

<b>Seatwork 11.1</b>	
<b>Planning</b>	
<b>Course Code:</b> CPE010	<b>Program:</b> Computer Engineering
<b>Course Title:</b> Data Structures and Algorithms	<b>Date Performed:</b> 10/21/2025
<b>Section:</b> CPE21S4	<b>Date Submitted:</b> 10/21/2025
<b>Name(s):</b> Bautista, Mariela Cabrera, Gabriel Pizarro, Jesus Guille Quioyo, Angelo	<b>Instructor:</b> Engr. Jimlord Quejado
<b>Proposed System A: Bank Simulator</b>	
<p><b>Functions:</b></p> <ul style="list-style-type: none"> <li>• Simulate bank functions</li> <li>• Account number and PIN number</li> <li>• Withdrawal, deposit, check balance, money transfer ,transaction history(*)</li> <li>• Management/Admin mode(adding, deleting accounts)</li> <li>• Queue, stack, linked list, vector</li> </ul> <p><b>Questions:</b></p> <ol style="list-style-type: none"> <li>1. How does the program assign bank account numbers to the user?</li> <li>2. Which data structure is the most efficient for storing account information?</li> <li>3. How does the program check for insufficient fund withdrawal?</li> <li>4. How would the transfer funds work?</li> <li>5. Which values should be updated after every transaction?</li> <li>6. Does the program save each value after every transaction?</li> <li>7. Can the program generate account statements?</li> <li>8. How can we ensure that the user can only transact on their own account?</li> <li>9. How can someone access the Management/Admin mode?</li> <li>10. Can we add date and time for the transaction history?</li> </ol>	

## **Proposed System B: Library Book Tracker**

### **Functions:**

- Add New Book – input book title, author, and ID number.
- View All Books – show all books currently stored in the system.
- Search Book – find a book by title or author.
- Borrow or Return Book – update the availability status of a book.
- Delete Book (Admin only) – remove a book record if it's no longer in the library.

### **Questions:**

1. What information should each book record contain (e.g., ID, title, author, status)?
2. Who can add or remove books? The admin only or regular users too?
3. How does the system know if a book is available or borrowed?
4. What happens if a user tries to borrow a book that's already taken?
5. Should users be able to search by both title and author?
6. How many books can the system store at once?
7. Should the system show borrowed books separately from available ones?
8. Does the system need to track which user borrowed a specific book?
9. Should the system save book records even after it's closed (using files)?
10. How should the admin view or manage all book records efficiently?

## **Proposed System C: Student Record Search & Sort System**

### **Functions:**

- Add Student Record – input student name, ID, and grade.
- View All Students – show all records stored in the system.
- Sort Students – arrange students by name or grade.
- Search Student – find a student using ID or name.
- Delete Record (optional) – remove a student record if needed.

### **Questions:**

1. What information should each student record contain (e.g., ID, name, grade)?
2. How will users add a new student to the system?
3. Who can edit or delete a student's record?
4. Should the system allow duplicate student IDs?
5. What order should be used when sorting — by name or by grade?
6. How will the user know if the search found the right student?
7. Should users be able to view all students at once?
8. How will errors (like wrong input) be handled?
9. Should the program save data even after it's closed (file saving)?

10. How many student records should the system be able to store?

### Gantt Chart

We chose the **Waterfall model** for the SDLC because a banking system requires a higher level of security and stability. Unlike systems that need frequent updates, banking systems benefit from a structured, linear development process with thorough documentation and rigorous testing at each stage to ensure reliability and compliance.

