**Step 1: Layout Rearrangement and Mobile Optimization**

* **Goal**: Rearrange the layout for better data entry and make it mobile-optimized with clear visual elements like larger buttons, a simple flow, and a project number & date selection.
* **Tasks**:
  + Move the project number selection and date picker to a **persistent header** so they are always visible.
  + Ensure all tabs (e.g., Workers, Equipment) are clearly labeled and have large, touch-friendly buttons.
  + Adjust CSS for mobile-friendliness (e.g., make all buttons bigger, simplify the layout for small screens).

**Step 2: Implement Tab Completion Indicators**

* **Goal**: Provide visual feedback when a user completes a section, using a darker blue color and a checkmark once a tab is completed.
* **Tasks**:
  + Create a global JavaScript state tracker to track whether each tab (e.g., Workers, Materials) has been completed.
  + After a user fills out a section and clicks the "Complete" button, update the tab’s visual state (change color, add a checkmark).
  + Disable the "Submit" button initially, and only enable it once all tabs are completed and a project number + date is selected.

**Step 3: Add Date Picker and Ensure All Critical Fields are Required**

* **Goal**: Ensure the user selects the correct date of the report along with the project number.
* **Tasks**:
  + Use an intuitive **date picker** for the date selection (in the header with the project number).
  + Ensure both project number and date are required fields and **validated before submission**.
  + Add validation logic in both the Flask backend and client-side JavaScript to ensure these fields are selected before proceeding.

**Step 4: Guided Workflow with Notifications**

* **Goal**: Make the user workflow simple and clear by guiding them through the process, including notifications for incomplete sections.
* **Tasks**:
  + Add JavaScript to **track tab openings**. When a user opens each tab (e.g., Workers, Materials), log it and display a notification that the section is in progress.
  + Once the section is marked as complete (via a "Complete Section" button in each tab), trigger a **visual change** and notify the user (e.g., "Workers added successfully").
  + Display a **progress bar** (either as a top indicator or percentage complete) so the user knows how far along they are.

**Step 5: Implement Conditional Display of Submit Button**

* **Goal**: Only display the submit button after all required sections have been completed and the project number & date are selected.
* **Tasks**:
  + Hide the "Submit" button by default.
  + Use JavaScript to conditionally show the "Submit" button only when:
    1. All tabs have been opened and completed.
    2. A valid project number and date are selected.
  + Add a **"Confirm Submission" modal** for a second confirmation step to prevent accidental submissions.

**Step 6: Implement Double Confirmation Before Submission**

* **Goal**: Introduce a two-step confirmation process to ensure data submission is intentional.
* **Tasks**:
  + When the user clicks "Submit", display a modal or overlay with a "Please confirm" message.
  + Add a second "Confirm Submission" button within this modal. Only after clicking this button will the data be saved to the Excel file.

**Step 7: Auto-Save and Save-for-Later**

* **Goal**: Add auto-save functionality to ensure users don't lose progress.
* **Tasks**:
  + Implement an **auto-save function** that saves data periodically (e.g., every 2 minutes) in the background to avoid data loss.
  + Provide a "Save and Continue Later" button that allows workers to save their progress without submitting.

**Step 8: Test & Refine Error Handling**

* **Goal**: Ensure users are aware of errors and missing inputs with clear notifications.
* **Tasks**:
  + Implement **error messages** when required fields are missing or incorrectly filled out (e.g., missing project number, missing date).
  + Display these errors both inline (next to the input fields) and as **notification pop-ups**.
  + Thoroughly test the form with different workflows to ensure all edge cases (e.g., leaving mid-entry, missing data, skipping sections) are handled gracefully.

**Step 9: Voice Input (Optional Enhancement for the Future)**

* **Goal**: Allow workers to use voice input for fields like "General Notes".
* **Tasks**:
  + Add **voice-to-text** functionality for large text areas, making it easy for workers to provide detailed notes.
  + Provide **clear instructions** and an easy-to-access button for enabling this feature.
  + Test thoroughly to ensure reliability in noisy environments (optional enhancement after main steps are implemented).

**Step-by-Step Execution Timeline**

1. **Day 1-3**: Layout optimization for mobile, header adjustments (Step 1).
2. **Day 4-5**: Implement tab completion indicators and state tracking (Step 2).
3. **Day 6-7**: Add date picker, project number validation, and ensure required fields (Step 3).
4. **Day 8-9**: Guided workflow and notifications implementation (Step 4).
5. **Day 10**: Conditional submit button display logic (Step 5).
6. **Day 11-12**: Two-step submission confirmation logic (Step 6).
7. **Day 13**: Auto-save and save-for-later functionality (Step 7).
8. **Day 14**: Error-handling and edge-case testing (Step 8).

By following these steps, you'll ensure a user-friendly experience with a clear workflow that encourages daily reporting without overwhelming site employees.

**Hotkeys:**

* W: Continue