Phase 6: User Interface Development

👉 **Goal:** Build a user-friendly interface in Salesforce to simplify service request creation, tracking, and management.

---

1. Lightning Record Pages

- Create a **custom record page** for the **Service Request** object.

- Use the **Lightning App Builder** to organize fields into sections (Request Details, Assignment, Resolution).

- Add related lists like **Contacts** and **Accounts** for quick reference.

---

2. Quick Actions

- Add global and object-specific quick actions:

  - **Log Service Request** → allows staff to quickly create a new request.

  - **Reassign Staff** → lets managers reassign a request in fewer clicks.

  - **Close Request** → updates status and prompts for resolution notes.

---

3. Dynamic Forms

- Configure **Dynamic Forms** on Service Request pages.

- Show or hide fields based on conditions:

  - If Status = "Closed", then display **Resolution Notes**.

  - If Request Type = "IT", show **Technical Details** section.

4. App Navigation

- Create a dedicated app called **Smart Service Request**.

- Add navigation tabs for **Accounts, Contacts, Service Requests, Reports, and Dashboards**.

- Keep navigation simple so staff can move between objects without confusion.

5. Related Lists & Components

- On the **Contact page**, display a related list of Service Requests raised by that contact.

- On the **Account page**, display all Service Requests linked to that account.

- Use components like **Highlights Panel** to show key fields (Request ID, Priority, Status).

6. UI Testing

- Test layouts in both **desktop and Salesforce Mobile App**.

- Ensure that **compact layouts** display the most important information (Request ID, Priority, Status, Assigned Staff).

- Collect feedback from staff and managers to refine the interface.

✅ With this setup, Smart Service Request System provides an **intuitive and responsive user interface**, ensuring staff, managers, and customers can access and manage requests efficiently.