InCollege Project: Profile Viewing & Basic Search

Objective: This week, your team will enhance the InCollege application by enabling users to view their own complete profile information and to perform a basic search for other users. This is a critical step towards fostering connections within the InCollege community. All program input will continue to be read from a file, all output will be displayed on the screen, and that same output will also be written to a file.

Focus Areas:

1. **Enhanced Self-Profile Viewing:**
   o The "View My Profile" option, which was introduced in Week 2, must now reliably display all information that a user has entered into their profile (First Name, Last Name, University/College, Major, Graduation Year, About Me, Experience entries, and Education entries).
   o The displayed profile should be clearly formatted for readability in the console.
2. **Basic User Search Functionality:**
   o The "Find someone you know" option, previously an "under construction" message, will now become functional.
   o Users will be able to search for other registered InCollege users by their full name (e.g., "John Doe").
   o The search should perform an exact match against the stored user names.
   o If a match is found, the system should display the full profile of the found user.
   o If no match is found, the system should inform the user that no one by that name could be found.
3. **I/O Requirements:**
   o **Input:** All user input (e.g., menu selections, search queries) will be read from a predefined input file.
   o **Output Display:** All program output (e.g., prompts, confirmation messages, displayed profiles, search results) must be displayed on the screen (standard output).
   o **Output Preservation:** The exact same output displayed on the screen must also be written to a separate output file for testing and record-keeping purposes.

COBOL Implementation Details (For Programmers):

- **Profile Display Modules:** Develop or enhance COBOL modules responsible for retrieving a user's complete profile data from your persistent storage and formatting it for console output. Ensure all fields from Week 2 are displayed.
- **Search Logic:** Implement COBOL routines to read through your stored user account/profile data. This routine will compare user input against existing user names.
- **Data Retrieval:** Efficiently retrieve the entire profile record once a match is found during the search.

- **User Interaction:** Design clear prompts for the search functionality (e.g., "Enter the full name of the person you are looking for:"), ensuring these prompts are part of the duplicated output.
- **Input File Handling:** Continue to implement COBOL READ statements to read user input from the designated input file for all menu selections and search queries.
- **Output File Handling:** Ensure all program output, including displayed profiles and search results, is written to your dedicated output file *identically* to what is displayed on the screen.

Testing Responsibilities (For Testers):

- **Test Case Development:** Create comprehensive test cases in Jira for all Week 3 functionalities, including:
    - **Positive Test Cases:** Scenarios for successfully viewing one's own profile, and successfully searching for and finding existing users by their exact name.
    - **Negative Test Cases:** Scenarios for searching for non-existent users, or users with partial names.
    - **Edge Cases:** Consider searching for users with very long names, or names that are numerically similar to others.
- **Test Execution:** Execute all developed test cases using the specified input file.
- **Bug Reporting:** For every issue or discrepancy found, create a detailed bug ticket in Jira. Include steps to reproduce, actual results, and expected results.
- **Output Verification:** Meticulously compare the program's console output against the generated output file to ensure they are absolutely identical for all profile viewing and search scenarios.
- **Collaboration:** Work closely with the programmers to help them understand and reproduce bugs.

Jira Requirements (For Scrum Master, Programmers, & Testers):

Your team's Jira board for Week 3 should include:

- **Epic Updates:** Update the "User Profile Management" Epic to include User Stories for comprehensive profile viewing. Create a new Epic, **Epic #3: User Search & Connections (Part 1)**, to house the search functionality.
- **User Stories for Profile Viewing:**
    - "As a logged-in user, I want to view my complete saved profile, including all optional fields."
    - "As a logged-in user, I want my profile to be displayed in an easy-to-read format."
- **User Stories for Basic Search:**
    - "As a logged-in user, I want to search for other users by their full name."
    - "As a user performing a search, I want to see the profile of the user if a match is found."

- o "As a user performing a search, I want to be informed if no user matches my search query."
  - o "As a user, I want the 'Find someone you know' option to allow me to search for other users."
- **Updated User Story (for Testing):** "As a tester, I want the program to read all user inputs for profile viewing and search from a file so I can automate testing."
- **Updated User Story (for Testing):** "As a tester, I want the program to write all screen output related to profile viewing and search to a file so I can easily verify results."
- **Tasks:** Break down each User Story into granular tasks that individual team members can work on. (e.g., *Programmers:* "Develop COBOL routine to display all profile fields," "Implement COBOL search algorithm for exact name match," "Integrate search option into post-login menu," "Update input/output routines for profile viewing and search to use file I/O consistently." *Testers:* "Develop test cases for user search," "Execute profile viewing tests with various data sets," "Log bugs related to search functionality," "Verify I/O consistency for search features").
- **Bug Tickets:** Log any issues found during development and testing.

GitHub Requirements:

- **New Modules/Files:** Commit any new COBOL modules or updated existing files related to profile display and user search.
- **Branching:** Continue to follow your team's established branching strategy for new features.
- **Regular Commits:** Ensure consistent, descriptive commits throughout the week. Testers should commit their test files.
- **README.md Update:** Update your README.md to reflect the new functionality for profile viewing and search, explicitly noting how to prepare input for these features and where to find the corresponding output.

Deliverables for End of Week 3:

1. **Roles.txt**: List of team members and the roles that they played this week.
2. **InCollege.cob: Working COBOL Program:** A console-based COBOL application that seamlessly integrates profile viewing and the basic user search functionality with the existing login and profile creation features. All inputs must be read from a file, all outputs displayed on the screen, and the exact same outputs written to a separate file.
3. **InCollege-Input.txt: Sample Input File:** A sample text file demonstrating the format of input your program expects for Week 3's functionality (e.g., login credentials, menu choices, search queries).
4. **InCollege-Output.txt::** A sample text file showing the expected output for a typical run of your program, demonstrating both successful profile viewing and successful/unsuccessful searches.

--- SAMPLE_OUTPUT_WEEK3.TXT ---
Welcome to InCollege!
1. Log In
2. Create New Account
Enter your choice:
Please enter your username:
Please enter your password:
You have successfully logged in.
Welcome, TestUser!
. Create/Edit My Profile
2. View My Profile
3. Search for a job
4. Find someone you know
5. Learn a New Skill
Enter your choice:
--- Your Profile ---
Name: Test User
University: State University
Major: Computer Science
Graduation Year: 2025
About Me: Passionate about software development.
Experience:
  Title: Intern
  Company: Tech Solutions
  Dates: Summer 2024
  Description: Assisted with front-end development.
Education:
  Degree: Bachelor's
  University: State University
  Years: 2021-2025
----
1. Create/Edit My Profile
2. View My Profile
3. Search for a job
4. Find someone you know
5. Learn a New Skill
Enter your choice:
Enter the full name of the person you are looking for:
--- Found User Profile ---
Name: Another Student
University: City College
Major: Electrical Engineering
Graduation Year: 2026
About Me: Robotics enthusiast.
Experience: None
Education: None
----
1. Create/Edit My Profile
2. View My Profile
3. Search for a job
4. Find someone you know
5. Learn a New Skill
Enter your choice:

Enter the full name of the person you are looking for:
No one by that name could be found.
1. Create/Edit My Profile
2. View My Profile
3. Search for a job
4. Find someone you know
5. Learn a New Skill
Enter your choice:
--- END_OF_PROGRAM_EXECUTION ---

5. **Epic3-Storyx-Input.zip: Test Input Files:** A set of test input files used by the testers, covering positive, negative, and edge cases for each of this week's stories.
6. **Epic3-Storyx-Output.zip: Actual Test Output Files:** The exact output generated by running your program with the Epic3-Storyx-Input input Files, submitted for review.
7. **Jira.jpg: Updated Jira Board:** All relevant User Stories, tasks, and bugs (with their status) for Week 3 should be updated in Jira.
8. **GitHub.jpg:** Go to the repository's main page. Click the "Commits" link (next to the green "Code" button). Show a chronological list of all commits with messages, authors, and timestamps.

Your testers will be critical this week in verifying that profile data is displayed correctly, that the search function accurately finds existing users, and that it correctly handles cases where no user is found. They will also be meticulously comparing the console output with the generated output file to ensure perfect consistency. The scrum master's role in guiding the team through these new features and managing the sprint will be key to success.