[Your College Logo]

Industrial Internship Report on

” Secure GUI-Based Password Manager”

Prepared by

Reshma Raji U

Executive Summary
This report outlines the six-week Industrial Internship conducted by **Upskill Campus** and **The IoT Academy**, in collaboration with **UniConverge Technologies Pvt Ltd (UCT)**. The internship focused on solving an industrial-level problem assigned by UCT and included building, testing, and documenting the solution. My project was titled **“Secure GUI-Based Password Manager”**, which aimed to provide a desktop application for securely storing, generating, and managing passwords using a Tkinter-based GUI with SHA-256 authentication. This internship offered invaluable hands-on experience in understanding industrial problem-solving processes, design architecture, and the deployment of secure and user-friendly Python applications. It was a highly enriching experience that strengthened my technical and analytical skills.

TABLE OF CONTENTS

1 Preface

Over the course of six weeks, I developed a secure Password Manager using Python. The project included GUI development with Tkinter, password encryption using SHA-256, and a strong password generator. This experience provided exposure to real-world design considerations such as usability, security, and cross-platform support.

The internship, provided by Upskill Campus (USC) and UniConverge Technologies Pvt Ltd (UCT), was well-structured with weekly tasks, mentoring sessions, and feedback. I would like to thank my mentors, especially those from UCT and USC, for their constant guidance and support.

This internship gave me an early taste of industrial challenges and reaffirmed my desire to pursue software engineering with a focus on security and full-stack development.

2 Introduction

2.1 About UniConverge Technologies Pvt Ltd

Established in 2013, UCT works in the domain of Digital Transformation and Industrial IoT. They develop products using cutting-edge technologies like IoT, Cybersecurity, Cloud Computing, ML, 4G/5G, and more. Their platforms include:

- **UCT Insight** – IoT platform
- **Factory Watch** – Smart factory platform
- **LoRaWAN-based solutions** – For agriculture, metering, and smart cities
- **Predictive Maintenance** – Using embedded and machine learning

2.2 About Upskill Campus (USC)

USC is a career development platform that collaborated with UCT and IoT Academy to facilitate this internship and manage logistics, content delivery, and coordination.

2.3 The IoT Academy

The EdTech arm of UCT offering executive programs with IITs and EICT Academy in domains like IoT, AI, and cybersecurity.

2.4 Objectives of the Internship

- Gain industrial exposure and hands-on experience
- Solve real-world challenges
- Improve communication and coding skills
- Enhance problem-solving abilities

3 Problem Statement

Develop a **desktop-based Password Manager** with the following features:

- Secure authentication using SHA-256 hashing
- Password encryption and storage
- Password generation based on customizable rules
- User-friendly GUI for non-technical users
- Support for managing multiple users (prototype)

4 Existing and Proposed solution

Tools like LastPass or KeePass provide secure password storage. However, many are either fully cloud-based (raising privacy concerns) or lack customization and local control.

4.1.1 Proposed Solution:

- Local-first: Entirely offline storage using local encrypted files
- Secure login: Master password authentication using SHA-256
- Password Generator: Built-in generator with rules (length, symbols, etc.)
- Prototype multi-user support
- User-friendly UI with error handling and visual feedback

Code Submission

https://github.com/ReshmaRajiu29/upskillCampus--.git

Report Submission

https://github.com/ReshmaRajiu29/upskillCampus--.git

5 Proposed Design/ Model

5.1 High Level Diagram

User → GUI → Authenticator → Encrypted Database
↘
Password Generator

5.2 Low Level Diagram

- GUI → Login Page → Main Panel → Add/View Passwords
- SQLite (or JSON) storage with hashed credentials
- All passwords encrypted before saving

5.3 Interfaces

- Tkinter for frontend
- hashlib for hashing
- os & json for local file handling
- ttk for styling and themes

6.Performance Test

6.1 Constraints

- Response time for large password databases
- CPU usage during encryption
- Secure password hashing
- UI responsiveness

6.2 Test Plan

- Insert/delete passwords under various loads
- Attempt login with incorrect credentials
- Generate passwords repeatedly

6.3 Performance Outcome

- Application handled over 200 records without slowdown
- Login verified within 200ms average
- Generator maintained randomness and strength across tests

7.My learnings

1.Implemented cryptographic concepts like SHA-256

2. Built a complete GUI application using Tkinter and ttk

3.Understood modular design and code reusability

4. Gained experience in user-centered design

5. Strengthened skills in debugging, Git, and deployment

8.Future work scope

- Add real multi-user support with file-based isolation
- Integrate biometric login (e.g., facial recognition)
- Sync encrypted data via the cloud securely
- Convert to a web-based application using Flask or Django
- Create an installer or EXE version using PyInstaller