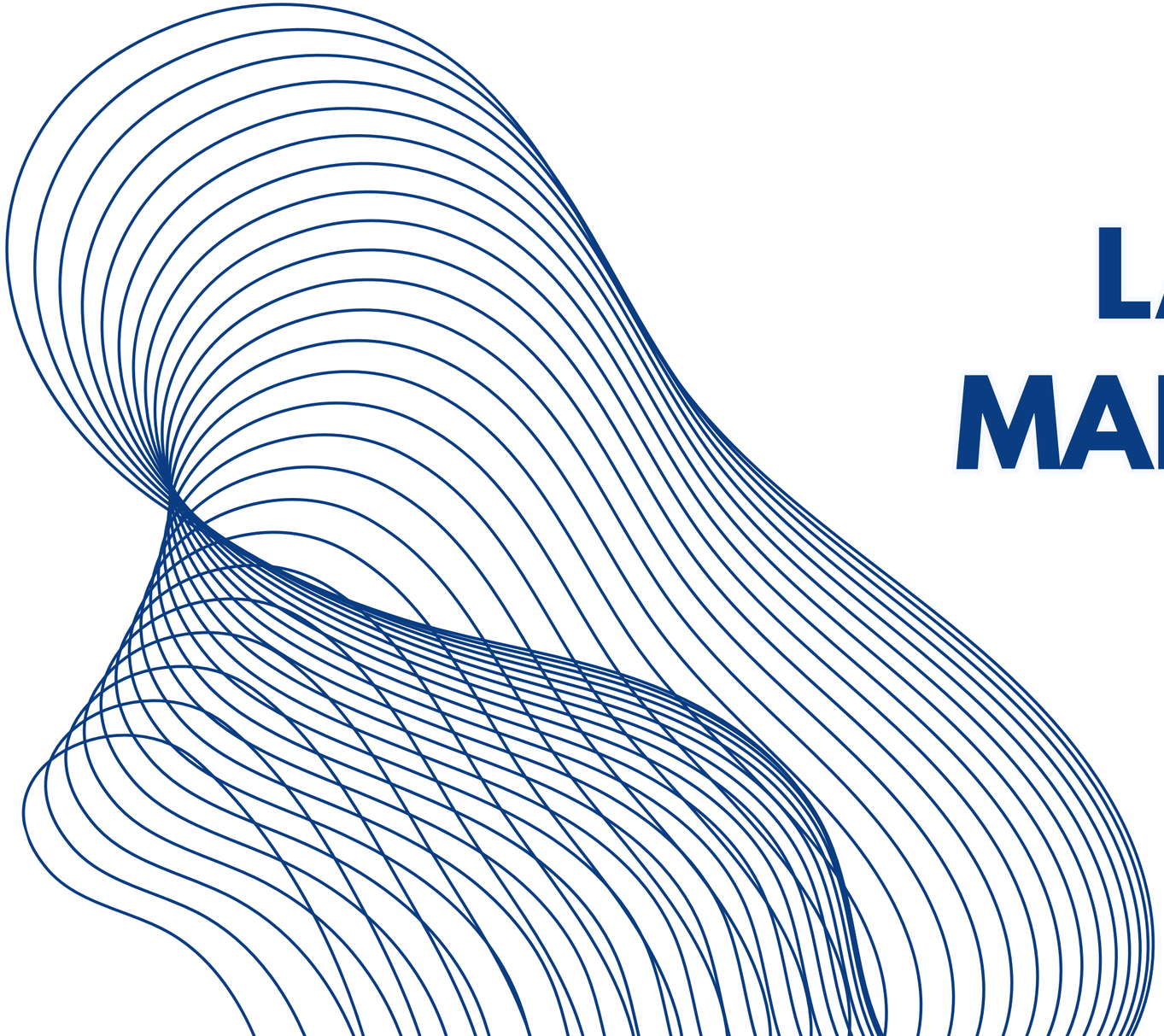


LAB MATERIALS MANAGEMENT GUI

1. OBJECTIVES(S)

GENERAL OBJECTIVE:

- The program should have a Graphical User Interface (GUI)
- The program should utilize OOP Concepts (Class, Object, Polymorphism, Inheritance, Encapsulation, Abstraction)
- The program must be related to Computer Engineering

SPECIFIC OBJECTIVE:

- The program will be done using the Python programming language
- The program will utilize lists to hold instances of the materials, and dictionaries to store the materials with their respective quantities

2. INTENDED LEARNING OUTCOMES (ILOS)

- Students will demonstrate their ability to apply object-oriented programming concepts
- Students will acquire proficiency in creating user-friendly graphical user interfaces (GUIs) using Python and relevant libraries
- Students will apply fundamental data structures (such as lists and dictionaries) to effectively manage and manipulate the materials database within the system.

3. DISCUSSION

I. BACKGROUND AND CONCEPTS

- This program simplifies managing laboratory materials by tracking items and allowing students to borrow them easily. It ensures accurate inventory and equal access for all students.
- The program uses OOP to organize code, with classes defining materials, polymorphism handling different materials, inheritance sharing features, encapsulation protecting data, and abstraction simplifying functions.
- The GUI is created with PyQt5, making it easy for students and admins to borrow materials with buttons and text fields.
- Libraries Used:
  - sys: For system tasks.
  - csv: Handles material data.
  - PyQt5: Builds the GUI.
  - datetime: Shows the current time.
  - pytz: Manages time zones.

3. DISCUSSION

II. SCOPE AND LIMITATIONS

- **Scope**
  - The program helps manage lab materials, allowing students and admins to track and borrow items easily. It provides a simple interface for user registration and logging borrowings.

- **Limitations**
  - Limited to PC users
  - Is not connected to the internet
  - Data updates are local, which may cause inaccuracies with multiple users.

4. MATERIALS AND EQUIPMENT

- Personal Computer or Laptop with Python installed (Python 3.x)
  - Use any IDE that you are comfortable using to develop, run, and debug the program such as:
    - PyCharm
    - Visual Studio Code (VS Code)
    - Spyder

- Make sure that PyQt5 and pytz libraries are installed using pip

5. PROCEDURE

- **ILO A:** The students will demonstrate their ability to apply object-oriented programming concepts

- **ILO B:** Students will acquire proficiency in creating user-friendly graphical user interfaces (GUIs) using Python and relevant libraries

- **ILO C:** Students will apply fundamental data structures (such as lists and dictionaries) to effectively manage and manipulate the materials database within the system.

All Intended Learning Outcomes (ILOs) have been successfully achieved.

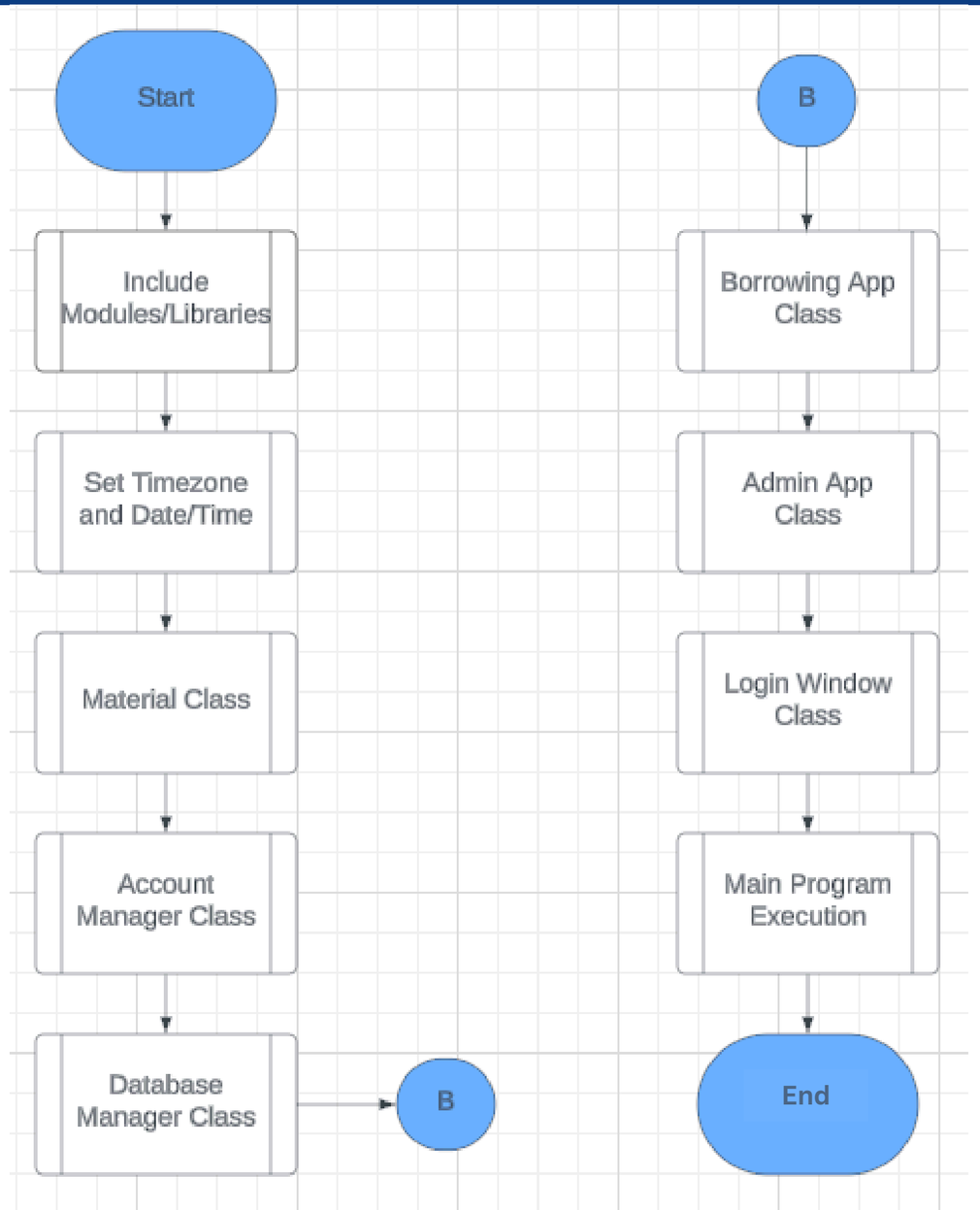
5. PROCEDURE

1. Include Modules/Libraries
2. Set Timezone and Date/Time
3. Material Class
4. Account Manager Class
    - Initialize with a file (account.txt) to store user information.
    - Add registration and login methods to manage user accounts and credentials.
5. Database Manager Class:
    - Initialize with a file (database.txt) to manage materials.
    - Add methods to load materials, and manage (add, remove, save) the materials database.

5. PROCEDURE

6. Borrowing App Class:
    - Initialize with the student's information.
    - Set up the user interface for borrowing materials and manage the borrowing process.
7. Admin App Class:
    - Initialize with an interface for managing the materials database.
    - Add methods for adding, updating, and removing materials.
8. Login Window Class:
    - Handle user login and registration.
    - Set up input fields and buttons for login and registration, and authenticate users.
9. Main Program Execution:
    - Initialize the application, create an instance of the Login Window, and start the program.

5. PROCEDURE


Start
Include Modules/Libraries
Set Timezone and Date/Time
Material Class
Account Manager Class
Database Manager Class
B
B
Borrowing App Class
Admin App Class
Login Window Class
Main Program Execution
End

6. OUTPUT

7. CONCLUSION

- Provides an easy-to-use interface for students and admins.
- Includes secure login and registration for student accounts.
- Admins can update and manage materials to keep the inventory accurate.
- Allows students to easily borrow materials with error handling for common issues.
- Overall, improves the efficiency and ease of managing lab materials.

8. RECOMMENDATIONS

- Allow User Feedback
- Regular Updates
- Mobile Version
- Enhanced Security
- Real-Time Inventory
- Database Backup
- Integration with Academic Systems

MEMBERS:

JAMES RUSSEL ADIA

CARL JERVIE CARAG

MARK JEONEL KENN GOB

DANICA GUARIÑO

ARIES RIO

THANK YOU