**Project Title :** **AI-Powered Progress Tracker Screen**

**Objective:**  
Build an interactive progress tracking screen that shows a user’s daily or course progress, powered by AI-generated insights and smooth animations.

**Name:** Pooja S

**Date:** 24-10-2025

**1. Abstract**

The AI-Powered Progress Tracker Screen is designed to help users **track their daily tasks in an interactive and visual way**. Each completed task adds a **floor to a virtual house**, providing a fun and motivating way to monitor progress. AI-generated messages give encouragement, while animations and interactive elements enhance the user experience.

## ****2. Features****

* **Add Tasks:** Users can input daily tasks into a dedicated field.
* **Task List Display:** Tasks are displayed in a list format; completed tasks are visually marked.
* **Dynamic House Building:** Completing a task adds a colored floor block to a virtual house.
* **Windows with Blinking Lights:** Each floor has windows that blink, simulating life in the house.
* **Interactive Floors:** Clicking a floor removes it and updates the progress and task status.
* **AI Suggestions:** Random motivational messages appear after each task completion.
* **Progress Bar:** Displays the percentage of tasks completed relative to total tasks.
* **Smooth Animations:** Floors appear with a falling animation; hover effects enhance interaction.

## ****3. Technology Stack****

* **HTML5** – Structure and layout of the page.
* **CSS3** – Styling, animations, responsive design, and interactive visual effects.
* **JavaScript (ES6)** – Dynamic task management, building/removing floors, AI messages, and progress tracking.

## ****4. How it Works****

1. **Adding Tasks:** Users enter a task and click “Add Task”; the task is added to the task list.
2. **Completing Tasks:** Clicking on a task marks it completed, adds a floor block to the house, and generates a motivational AI message.
3. **Building Floors:**
   * Floors are **rectangular and coloured.**
   * Two windows are added to each floor with **blinking animation**.
   * Floors are stacked in a vertical column, simulating a building.
4. **Removing Floors:** Clicking a floor removes it, updates the task status, AI message, and progress bar.
5. **AI Messages:** Provides encouragement like “Great job!” after task completion.
6. **Progress Bar:** Calculates the percentage of completed tasks and fills the progress bar accordingly.

## ****5. Future Enhancements****

* Enable **editing or deleting tasks**.
* Store tasks in **local storage or cloud database** to persist progress.
* Add **custom house themes and floor colors**.
* Include **AI suggestions for prioritizing tasks**.
* Enhance **mobile responsiveness** for smaller screens.

## ****6. Conclusion****

The project successfully demonstrates an **interactive, AI-powered progress tracker**. Users can visualize their daily accomplishments through **house building**, while **motivational AI messages** encourage continued task completion. Animations, progress bars, and blinking windows make tracking tasks **fun, interactive, and engaging**.

**Appendix A – Code**

**HTML (index.html)**

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Progress Tracker!!</title>

  <link rel="stylesheet" href="style.css">

</head>

<body>

  <header>

    <h1>🏠 My Daily Progress House</h1>

    <p>Complete tasks to build your house floor by floor!</p>

  </header>

  <div class="main-container">

    <!-- Left: Task Section -->

    <div class="task-section-wrapper">

      <div class="task-section">

        <input type="text" id="taskInput" placeholder="Enter the tasks of your day">

        <button id="addTaskBtn">Add Task</button>

      </div>

      <div class="task-list">

        <h3>Today's Tasks</h3>

        <ul id="taskList"></ul>

      </div>

      <div id="aiMessage" class="ai-message">

        🌱 Start building your house by completing a task!

      </div>

    </div>

    <!-- Right: House Area -->

    <div class="house-section">

      <!-- Progress Bar -->

      <div class="progress-container">

        <div class="progress-bar" id="progressBar"></div>

      </div>

      <!-- House Floors -->

      <div class="house-area">

        <div id="house"></div>

      </div>

    </div>

  </div>

  <script src="script.js"></script>

</body>

</html>

**CSS (style.css)**

\* {

  margin: 0;

  padding: 0;

  box-sizing: border-box;

  font-family: "Poppins", sans-serif;

}

body {

  background: url("HD-wallpaper-anime-landscape-building-sunset-clouds-scenic-anime.jpg") no-repeat center center/cover;

  min-height: 100vh;

  display: flex;

  flex-direction: column;

  align-items: center;

  justify-content: center;

  color: #fff;

  text-align: center;

}

header {

  margin-bottom: 20px;

}

header h1 {

  font-size: 2.5em;

  color: #ffe066;

  text-shadow: 2px 2px 8px rgba(0,0,0,0.5);

}

header p {

  font-size: 1.2em;

  color: #f0f0f0;

}

.main-container {

  display: flex;

  justify-content: center;

  align-items: flex-start;

  gap: 30px;

  width: 90%;

  max-width: 1000px;

}

.task-section-wrapper {

  flex: 1;

  display: flex;

  flex-direction: column;

  align-items: center;

}

.task-section {

  display: flex;

  gap: 10px;

  margin-bottom: 20px;

}

#taskInput {

  flex: 1;

  padding: 12px;

  font-size: 1em;

  border-radius: 8px;

  border: none;

}

button {

  padding: 12px 18px;

  background: #4caf50;

  color: #fff;

  border: none;

  border-radius: 8px;

  font-weight: bold;

  cursor: pointer;

  transition: 0.3s;

}

button:hover {

  background: #45a049;

  transform: scale(1.05);

}

.task-list {

  width: 100%;

  background: rgba(0,0,0,0.5);

  border-radius: 12px;

  padding: 12px;

}

.task-list h3 {

  color: #ffd369;

  margin-bottom: 10px;

}

.task-list ul {

  list-style: none;

}

.task-list li {

  background: rgba(255,255,255,0.2);

  padding: 10px;

  margin: 6px 0;

  border-radius: 8px;

  color: #fff;

  cursor: pointer;

  font-size: 1em;

}

.ai-message {

  margin-top: 15px;

  background: rgba(255,255,255,0.2);

  padding: 12px;

  border-radius: 12px;

  color: #fff;

  font-size: 1em;

  font-style: italic;

  min-height: 45px;

}

.house-section {

  flex: 1;

  display: flex;

  flex-direction: column;

  align-items: center;

}

.progress-container {

  width: 80%;

  height: 20px;

  background: rgba(255,255,255,0.2);

  border-radius: 12px;

  margin-bottom: 12px;

  overflow: hidden;

}

.progress-bar {

  width: 0%;

  height: 100%;

  background: #13d619;

  transition: width 0.5s ease;

  border-radius: 12px;

}

.house-area {

  width: 70%;

  max-width: 500px;

  height: 600px;

  display: flex;

  flex-direction: column-reverse;

  align-items: center;

  background: rgba(0,0,0,0.3);

  border-radius: 12px;

  padding: 5px 0;

  overflow-y: auto;

}

.floor {

  width: 95%;

  height: 70px;

  margin: 5px auto;

  border-radius: 10px;

  display: flex;

  justify-content: space-between;

  align-items: center;

  padding: 0 20px;

  box-shadow: 2px 2px 12px rgba(0,0,0,0.5);

  cursor: pointer;

  animation: fall 0.5s ease forwards;

}

.window {

  width: 25px;

  height: 25px;

  background: #fff;

  border-radius: 4px;

  border: 2px solid #333;

  margin-top: 25px;

  animation: blink 1.5s infinite;

}

.floor:hover {

  transform: translateY(-5px) scale(1.02);

  box-shadow: 4px 4px 15px rgba(0,0,0,0.7);

  transition: 0.3s;

}

/\* Animations \*/

@keyframes fall {

  from { transform: translateY(-120px); opacity: 0; }

  to { transform: translateY(0); opacity: 1; }

}

@keyframes blink {

  0%, 50%, 100% { background: #fff; }

  25%, 75% { background: #ffe066; }

}

**JavaScript (script.js)**

document.addEventListener("DOMContentLoaded", () => {

  const taskInput = document.getElementById("taskInput");

  const addTaskBtn = document.getElementById("addTaskBtn");

  const taskList = document.getElementById("taskList");

  const house = document.getElementById("house");

  const aiMessage = document.getElementById("aiMessage");

  const progressBar = document.getElementById("progressBar");

  let tasks = [];

  // Add task

  addTaskBtn.addEventListener("click", () => {

    const taskText = taskInput.value.trim();

    if(taskText === "") { alert("Enter a task!"); return; }

    tasks.push({ text: taskText, completed: false });

    displayTasks();

    taskInput.value = "";

    updateProgress();

  });

  function displayTasks() {

    taskList.innerHTML = "";

    tasks.forEach((task, index) => {

      const li = document.createElement("li");

      li.textContent = task.text;

      if(task.completed){

        li.style.textDecoration = "line-through";

        li.style.color = "#a1ffa1";

      }

      li.addEventListener("click", () => completeTask(index));

      taskList.appendChild(li);

    });

  }

  function completeTask(index){

    if(tasks[index].completed){

      aiMessage.innerHTML = "✅ Task already completed!";

      return;

    }

    tasks[index].completed = true;

    displayTasks();

    buildFloor(tasks[index].text);

    generateAISuggestion(tasks[index].text);

    updateProgress();

  }

  function buildFloor(taskName){

    const floor = document.createElement("div");

    floor.classList.add("floor");

    // Random color

    const colors = ["#ff8c66","#66b3ff","#ffcc66","#8aff66","#ff66b3"];

    floor.style.background = colors[Math.floor(Math.random()\*colors.length)];

    // Windows

    const leftWindow = document.createElement("div");

    leftWindow.classList.add("window");

    const rightWindow = document.createElement("div");

    rightWindow.classList.add("window");

    floor.appendChild(leftWindow);

    floor.appendChild(rightWindow);

    floor.title = `🏗️ ${taskName} completed`;

    // Click to remove floor

    floor.addEventListener("click", () => {

      house.removeChild(floor);

      aiMessage.innerHTML = `🛠️ You removed "${taskName}".`;

      const task = tasks.find(t => t.text === taskName);

      if(task) task.completed = false;

      displayTasks();

      updateProgress();

    });

    house.appendChild(floor);

    house.scrollIntoView({ behavior: "smooth", block: "end" });

  }

  function generateAISuggestion(taskName){

    const messages = [

      `🌟 Great job completing "${taskName}"! Your house grows taller!`,

      `💪 "${taskName}" done! Keep it up!`,

      `🏠 "${taskName}" completed! Your home is growing!`,

      `🔥 Fantastic! "${taskName}" done!`,

      `🌈 Step by step, your house of dreams grows!`

    ];

    const randomMsg = messages[Math.floor(Math.random() \* messages.length)];

    aiMessage.innerHTML = randomMsg;

  }

  function updateProgress(){

    const total = tasks.length;

    const completed = tasks.filter(t=>t.completed).length;

    const percent = total===0 ? 0 : Math.round((completed/total)\*100);

    progressBar.style.width = percent + "%";

  }

});

**Appendix B – Output Screenshots**









