DESIGN DOCUMENT

- **ARCHITECTURE**:

The Library Management System was implemented using python . The GUI is created using Python PyQt5 package and the back end code is written in python . Both GUI and implementation are done in a single python file.

- **DATABASE CREATION AND CONNECTION :**

1.The data was imported into database from books.csv and borrowers.cv files provided by using MySql Workbench .

2. Database: LIBRARY_MANAGEMENT consisted of the tables given in schema that were made backward compatible with the appropriate key constraints.

Data was normalized using the following steps:
> ISBN (ISBN10) was chosen as unique identifier and used as the primary key for consequential tables instead of the both the ISBN10 and ISBN13.
> Multiple attributes were combined into a single attribute through concatenation in the case of bname (first name, last name) and address (address, city, state).
> The book table was populated with another attribute, status, a boolean value that states whether the book exists in the database to check_out.

3.A local mysql server was created to host the database . We can connect the GUI to the database from python using MySqldb dependency . It has cursor( ) object which can be used to connect to DB.

- **GUI IMPLEMENTATION:**

The index.py file has both GUI implementation and backend code for executing queries and updating tables . GUI was implemented using Python's PyQt5 package . The GUI has 5 tabs , each of them implementing a single functionality of the library system**:** SEARCH, CHECK_IN, CHECK_OUT, BORROWERS AND FINES.

SEARCH TAB:

- A text field to search the database through any combination of ISBN13, title or the author name was coupled to the Search button with an action event listener. Upon called, the data fetched is displayed in the table view provided.

CHECK_OUT TAB:

- Consists of two text fields, ISBN number and Card Number which together are used to check out the book when Check Out button is used. A maximum of 3 books are allowed for a card number .

CHECK_IN TAB:

- This tab has 3 text input fields along with one table view and 2 buttons .

- Books that are borrowed are displayed by entering at least one of ISBN/Card number/Name and clicking on Check for Loans button.

- Books can be checked in by selecting ISBN of the book to be checked in displayed in the table view and clicking the check in button.

BORROWER TAB:

- This tab has 5 input fields Name , Email, SSN , Phone and address and 2 buttons. All the fields are required to successfully add a borrower into borrower table.Each borrower has a unique SSN . Validations are used for Phone,SSN , Email to check the if data in these fields is appropriate . A borrower with a new SSN can be successfully added by filling in all the details and clicking on add borrower button .

- A Reset Details button is provided to clear all the details to add a new borrower

FINES TAB:

- This tab has one input field , one table view and 4 buttons . Fines can be checked by entering the Card Number in the input field and clicking on Check Fines button . The data is displayed in the table view .

- Fines can be paid by selecting the row and clicking on Pay Fine button .Fine Amount for a Card Number can be displayed by clicking on Fine Amount button

- Refresh Fines button refreshes Fines table and updates fine amount that borrower has paid .

- Validation is provided to reject Fine amount if the borrower has not checked in the book .