What I did this week:

1. **Front-End Enhancements:**

• Expanded the AddEditEmployeeForm functionality to support both adding and editing employees.

• Implemented username auto-generation, ensuring usernames are unique (e.g., jperez01, jperez02) and cannot be manually edited.

• Added password features:

• Password reveal/hide toggle.

• Strong password generator with random special characters, digits, and uppercase letters.

• Real-time password strength advisor for user feedback.

• Populated fields correctly for editing, resolving issues with combo boxes failing to select appropriate items (e.g., position and site).

1. **Back-End Enhancements:**

• Built a secure PUT API endpoint for updating employees.

• Integrated password encryption using BCrypt for secure storage of passwords.

• Extended error handling to provide clear feedback for failed operations.

• Ensured seamless data flow between front-end forms and back-end API.

1. **Validation and Security:**

• Introduced strong field validation for emails, password rules, and mandatory fields before submission.

• Hardened the back-end to handle edge cases, including invalid IDs or permission levels during updates.

• Refined login flow, maintaining hashed password verification for authentication.

1. **Debugging and Refinement:**

• Fixed combo box selection issues by mapping objects to their equals() implementations for correct comparisons.

• Simplified and documented the HashUtil class, confirming password hashing is non-reversible for enhanced security.

What I plan to do next week:

1. **Finalize the employee management flow:**

• Ensure password handling during edits is secure and intuitive for users (e.g., leave unchanged unless explicitly updated).

• Allow for full CRUD functionality (mostly dealing with more uniform error handling) on employees via front-end forms.

1. **Extend the item management features, enabling warehouse managers to edit items through the application.**
2. **Begin implementing form aesthetics overhaul, considering improved wireframes if necessary.**

**4. Investigate and implement auditing features:**

• Logging login attempts and actions for compliance.

• Utilizing the txnaudit table appropriately for item tracking.

Any problems so far?

1. **Password Flow for Editing Employees:**

• Current design leaves the password blank during editing for security reasons. This requires further refinement to ensure secure handling and minimal disruption to user experience.

1. **Security Concerns:**

• While password encryption and validation are in place, there’s still no overarching security framework. Considering integrating Spring Security for more robust access control.

1. **Auditing Complexity:**

• Unsure how to best leverage the txnaudit table without cluttering it with unnecessary details vs using a different logging file for other actions. Need to decide which fields are relevant for logging and tracking user activity effectively.