**OVERVIEW: Agile Setup for AutoZone Dashboard**

* **Tool:** Jira Software (Cloud)
* **Framework:** Scrum (Sprints, Backlog, Story Points)
* **Team:** Solo (You) — Simulate team collaboration
* **Sprints:** 4 Sprints (each ~1 week)

**✅ Step-by-Step Jira Setup**

**✅ 1. Create a Jira Account**

Go to: https://www.atlassian.com/software/jira

* Click **Get it free**
* Choose **Jira Software**
* Select **Scrum Project**
* Name it AutoZone Dashboard

**✅ 2. Setup Scrum Board**

* Create a **Scrum board** (select during project creation)
* Navigate to **Backlog** tab
* You’ll see:
  + **Backlog (unplanned tasks)**
  + **Sprint 1 (planned tasks)**

**✅ 3. Create Epics (High-Level Features)**

Go to **Backlog → Epics Panel → Create Epic**:

| **Epic Name** | **Description** |
| --- | --- |
| Project Setup | Vite, Tailwind, Git |
| Authentication | Login, role-based auth |
| User Dashboard | View cars, book services |
| Company Dashboard | Manage cars, parts, services |
| Admin Dashboard | Admin-level CRUD + analytics |
| UI & Responsiveness | Tailwind styling + responsive design |
| Deployment & Docs | GitHub, hosting, documentation |

**✅ 4. Add User Stories (Tasks) Under Each Epic**

Example under “User Dashboard”:

* ✅ View car listings
* ✅ Filter/search cars
* ✅ Book a service
* ✅ View booking history

Assign **Story Points** (Effort Estimation):

| **Story** | **Story Points** |
| --- | --- |
| Setup login form | 2 |
| Create car listing UI | 3 |
| Integrate API for services | 5 |
| Add role-based navigation | 3 |

🎯 Use Fibonacci scale (1, 2, 3, 5, 8...)

**✅ 5. Sprint Planning**

| **Sprint** | **Focus** | **Estimated Story Points** |
| --- | --- | --- |
| Sprint 1 | Setup, Routing, Auth | 8 |
| Sprint 2 | User Dashboard | 13 |
| Sprint 3 | Company & Admin Dashboards | 15 |
| Sprint 4 | Polish UI, Testing, Deployment | 10 |

Each Sprint is **5-7 days**. Adjust based on your availability.

**📆 Suggested Sprint Schedule**

| **Sprint** | **Tasks** | **Story Points** |
| --- | --- | --- |
| **Sprint 1: Setup + Auth (Day 1–3)** | <ul><li>Vite + Tailwind Setup</li><li>Folder Structure</li><li>Routing Setup</li><li>Mock Login Page</li><li>Role-based Route Protection</li></ul> | 8 |
| **Sprint 2: User Dashboard (Day 4–7)** | <ul><li>View Cars</li><li>Book Services</li><li>Track Status</li><li>Axios Integration</li><li>Basic Zustand Store</li></ul> | 13 |
| **Sprint 3: Admin + Company (Week 2)** | <ul><li>Company CRUD for Cars, Parts, Services</li><li>Admin – View All Bookings</li><li>Delete/Edit Options</li><li>Service Approval</li></ul> | 15 |
| **Sprint 4: UI Polish + Deployment (Week 3)** | <ul><li>Responsive Design</li><li>Error Handling</li><li>Login Validation</li><li>GitHub Push</li><li>Deploy on Vercel/Netlify</li></ul> | 10 |

**🧠 Best Practices in Jira**

* Use **Labels**: frontend, auth, bug, API
* Use **Checklists** in stories (Break into subtasks)
* **Drag & Drop** tasks into Sprint
* Use **Burndown chart** to track progress

**✅ Final Deliverables (After 4 Sprints)**

* Clean, responsive dashboard
* Role-based routes with CRUD logic
* JSON backend connected
* Project documented in Git
* Deployed live + GitHub repo link
* Sprint reports + screenshots (for interview)