

GANTT CHART

1. List the tasks (modules)

We started by writing down all the parts of the system that need to be built, based on the DFD.
Example: Database, Landing Page, Login Page, Cart, Admin Dashboard, etc.

Think of this like writing your **to-do list** for the project.

2. Group tasks into RAD phases

Since we are using **RAD (Rapid Application Development)**, we follow its stages:

1. **Requirements Planning** – gather needs.
2. **User Design** – build simple prototypes (sketches, wireframes).
3. **Construction** – build the actual system.
4. **Cutover** – test and deploy.

Place each task under the phase where it belongs.

3. Estimate time for each task

For every task, we estimate how long it would take in **days**.

- Small pages → 1–3 days
- Bigger things (like database, admin dashboard) → 2–4 days

These are **estimates**, not exact. The idea is to plan ahead.

4. Identify dependencies (what comes first)

Some tasks **depend on others**.

- Example: You need the **Database** before you can make the **Sign-Up Page**.
- You need **Sign-Up** done before **Login** makes sense.

Make sure tasks are in the correct order.

5. Plan parallel work (teamwork)

Since we have **3-5 members**, some tasks can happen at the same time.

- Example: While one member is building the **Database**, another can start designing the **Landing Page**.

👉 This makes the project faster.

6. Assign people

We assigned **all 5 members** to each task (so it's flexible who does what).

In real life, you might say:

- 2 people → database
- 2 people → frontend pages
- 1 person → testing

7. Put it in a timeline

Finally, we placed all tasks into a **calendar view**:

- Start date → Sept 29, 2025
- End date → calculated based on task durations and dependencies
- Tasks show as **bars** in the Gantt chart

👉 This way, you can see **which tasks overlap** and when the project will finish.

Guide for Individual Members

Scenario: We have 5 members:

Ariel, Bea, Carlos, Dante, Elaine

Roles (special focus areas)

Even if all 5 can help each other, it's easier if each has a **main role**:

- **Ariel – Database Specialist**
Focus on database design, schema, and connections.
- **Bea – Frontend Designer**
Focus on how pages look (landing, sign-up, login, cart, etc.).
- **Carlos – Backend Developer**
Focus on how the system works behind the scenes (PHP logic, APIs).
- **Dante – Tester & Integrator**
Focus on testing modules and making sure they work together.
- **Elaine – Project Coordinator**
Helps manage schedules, checks progress, and supports where needed.

. Task Distribution (per phase)

Phase 1: Requirements Planning

- **All Members:** Meet, list requirements, and agree on features.
 Everyone contributes here.

Phase 2: User Design (Prototypes)

- **Bea:** Makes wireframes (Landing Page, Sign-Up, Login, Cart).

- **Elaine:** Helps gather feedback from the group and adjust designs.
👉 Others just review and give feedback.
-

📌 Phase 3: Construction (Main Build)

Here's how tasks can be divided (some in parallel):

| Task/Module | Main Person | Support Member |
|------------------------------|-------------|----------------|
| Database Schema & Setup | Ariel | Carlos |
| Landing Page Interface | Bea | Elaine |
| Sign-Up Page | Carlos | Bea |
| Login Page | Carlos | Ariel |
| Client Landing (after login) | Bea | Elaine |
| Cart & Place Ordering | Ariel | Carlos |
| Track Order Page | Carlos | Dante |
| Order History Page | Ariel | Dante |
| Admin Dashboard | Bea | Elaine |
| Sales Summary Page | Ariel | Bea |
| Managing Items Page | Carlos | Ariel |
| Managing Orders Page | Dante | Elaine |

Phase 4: Cutover (Testing & Deployment)

- **Dante:** Lead tester (checks each module, finds bugs).
- **Elaine:** Coordinates testing schedule, documents results.
- **All Members:** Fix bugs together, then prepare for deployment.

Conclusive Thoughts

Work in parallel: More than one task can happen at the same time.

Communicate often: Update the group daily (short meeting).

Support each other: Even if you have a main role, help teammates when free.

Test early: Don't wait until the end—test as soon as a page is ready.