### 1. ****Project Proposal****

**What to do:**

* Write an overview of your project.
* Clearly state **the problem your project solves or the need it addresses**.
* Include the **objectives** (what you want to achieve).
* Define the **project scope** (what’s included and what’s not).

**Example Content:**

* **Project Name:** Smart IoT Control System
* **Objective:** Develop a web-based IoT platform for remote device control and simulation.
* **Scope:** Includes user login system, IoT device control interface, data monitoring, and simulation tools. Excludes hardware manufacturing.

### 2. ****Project Plan****

**What to do:**

* Create a **Gantt chart** to show the project timeline.
* Identify **key milestones** (important deadlines).
* List **deliverables** (what you’ll submit/deliver at each stage).
* Detail **resource allocation** (who works on what and needed tools/resources).

**Tools you can use:** Excel, MS Project, Trello, or GanttProject.

**Example Content:**

* **Milestones:**
  + Week 1–2: Requirements gathering
  + Week 3–4: UI/UX design
  + Week 5–6: Backend development
  + Week 7: Testing
  + Week 8: Final report & presentation

### 3. ****Task Assignment & Roles****

**What to do:**

* Assign **specific tasks** to each team member.
* Mention **roles/responsibilities** clearly (project manager, developer, designer, tester, etc.)

**Example Table:**

| **Name** | **Role** | **Tasks** |
| --- | --- | --- |
| You | Project Manager | Planning, report writing, coordination |
| Member A | Backend Developer | API & database development |
| Member B | Frontend Developer | UI design, client-side logic |
| Member C | QA Tester | Testing, bug reporting |

### 4. ****Risk Assessment & Mitigation Plan****

**What to do:**

* List potential **risks** (technical, time, resource, scope creep, etc.).
* For each risk, suggest a **mitigation strategy** (what you’ll do to prevent or solve it).

**Example Table:**

| **Risk** | **Impact** | **Mitigation Strategy** |
| --- | --- | --- |
| Delay in development | High | Buffer time in timeline, weekly reviews |
| Team member unavailability | Medium | Cross-training, backup resource allocation |
| Hardware or software incompatibility | High | Early testing, use widely supported platforms |
| Scope creep | Medium | Strict scope control and documentation |

### 5. ****KPIs (Key Performance Indicators)****

**What to do:**

* Identify **metrics** that define project success.
* Choose measurable, relevant indicators based on your project type.

**Examples for a software/IoT project:**

* **System Uptime** – Target: 99%
* **Response Time** – Target: < 2 seconds
* **User Adoption Rate** – Target: 80% of test users use it regularly
* **Bug Rate** – Target: < 5 bugs per 1000 lines of code
* **Feature Completion Rate** – Target: 100% by deadline