Phase 6: User Interface Development

6.1 Lightning App Builder Configuration

6.1.1 Create Custom App

- **App Label:** Clinic Management
- **Developer Name:** Clinic_Management
- **Description:** Complete clinic management system for healthcare providers
- **Navigation Style:** Console
- **Utility Items:** Notes, History, Recent Items
- **Navigation Items:** Home, Patients (Contacts), Doctors, Appointments, Medical Cases, Bills, Reports, Dashboards

6.1.2 Custom Record Pages

- **Patient (Contact) Record Page**
  Template: Record Page (Clone from Standard Contact Record Page)
  Sections:
  - Patient Information → Name, Email, Phone, Birthdate, Address, Emergency Contact, Insurance
  - Medical Summary → Recent Appointments, Active Medical Cases, Outstanding Bills
    Sidebar: Activity Timeline, Quick Actions (New Appointment, New Case, Send Message), Files and Notes
- **Appointment Record Page**
  Template: Record Page
  Sections:
  - Appointment Details → Patient, Doctor, Date, Time, Status, Type, Duration, Reason for Visit
  - Medical Information → Related Medical Case, Notes, Instructions, Follow-ups
  - Related Records → Bills, Tasks, Files
    Sidebar: Path (Status), Activity Timeline, Quick Actions (Complete, Reschedule, Cancel, Create Case)

6.2 Lightning Web Components (LWC)

6.2.1 Doctor Availability Calendar Component

Purpose: Allow staff to view doctor availability, search patients, and book appointments directly.

- **Files:**
  doctorAvailabilityCalendar.html, doctorAvailabilityCalendar.js, doctorAvailabilityCalendar.css, doctorAvailabilityCalendar.js-meta.xml
- **Features:**
  - Select doctor from combobox (active doctors only)
  - Select date (future only)
  - Display available time slots with status indicators
  - Search patients by name or email
  - Booking form with Appointment Type & Reason for Visit
  - Toast messages for success/error feedback
- **Key UI Behaviors:**
  - Available slots → displayed as buttons (success = available, neutral = booked, brand = selected)
  - Booking form shows only after selecting a time slot
  - Appointment booking triggers Apex `createAppointment` and refreshes slots

---

6.3 Apex Controller (DoctorAvailabilityController)

Purpose: Backend logic for LWC to fetch doctors, check slots, search patients, and create appointments.

- **Methods (exposed to LWC):**
  - `getDoctors()` → fetch active doctors with specialization
  - `getAvailableSlots(doctorId, selectedDate)` → return available times
  - `searchPatients(searchTerm)` → return patients matching name/email
  - `createAppointment(appointmentData)` → create new appointment record

This is My project LMC code (In preogress)

1) **Appointment**

fileUploadExample.html

```
<template>
    <lightning-card title="Upload Medical Report" icon-
name="custom:custom63">
        <div class="slds-m-around_medium">
            <lightning-file-upload
                label="Attach Files"
                name="fileUploader"
                accept=".pdf,.png,.jpg,.jpeg"
                record-id={recordId}
                onuploadfinished={handleUploadFinished}
                multiple>
            </lightning-file-upload>
        </div>
    </lightning-card>
</template>
```

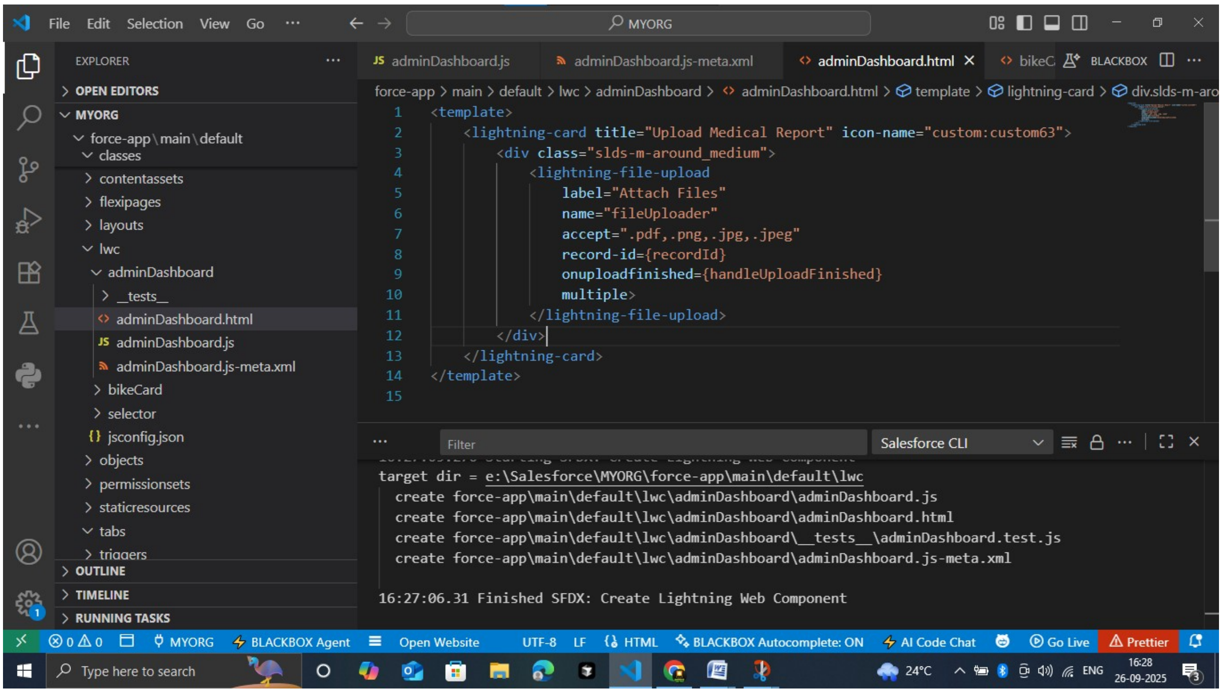
fileUploadExample.js

```
import { LightningElement, api } from 'lwc';
import { ShowToastEvent } from 'lightning/platformShowToastEvent';

export default class FileUploadExample extends LightningElement {
    @api recordId; // Attach files to a record, e.g., Patient__c or
Appointment__c

    handleUploadFinished(event) {
        // Get uploaded file details
        const uploadedFiles = event.detail.files;
        let fileNames = uploadedFiles.map(file => file.name).join(', ');

        // Show success message
        this.dispatchEvent(
            new ShowToastEvent({
                title: 'Files Uploaded',
                message: fileNames + ' uploaded successfully!',
                variant: 'success',
            }),
        );
    }
}
```

fileUploadExample.js-meta.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<LightningComponentBundle xmlns="http://soap.sforce.com/2006/04/metadata">
    <apiVersion>59.0</apiVersion>
    <isExposed>true</isExposed>
    <targets>
        <target>lightning__RecordPage</target>
        <target>lightning__AppPage</target>
    </targets>
</LightningComponentBundle>
```

VS code setup and LWC Code



```
<template>
    <lightning-card title="Upload Medical Report" icon-name="custom:custom63">
        <div class="slds-m-around_medium">
            <lightning-file-upload
                label="Attach Files"
                name="fileUploader"
                accept=".pdf,.png,.jpg,.jpeg"
                record-id={recordId}
                onuploadfinished={handleUploadFinished}
                multiple>
            </lightning-file-upload>
        </div>
    </lightning-card>
</template>
```

```
target dir = e:\Salesforce\MYORG\force-app\main\default\lwc
  create force-app\main\default\lwc\adminDashboard\adminDashboard.js
  create force-app\main\default\lwc\adminDashboard\adminDashboard.html
  create force-app\main\default\lwc\adminDashboard\__tests__\adminDashboard.test.js
  create force-app\main\default\lwc\adminDashboard\adminDashboard.js-meta.xml

16:27:06.31 Finished SFDX: Create Lightning Web Component
```