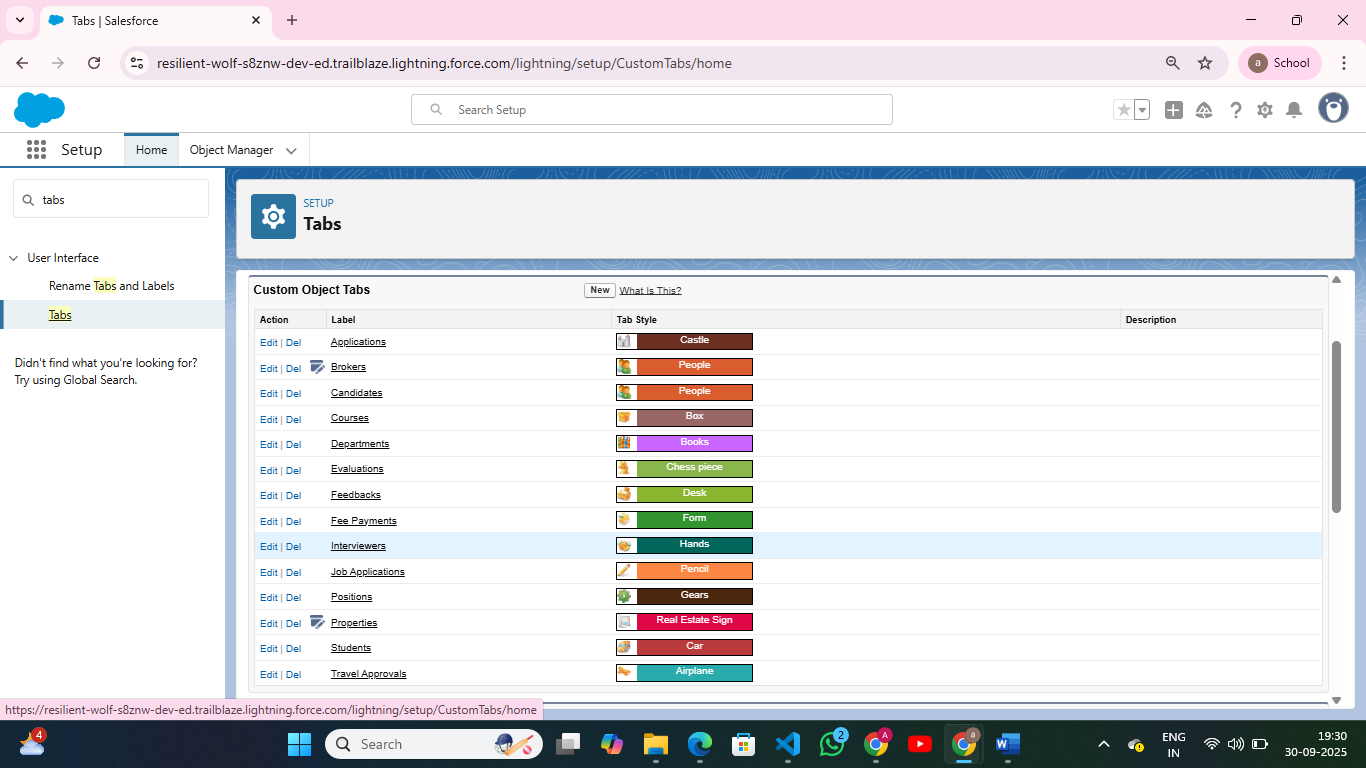
**Phase 6: User Interface Development**

**1. Lightning App Builder**

* Build custom Lightning Pages for apps, record pages, and home pages.
* **Navigation**: Setup → Lightning App Builder → New → Select Page Type → Add Components → Save & Activate.

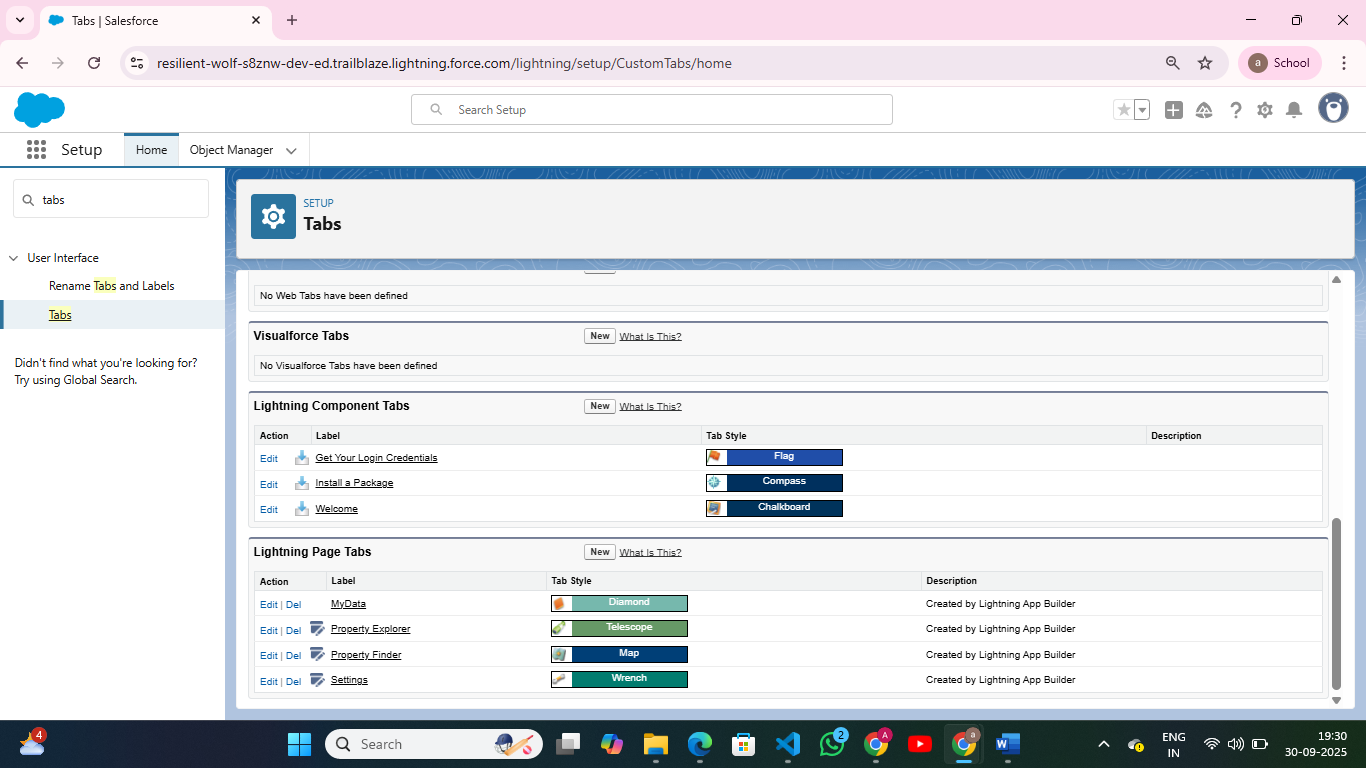


**2. Record Pages**

* Customize record detail layout for objects.
* **Navigation**: Setup → Lightning App Builder → Record Page → Drag Components (Related Lists, Tabs, Custom Components).

**3. Tabs**

* Add standard or custom tabs to Lightning apps.
* **Navigation**: Setup → Tabs → New → Select Object or Visualforce/LWC → Add to App.



**4. Home Page Layouts**

* Customize user home page with key metrics, reports, and components.
* **Navigation**: Setup → Lightning App Builder → Home Page → Add Components → Save & Assign to Profiles.

**5. Utility Bar**

* Add quick access tools (e.g., Recent Records, Notifications).
* **Navigation**: Setup → App Manager → Edit App → Utility Bar → Add Components.

**6. LWC (Lightning Web Components)**

* Build custom UI components using modern JavaScript and Salesforce APIs.
* **Navigation**: VS Code → SFDX: Create Lightning Web Component → Deploy to Org → Add to App/Page.

**7. Apex with LWC**

* Use Apex methods in LWC for server-side logic.
* **Navigation**: Create Apex Class → @AuraEnabled methods → Import and call in LWC JS controller.

**8. Events in LWC**

* Handle inter-component communication.
* **Navigation**: LWC JS → Use CustomEvent for child → parent communication.

**Imperative Apex Calls**

* Call Apex methods on user action (button click).
* **Navigation**: LWC JS → Import Apex → Call method inside JS function → Handle Promise .then() / .catch().