

# HLD Output 7 | Qwen3:4b

Timesheet App

## High-Level Design (HLD) Document: TimeSheet App

### 1. Executive Summary

The TimeSheet app is a Power Apps solution designed to help users track and manage their time spent on various projects. The application allows users to enter time sheet records, including the date, project, hours worked, and comments. It also provides a daily view of time sheets for the current week, enabling users to review and manage their work hours efficiently.

The app is built using Power Apps with integration into Microsoft Dataverse (likely SharePoint or Dynamics 365) for data storage. The solution includes validation logic to ensure that hours entered are within acceptable ranges and that all required fields are populated before saving a time sheet record.

---

### 2. Application Architecture Overview

#### Key Modules

- **TimeSheet Entry Module:** Allows users to create, edit, and save time sheet records.
- **Daily View Module:** Displays a list of time sheets for the current week, filtered by selected dates.
- **Project Selection Module:** Enables users to select projects from a dropdown list.
- **Validation Module:** Ensures that hours entered are valid (between 0 and 24) and that all required fields are filled.

#### Screens

- **TimeSheet Entry Screen:** A form for entering time sheet data, including date, project, hours worked, and comments.

- **Daily View Screen:** Displays a gallery of time sheets for the selected week.

## Components

- **Date Pickers:** Allow users to select dates for the current week.
- **ComboBox:** Selects projects from a list of available projects.
- **TextInput Fields:** For entering hours worked and comments.
- **Gallery:** Displays a list of time sheet records with options to view or delete entries.
- **Buttons:** Save, Clear, and other action buttons.

## Data Sources

- **Timesheets (Dataverse Table):** Stores all time sheet records.
- **Projects (Dataverse Table):** Contains project data for the Project ComboBox.
- **User Context:** Current user is stored in `currentUser` variable to filter data by user.

# 3. Data Model

## Entities and Collections

Entity/Collection	Description
<b>Timesheets</b> (Dataverse Table)	Stores time sheet records, including DateWorked, Project, HoursWorked, Comments, and other metadata.
<b>Projects</b> (Dataverse Table)	Contains project names and details for the Project ComboBox.
<b>colTimesheets</b> (Collection)	A filtered collection of time sheets for the current user.
<b>colProjects</b> (Collection)	A collection of projects used to populate the Project ComboBox.

## Relationships

- **Timesheets** is related to **Projects** via a foreign key (likely `ProjectId` ).

- **colTimesheets** is derived from the **Timesheets** table, filtered by the current user.
  - **DailyView\_Gallery** filters time sheets based on the selected week dates.
- 

## 4. Navigation and User Flow

### Main User Journey

1. **Login/Authentication:** The app starts with the **OnStart** logic, which initializes the current user and loads project data.
2. **TimeSheet Entry Screen:**
  - User selects a date range (week) using date pickers.
  - User enters time sheet details: date, project, hours worked, comments.
  - Save button is enabled only if all fields are valid.
  - Time sheet record is saved to the **Timesheets** table and added to **colTimesheets**.
3. **Daily View Screen:**
  - Displays a gallery of time sheets for the selected week.
  - Each entry includes project, hours worked, comments, and a delete option.
  - User can view or delete individual time sheet records.

### Navigation Triggers

- **Save\_Button:** Triggers saving of a new time sheet record to the database.
  - **DateWorked\_DatePicker:** Sets the start date for the week, which filters the gallery.
  - **DailyView\_Gallery:** Displays time sheets based on selected dates and user context.
- 

## 5. Custom Logic and Key Interactions

### Key Formulas/Triggers

- **OnStart:**

- Initializes `currentUser` .
- Filters `colTimesheets` by current user.
- Loads project data into `colProjects` .
- **HoursWorked\_TextInput.OnChange:**
  - Validates that hours are between 0 and 24.
  - Sets `varHoursEntryValidation` to true if valid, otherwise false.
- **Save\_Button.OnSelect:**
  - Triggers a Patch function to save the time sheet record to the **Timesheets** table.
  - Clears and refreshes `colTimesheets` .
- **DailyView\_Gallery.OnSelect:**
  - Displays details of a selected time sheet (project, hours worked, comments).
  - Provides a delete option for the selected record.

## Business Rules

- Hours must be between 0 and 24.
  - All required fields (date, project, hours) must be filled before saving.
  - Time sheets are filtered by current user in `colTimesheets` .
- 

## 6. Integration Points

- **Dataverse (Microsoft Dynamics 365/SharePoint):** Used for storing time sheet records and project data.
  - **User Context:** Current user is used to filter data and ensure that only the user's time sheets are displayed.
- 

## 7. Special Features or Business Rules

- **Validation Logic:**
  - Hours must be between 0 and 24.

- Required fields (date, project, hours) must be filled before saving a record.
- **User-Specific Data:**
  - Time sheets are filtered by the current user to ensure data privacy and accuracy.
- **Weekly View:**
  - Users can select a week to view time sheets for that period.

## 8. Component Summary Table

Component	Description
<b>TimeSheet Entry Screen</b>	Form for entering time sheet records with date, project, hours worked, and comments.
<b>DailyView_Gallery</b>	Displays a list of time sheets for the selected week, with options to view or delete entries.
<b>DateWorked_DatePicker</b>	Allows selection of the start date for the week.
<b>Project_ComboBox</b>	Dropdown to select a project from a list of available projects.
<b>HoursWorked_TextInput</b>	Input field for entering hours worked, with validation logic.
<b>Comments_TextInput</b>	Multi-line text input for comments on the time sheet.
<b>Save_Button</b>	Saves the time sheet record if all fields are valid.
<b>Rectangle2_1, Rectangle1, Label1</b>	UI components for layout and title display.

## Conclusion

This TimeSheet app is a well-structured Power Apps solution that enables users to efficiently track their work hours across multiple projects. The application includes validation logic, user-specific data filtering, and a daily view of time sheets, making it a robust tool for time management in a business environment. The architecture is modular, with clear separation of concerns between the entry screen, gallery view, and data storage layers.