ensure that the system can handle a large number of users taking quizzes at the same time.

Daily Task Scheduler :

- The Daily Task Scheduler should be tested to ensure that tasks are created, managed, and tracked correctly.
- The Daily Task Scheduler should be tested to ensure that users can receive reminders about upcoming tasks.

- The Daily Task Scheduler should be tested to ensure that the system can handle a large number of users and tasks.

CAFE BILLING APPLICATION :

- The CAFE BILLING APPLICATION should be tested to ensure that customer orders are taken correctly and that bills are calculated correctly.
- The CAFE BILLING APPLICATION should be tested to ensure that payments are processed correctly.
- The CAFE BILLING APPLICATION should be tested to ensure that the system can handle a large number of customers and orders.

6. CONCLUSION

The three projects, the Quiz Application project, the Daily Task Scheduler project, and the CAFE BILLING APPLICATION project, are all feasible and have the potential to be successful. However, there are some constraints that need to be considered, such as budget, timeline, and resources.

Some points

- All three projects are technically feasible and can be developed using Python and Jupyter.
- All three projects are functionally feasible and can be developed to meet the needs of the users.
- All three projects are cost feasible and can be developed within a reasonable budget.
- All three projects are time feasible and can be completed within a reasonable timeline.
- The risks associated with the projects are relatively low.