5-NEXT SPRINT — STANDARD EXECUTION PROMPT

5-NEXT SPRINT — STANDARD EXECUTION PROMPT

🧠 CONTEXT

You are inside an ongoing product sprint execution loop.

The product's core documentation lives in the `/project-docs/` folder. All sprints are tracked in `/project-docs/sprints.md` .

You've already completed one or more sprints. The documentation, codebase, and priorities are up to date.

✅ OBJECTIVE

Pick up and execute the **next sprint** from `/project-docs/sprints.md` .

Follow the **Sprint Execution Loop**, **Ten Commandments**, and **Documentation Hygiene Protocol** with full discipline.

☐ DO THIS:

1. SELECT SPRINT

- Open `/project-docs/sprints.md` .
- Find the first sprint marked as 🟡 `In Progress` or ⚪ `Not Started` .
- Lock in that sprint as your next task list.

2. REVIEW CONTEXT

- Re-read `/project-docs/plan.md` , `bugs-and-issues.md` , and `ui-ux-gaps.md` .
- Open any relevant files or folders that relate to this sprint (e.g. `src/auth/` , `components/` , `api/` , `utils/` , etc.).
- If the sprint has tags (`#auth` , `#ui` , `#api` , etc.) — check the corresponding systems.

3. IMPLEMENT TASKS

- Build every item in the sprint scope (no placeholders).
- Clean up console, TS warnings, and side effects.
- Run `[🧪 TEST]` blocks as you go.
- Preserve all existing functionality unless explicitly updating it.

4. UPDATE DOCUMENTATION

Update all relevant files:

- `sprints.md` → mark this sprint as ✅ Complete
- `plan.md` → remove done tasks or add new ones that emerged
- `bugs-and-issues.md` → close solved issues or log new ones
- `ui-ux-gaps.md` → log any UX tweaks or design debt solved
- `incomplete-features.md` → log features you intentionally deferred

Also update:

- `data-integrations.md` or `database-setup.md` if schema or API logic changed
- `auth-system.md` for login/auth changes
- `config-and-env.md` if you added new scripts, env vars, build changes

5. SUBMIT SPRINT REPORT

When the sprint is finished, output this summary in Markdown:

```
## ✅ Sprint [X] Complete

### Sprint Tasks Completed
- [x] [task name + short tag: #api, #auth, etc.]
- [x] ...
- [x] ...

### Testing + Verification
- ✅ [🧪 TEST] List of critical tests passed
- ✅ Console clean, linter clean
- ✅ No regressions introduced

### Docs Updated
- ✅ `sprints.md`
- ✅ `plan.md`
- ✅ `bugs-and-issues.md`
```

```
    – ✅ `...` (any others)


### Notes
– [Edge cases handled?]
– [Future ideas parked in incomplete-features.md?]
– [New blockers or follow-up tasks?]
```

6. WAIT FOR USER

Pause and wait for confirmation: `"next"` or `"next sprint please"` before continuing.

📜 REMINDERS

- 🧠 Don't rush. Quality > Speed.
- 📚 All code must be documented. All changes must be logged.
- 🧪 All features must be tested.
- 🛑 Never proceed without user's go-ahead.

🌀 FLOW RECAP

```
START →
  Read sprints.md →
  Lock next sprint →
  Review plan.md + code →
  Implement clean, test-ready logic →
  Update all docs →
  Submit Markdown report →
  WAIT →
  "Next" →
  Repeat
```

This keeps your agent in perfect sync with the build rhythm, while keeping Sifeddine in the captain's chair as tester and strategist.