### 1. Project Objectives

Implement a program that contains a record of people using a list structure. Use either contiguous or linked list implementation. Insert a database from a text file. Text file has a list of last names, first names, and ID numbers.

### 2. Task Breakdown

| **Task #** | | **Description** | | **Status** | | **Assigned To** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | | Understand class structure and data model | | Done | | Both |
| 2 | | Implement core functionalities | | Undergoing | | Both |
|  | 2-1 |  | Insert records in alphabetical order |  | Done | Jonathan Mabery |
| 2-2 | Print the contents of the record | Done | Andrew Aprile |
| 2-3 | Support search functionality | Done | Both |
| 3 | | Error handling and input validation | | Done | | Andrew Aprile |
| 4 | | Test application | | Done | | Both |
| 5 | | Finalize code and documentation | | Done | | Both |

### 3. Timeline

Make a tentative timeline to manage the project progress.

| **Date** | **Task** | **Notes** |
| --- | --- | --- |
| 9/13/25 | Start Project. Setup GitHub | Done |
| 9/15/25 | Complete task 1 | Done |
| 9/16/25 | Complete task 2 | Done |
| 9/20/25 | Complete task 3 | Done |
| 9/20/25 | Testing and debugging | Done |
| 9/21/25 | Submit project, code, report, documentation | Done |

### 4. Technical Details

* **Language:** C++ (Standard: e.g., C++11 or C++14)
* **Development Environment:** [e.g., Windows 11, Visual Studio Code, g++, etc.]
* **Key Concepts Used:**
  + OOP (Classes/Objects)
  + File I/O
  + List
  + Sorting
  + String matching and comparison
  + Efficient manipulation of a customized data type

### 5. File Structure

/p01\_group17.zip

│

├──/Code

├── List.h

├── utility.h

├── main.cpp

├── utility.cpp

├── makefile

├── data.txt

├── people.txt

├── smallpeople.txt

├── input\_simulate.txt

├── p01\_report.docx

├── p01-report.pdf

├── ProjectPlan.docx

├── ProjectPlan.pdf

└── README.md

### 6. Testing Plan

* **Manual test cases**
  + Create test cases
    - User selects invalid menu option.
      * Solution:
      * Display error message.
      * Display menu options again
    - User selects “Import List from File”. Then user inputs a file name that doesn’t exist
      * Solution:
      * Display message that says “Added 0 entries”
      * Display menu options again
    - User wants to display or search a record before adding entries
      * Solution:
      * Display a message saying that the record is empty
      * 
    - When the user selects the search list menu option and doesn’t enter the option for id or first last name.
      * Solution:
      * Display error message and ask again for input
      * A screenshot of a computer

        AI-generated content may be incorrect.
  + Input scenarios with expected vs actual output
* **Test in a standard environment**
  + Set up GitHub Actions
  + Compile the code
  + Optional: test cases
    - Duplicate entries
    - Invalid user input

### 7. Documentation

* **README.md** with:
  + Project description
  + How to compile and run
  + Usage instructions
* Code comments for clarity

### 8. Completion Checklist

| **Item** | **Done (Y / N)** |
| --- | --- |
| All features implemented | Y |
| Code compiles without errors | Y |
| Optional: all test cases pass | Y |
| Proper documentation | Y |
| Code is commented | Y |
| Submitted on time | Y |