A

Internship Training Report

On

“Python with Django”

By

Sr.No.	PRN No.	Full Name of the Student
1	23068111242056	Shivani Satish Mali

B.TECH. IN COMPUTER SCIENCE & ENGINEERING

Under the Guidance of

Mr. S. V. Chavan

Academic Year: 2024-2025

DR. BABASAHEB AMBEDKAR TECHNOLOGICAL UNIVERSITY, LONERE

&

SANJAY GHODAWAT INSTITUTE, ATIGRE

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

2024-2025

PROJECT APPROVAL SHEET

This Internship Training report entitled **Python with Django** by **Shivani Satish Mali**, **PRN No: 23068111242056** is approved for partial fulfillment for the B.Tech. in Computer Science & Engineering of Sanjay Ghodawat Institute, Atigre (Kolhapur).

Sr.No.	Full Name of Student	Signature
1	Shivani Satish Mali	
Name of the Mentor		Signature
2	Mr. A. A. Powar	
Name of Head of the Department		Signature
3	Mr. S. V. Chavan	
Name of the Principal		Signature
4	Dr. V. V. Giri	

Date: 22/08/2024 Place: Atigre

Index

1. Introduction:

- **Background:**

  Revolution IT Solutions is a recognized leader in the tech industry, known for offering a range of specialized courses, including Python with Django, Python with Machine Learning, Web Development, and UI/UX Design. My internship took place in the Computer Science & Engineering (CSE) department, where I concentrated on mastering Python with Django. Throughout this period, I was involved in developing various Django-based projects, which allowed me to apply and refine industry-standard web development practices.

- **Objective**:

  During my internship at Revolution IT Solutions, my primary focus was on acquiring a deep understanding of Python and the Django framework. The internship was designed to provide a comprehensive foundation in Python, from its basic concepts to advanced applications. I aimed to not only enhance my coding abilities but also to gain practical, hands-on experience by working on real-world web development projects. This experience was intended to equip me with the skills and knowledge necessary to excel in the tech industry, particularly in backend development.

- **Scope:**

  The scope of my internship was broad, encompassing a wide range of tasks that contributed to a well-rounded learning experience. I started with foundational Python concepts and progressively moved on to more complex Django projects. This hands-on work was directly aligned with my academic focus in Computer Science & Engineering, providing valuable experience in areas such as backend development, database management, and software engineering principles. The projects I worked on not only deepened my technical knowledge but also enhanced my problem-solving skills and ability to apply theoretical concepts in practical scenarios.

2. Company Profile:

- **Overview:**

  Revolution IT Solutions is a leading innovator in the technical industry, known for its commitment to empowering the next generation of tech professionals through cutting-edge training programs. The company stands at the forefront of the industry, continuously evolving to meet the demands of a rapidly changing technological landscape. My internship with Revolution IT Solutions was an invaluable experience, allowing me to immerse myself in the world of Python with Django within their prestigious Computer Science & Engineering (CSE) department. This opportunity provided me with a unique blend of theoretical knowledge and practical application, bridging the gap between academic learning and the real-world challenges of the tech industry. Working in such a dynamic and industry-driven environment not only enhanced my technical skills but also gave me a deeper understanding of how modern web development projects are executed in a professional setting.

- **Services:**

  Revolution IT Solutions offers a comprehensive range of specialized courses, each designed to equip learners with the skills necessary to excel in today's competitive tech industry. Their offerings include Python with Django, Python with Machine Learning, Web Development, and UI/UX Design. These programs are meticulously crafted to cater to both beginners and advanced learners, ensuring that all participants receive a robust education that is directly applicable to real-world scenarios. Beyond the structured curriculum, the company emphasizes hands-on project experience, allowing students to apply their learning in practical settings. This approach ensures that learners are not just theoretically sound but also industry-ready. During my internship, I chose to focus on the "Python with Django" course, which provided me with an in-depth understanding of web development and an opportunity to work on real projects that reflected current industry practices. The course's balance of theoretical concepts and practical application was instrumental in solidifying my knowledge and preparing me for future challenges in the tech field.

- **Organization Structure:**

  Revolution IT Solutions is structured into several key departments, each playing a critical role in the company's overall success. These departments include and Training & Education. My internship was based in the Training & Education department, where I was specifically placed within the Computer Science & Engineering (CSE) team. This department is known for its rigorous and thorough approach to training, ensuring that learners receive the highest quality education and mentorship. Under the guidance of experienced mentors, I focused on mastering Python with Django, gaining insights not only into the technical aspects of the framework but also into how it is applied in professional software development. The mentorship and support provided by the CSE team were invaluable. This structured and supportive environment allowed me to develop a strong foundation in Python with Django, which I am confident will serve me well in my future career.

3. Internship Tasks and Responsibilities:

- **Description of Work:** During my internship, I gained comprehensive experience given:

Python Programming:

a) **Fundamentals:** I mastered the core concepts of Python, including variables, data types, operators, and control flow using conditional statements (if, else, elif) and loops (for, while). I also developed functions to modularize code, making it more reusable and easier to manage.

b) **Exception Handling:** I learned how to handle errors gracefully using try and except, ensuring that programs run smoothly even when unexpected issues arise.

c) **Data Structures:** I effectively used essential data structures such as lists, tuples, dictionaries, and sets to organize and manage data efficiently.

d) **Object-Oriented Programming:**

I developed skills in defining classes and objects, which are the building blocks of OOP. I also implemented inheritance to reuse code across multiple classes and applied encapsulation to protect data and functions within a class.

- **Tools and Technologies Used:** Throughout the internship, I utilized a variety of tools and technologies including:

a) Programming Languages: Python.
b) Framework: Django.
c) Version Control: Git and GitHub for version control and collaboration.
d) IDE: PyCharm/VS Code for code development.
e) APIs: RESTful APIs for backend data handling.

- **Challenges Faced:**

a) Navigating Django's ORM for efficient database queries.
b) Implementing Django's security features to protect web applications.
c) Managing version control in a team setting effect.

4. Learning Outcomes:

- **Technical Skills:**

  During my internship, I significantly strengthened my technical abilities. I gained a deep understanding of Python programming and the Django framework, mastering both basic and advanced concepts. My database management skills improved as I worked with Django's ORM to handle data efficiently. I also became proficient in using version control systems like Git and GitHub, which helped me manage code changes and collaborate effectively with my team.

- **Soft Skills:**

  The internship also helped me develop essential soft skills. Regular team interactions enhanced my communication abilities, making it easier to share ideas and receive feedback. Working in a collaborative environment improved my teamwork skills, teaching me how to contribute effectively and work well with others. I also honed my time management skills, learning to prioritize tasks and meet deadlines consistently.

- **Practical Experience:**

  One of the most significant outcomes of this internship was the practical experience I gained, which bridged the gap between academic knowledge and its real-world application. The projects I worked on required me to apply theoretical concepts in a practical context, particularly in developing and deploying web applications using Python and Django. This hands-on experience was invaluable, as it gave me a deeper understanding of how software development works in a professional setting, from initial planning to final deployment. I also had the opportunity to tackle real-world challenges, such as implementing security features and managing version control in a team environment. This practical experience not only solidified my technical skills but also boosted my confidence in applying them to future projects.

5. Conclusion:

My internship at Revolution IT Solutions was an invaluable experience that greatly contributed to my academic and professional growth. The opportunity to work closely with Python and the Django framework allowed me to bridge the gap between theoretical knowledge and practical application, solidifying my skills in web development and backend technologies.

As part of my internship, I completed a project titled "Personal Bill Manager." The objective of this project was to develop a simple Bill Management System using Python and Tkinter. This system enables users to enter details about their bills, such as bill type, due date, amount, status, and payment method. Additionally, the system includes features for visualizing bill data through charts. Working on this project reinforced my understanding of database management, RESTful APIs, and full-stack development, making it a significant milestone in my learning journey.

Overall, this internship has equipped me with essential skills that are crucial for my future career in computer science. I would recommend future interns to take full advantage of the resources and mentorship offered at Revolution IT Solutions, as it provides a strong foundation in both technical skills and professional development. Engaging actively in projects and team activities will not only enhance their learning experience but also prepare them effectively for the challenges of the tech industry.

6. Mini Project:

- **Project Title:** Personal Bill Manager.

- **Objective:**

  The purpose of this project was to create a simple Bill Management System using Python and Tkinter. The system allows users to enter details about their bills, such as bill type, due date, amount, status, and payment method. Additionally, users can view charts to visualize their bill data.

- **Methodology:**
  a) **Planning:** We decided on the main features: adding bills, saving the data, and tracking the status of each bill. We also planned to add charts to show the status and payment methods of the bills.
  b) **Design:** We created a simple and user-friendly interface using Tkinter, where users can input bill details like type, due date, amount, status, and payment method.
  c) **Data Handling:** We used pandas to manage the bill data, which made it easy to store and update information.
  d) **User Interface:** Tkinter buttons and text boxes were used to let users add bill details. We displayed the entered bills in a table format for easy viewing.
  e) **Charts:** We added charts using Matplotlib. A pie chart shows how many bills are paid or unpaid, and a bar chart shows the different payment methods (cash or online).
  f) **Testing:** We tested the project at every step, fixing any issues that came up to make sure everything worked correctly.
  g) **Final Touches:** After testing, we checked everything to make sure the project worked smoothly and was accurate.

- **Implementation:**

  The project was implemented using Python, Tkinter for the user interface, pandas for data storage, and Matplotlib for chart generation. The following steps were taken during the implementation:

  a) **User Interface Creation:** We used Tkinter to create the graphical user interface where users can enter bill details, including the bill type, due date, amount, status (Paid/Unpaid), and payment method (Cash/Online).

  b) **Data Storage and Handling:** Bill details are stored in a pandas Data Frame, which allows for easy manipulation and updating of records. New bills are appended to this Data Frame, and it updates the table view with the latest data.

c) **Data Display:** A ttk. Treeview widget was used to create a table that displays all the bills entered by the user. This table automatically updates when new bills are added.

d) **Chart Generation:** Using Matplotlib, a pie chart and a bar chart were generated to visualize the status of the bills (paid vs. unpaid) and the payment methods used, respectively.

e) **Application Flow:** The Tkinter application runs in a loop, ensuring that the interface stays active and responsive to user input.

f) **Program:**

```
import tkinter as tk

from tkinter import ttk

import pandas as pd

import matplotlib.pyplot as plt

from tkinter import messagebox

from datetime import datetime

# File to store bills

FILE_NAME = "bills.csv"

# Load existing data if file exists

def load_data():

try:

df = pd.read_csv(FILE_NAME)

if 'Payment Method' not in df.columns:

df['Payment Method'] = ''  # Add column with empty values if it doesn't exist

return df

except FileNotFoundError:

return pd.DataFrame(columns=['Bill Type', 'Due Date', 'Amount', 'Status',
'Payment Method'])
```

```
# Function to add a bill
def add_bill(bill_type, due_date, amount, status, payment_method):
df = load_data()
new_data = pd.DataFrame([{
'Bill Type': bill_type,
'Due Date': due_date,
'Amount': float(amount),
'Status': status,
'Payment Method': payment_method }])
    df = pd.concat([df, new_data], ignore_index=True)
    df.to_csv(FILE_NAME, index=False)
    messagebox.showinfo("Success", "Bill added successfully!")
    update_table()
# Function to update the table in the GUI
def update_table():
    for row in table.get_children():
        table.delete(row)
    df = load_data()
    for index, row in df.iterrows():
        table.insert("", "end", values=(row['Bill Type'], row['Due Date'],
row['Amount'], row['Status'], row['Payment Method']))
# Function to generate pie chart
def generate_pie_chart():
    df = load_data()
```

```
    df.groupby('Status').sum().plot(kind='pie', y='Amount', autopct='%1.1f%%')
    plt.title("Bills by Status")
    plt.show()
# Function to generate bar chart
def generate_bar_chart():
    df = load_data()
    df['Due Date'] = pd.to_datetime(df['Due Date'])
    df.groupby(df['Due Date'].dt.date)['Amount'].sum().plot(kind='bar')
    plt.title("Daily Bills")
    plt.show()

# Main application window
root = tk.Tk()
root.title("Bill Management")
# Set black background and white font
label_font = ("Times New Roman", 12)
root.configure(bg="black")

# GUI elements for adding bill
tk.Label(root, text="Bill Type", font=label_font, fg="white",
bg="black").grid(row=0, column=0)
tk.Label(root, text="Due Date (YYYY-MM-DD)", font=label_font, fg="white",
bg="black").grid(row=1, column=0)
tk.Label(root, text="Amount", font=label_font, fg="white",
bg="black").grid(row=2, column=0)
```

```
tk.Label(root, text="Status (Paid/Unpaid)", font=label_font, fg="white",
bg="black").grid(row=3, column=0)

tk.Label(root, text="Payment Method (Cash/Online)", font=label_font,
fg="white", bg="black").grid(row=4, column=0)

bill_type_entry = tk.Entry(root)

due_date_entry = tk.Entry(root)

amount_entry = tk.Entry(root)

status_entry = tk.Entry(root)

payment_method_entry = tk.Entry(root)

bill_type_entry.grid(row=0, column=1)

due_date_entry.grid(row=1, column=1)

amount_entry.grid(row=2, column=1)

status_entry.grid(row=3, column=1)

payment_method_entry.grid(row=4, column=1)



# Add Bill Button

add_button = tk.Button(root, text="Add Bill", command=lambda:
add_bill(bill_type_entry.get(), due_date_entry.get(), amount_entry.get(),
status_entry.get(), payment_method_entry.get()), font=label_font, fg="white",
bg="black")

add_button.grid(row=5, column=1)

# Table for displaying bills

columns = ("Bill Type", "Due Date", "Amount", "Status", "Payment Method")

table = ttk.Treeview(root, columns=columns, show="headings")

for col in columns:

    table.heading(col, text=col)
```

```
table.grid(row=6, column=0, columnspan=3)

# Dashboard buttons with black background and white text

pie_button = tk.Button(root, text="Show Pie Chart",
command=generate_pie_chart, font=label_font, fg="white", bg="black")

bar_button = tk.Button(root, text="Show Bar Chart",
command=generate_bar_chart, font=label_font, fg="white", bg="black")

pie_button.grid(row=7, column=0)

bar_button.grid(row=7, column=1)

# Load data into table when program starts

update_table()

root.mainloop()
```

- **Results:** The system successfully allows users to:

a) Add and manage bills through a simple interface.
b) View bills in a clear, organized table.
c) Generate charts to analyse bill status and payment methods.