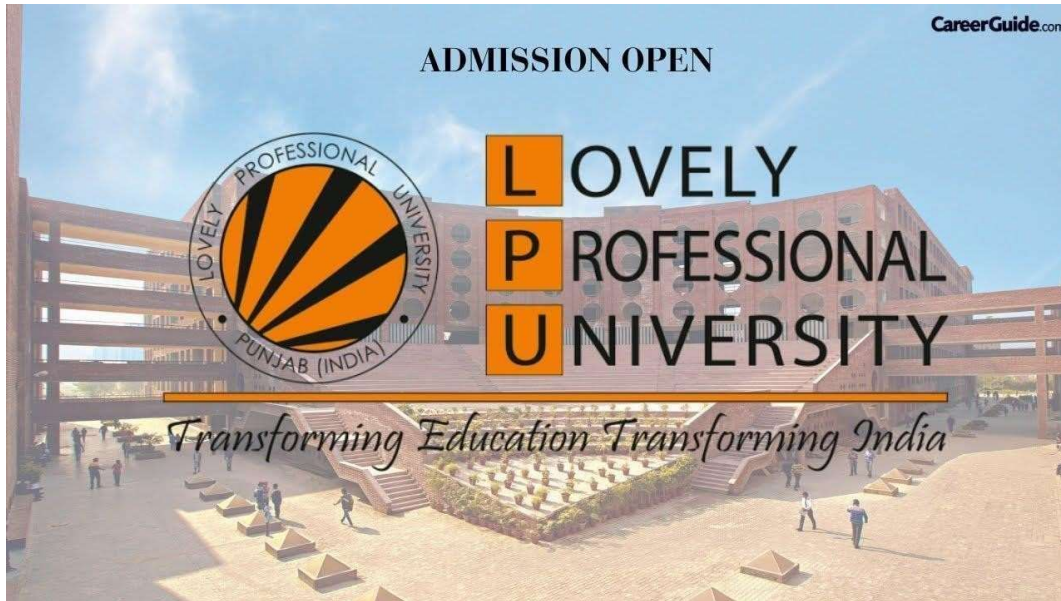


Interim Report CA-II



Course Name – **Python Project**

Course Code – **CSM216**

Section - **K23CH**

Student Group — **2**

Project Title – **VILLAGE DATA INFO HUB**



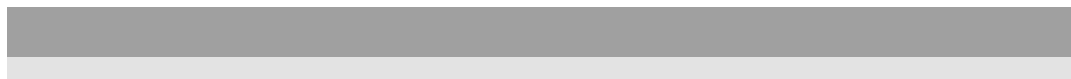
Project Title: **VILLAGE DATA INFO HUB**

Student Name: Banoth Srikanth

Reg No: **12303313**

Roll No: **47**

Submission Date: **18/11/2024**



Acknowledgment

I would like to express my sincere gratitude to my course instructor **Mr. Aman Kumar**, for his valuable guidance and feedback throughout the development of this project. I also appreciate the support from my peers who helped me with different aspects of this project. Their insights have been instrumental in enhancing the quality of my work.

Table of Contents

1. **Introduction**
 2. **Objectives and Scope of the Project**
 3. **Application Tools**
 4. **Project Design**
 5. **Flowchart**
-

I. Introduction

The **Village Data Information Hub** is an application designed to help manage the data of villages, including family and individual records. It is developed to serve the needs of **DEO (District Education Officers)** and **Village Officials**, enabling them to access, manage, and update family information, member details, and village statistics. The project simplifies the data entry and retrieval processes for village officials and DEOs while maintaining security and user-specific access.

This system offers features like:

- **Login functionality** for both DEOs and Village Officials.
- **Management of family records**, including adding, updating, and viewing families.
- **Tracking individual family members** with details such as gender, age, qualification, occupation, etc.
- **Role-based access control** ensuring that Village Officials can only manage the data relevant to their specific village.

The purpose of this project is to make village-level data management easier and more efficient, using a centralized database accessible via a user-friendly graphical interface.

II. Objectives and Scope of the Project

Objectives:

1. Develop a Python-based desktop application using **Tkinter** for a user-friendly interface.
2. Implement role-based login authentication for **DEOs** and **Village Officials**.
3. Enable **DEOs** to create user IDs and passwords for **Village Officials**.
4. Provide the ability for **Village Officials** to manage family and member data including adding, editing, and viewing records.
5. Store data in a **MySQL database**, ensuring persistence and easy retrieval of records.
6. Integrate the **DEO's** ability to oversee data and allow **Village Officials** to manage only specific data related to their respective villages.

Scope:

- The project focuses on the **user interface** for easy access to data and **database management** for secure data storage.
 - It targets **DEOs** and **Village Officials** as primary users, allowing them to input and retrieve family and member records efficiently.
 - The project is intended for **local deployment** with access to MySQL for data persistence.
-

III. Application Tools

Programming Language:

- **Python** – The main language used for the application development.

IDEs:

- **PyCharm** or **Visual Studio Code** for coding and debugging the Python application.

Libraries/Packages:

- **Tkinter** – For the graphical user interface (GUI).
- **MySQL Connector** – For connecting the Python application to a MySQL database.

Version Control:

- **Git**.

Other Tools:

- **MySQL** – For database management and storage.

IV. Project Design

The **Village Data Information Hub** project is structured into various components, with separate modules for handling database connections, user interfaces, and application logic. The major parts of the project are as follows:

1. Login and Authentication:

- The login page allows two types of users: **DEO** and **Village Official**.
- **DEOs** can create and manage user credentials for **Village Officials**. Upon login, DEOs can access a page for managing villages, creating accounts for officials, overseeing & also managing family data.
- **Village Officials** log in using their credentials to manage family records for their specific village.

2. Family Management:

- Village Officials can add a new family by entering the **head of the family's name** and **ration card number**.
- The system allows adding **family members** by selecting an existing family ID and entering personal details such as name, gender, date of birth, aadhar number, phone number, blood group, and occupation.
- Village Officials can **view family records**, including family heads and family members.

3. Database Operations:

- The project uses **MySQL** to store user credentials, family, and member information.
- It includes tables for **DEOs**, **Village Officials**, **Families**, and **Family Members**.

4. Navigation:

- The user interface is designed to have forward and backward navigation buttons to move between the pages for **adding families**, **viewing families**, and **managing family members**.

5. Security:

- Role-based access control ensures that Village Officials can only access data relevant to their villages, while DEOs have broader access to manage all villages.

V. Flowchart

