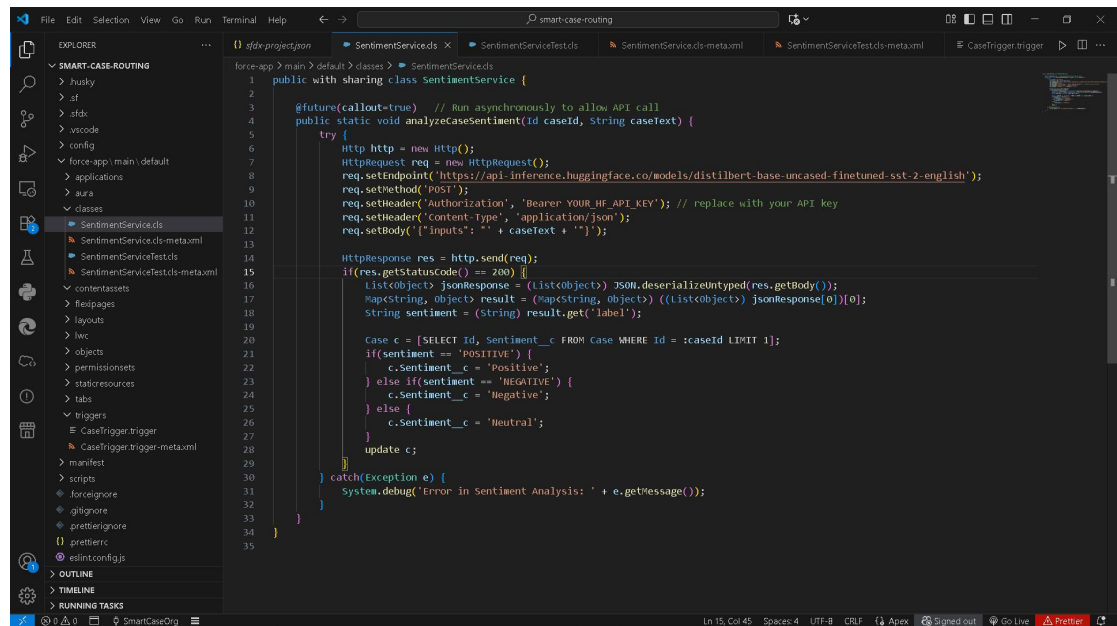


Phase 5: Apex Development

1. Classes & Objects

Apex Class: SentimentService.cls → Handles API callout to Hugging Face.

Objects: Case (standard), optional Sentiment_Log__c for debug storage.



2. Apex Triggers (after insert)

Trigger: CaseTrigger → Runs after Case insert.

Action: Calls SentimentService to analyze Description text.

3. Trigger Design Pattern

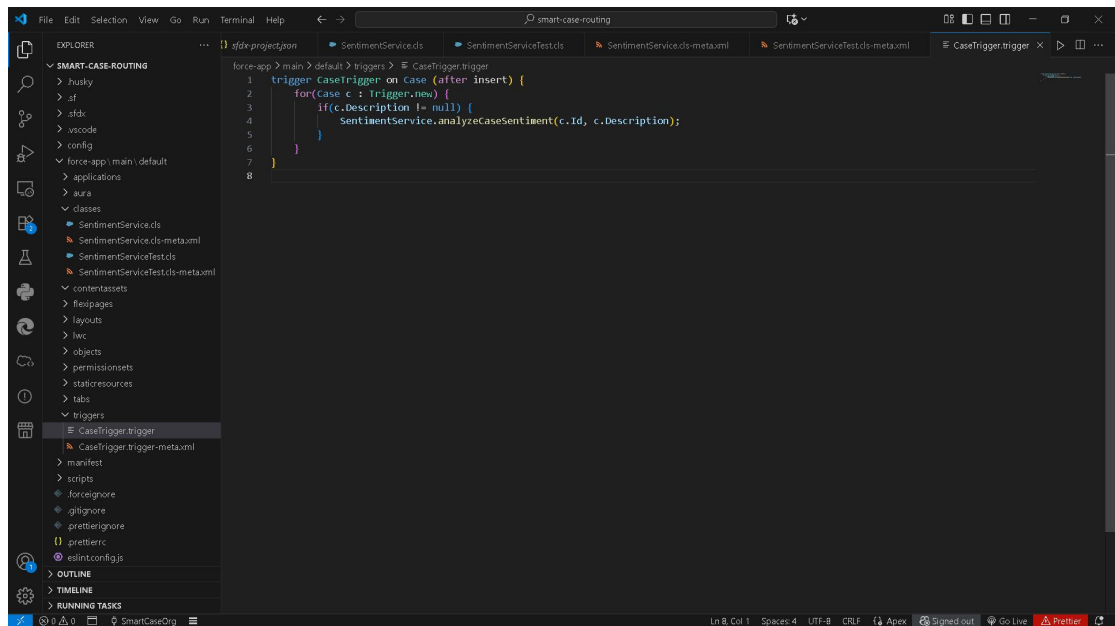
One Trigger per Object.

Logic moved to Handler class for bulk-safe, reusable code.

Example structure:

CaseTrigger.trigger

CaseTriggerHandler.cls



4. SOQL & SOSL

SOQL Example:

```
List<Case> cases = [SELECT Id, Subject, Description, Sentiment__c
                    FROM Case
                    WHERE Priority_Level__c = 'High'];
```

SOSL Example:

```
List<List<SObject>> results = [FIND 'error' IN ALL FIELDS RETURNING
                               Case(Subject, Description, Sentiment__c)];
```

5. Collections: List, Set, Map

List<Case> → Bulk insert/update.

Map<Id, Case> → Efficient lookups.

Set<String> → Track unique keywords in description.

6. Asynchronous Apex

Future Method: Used for external API callouts (Hugging Face).

Queueable Apex: Could be used if chaining or complex callout handling is needed.

Batch Apex: For re-analyzing thousands of old cases in bulk.

Scheduled Apex: SLA monitoring job that runs nightly.

7. Exception Handling

Try-Catch in SentimentService to handle API errors.

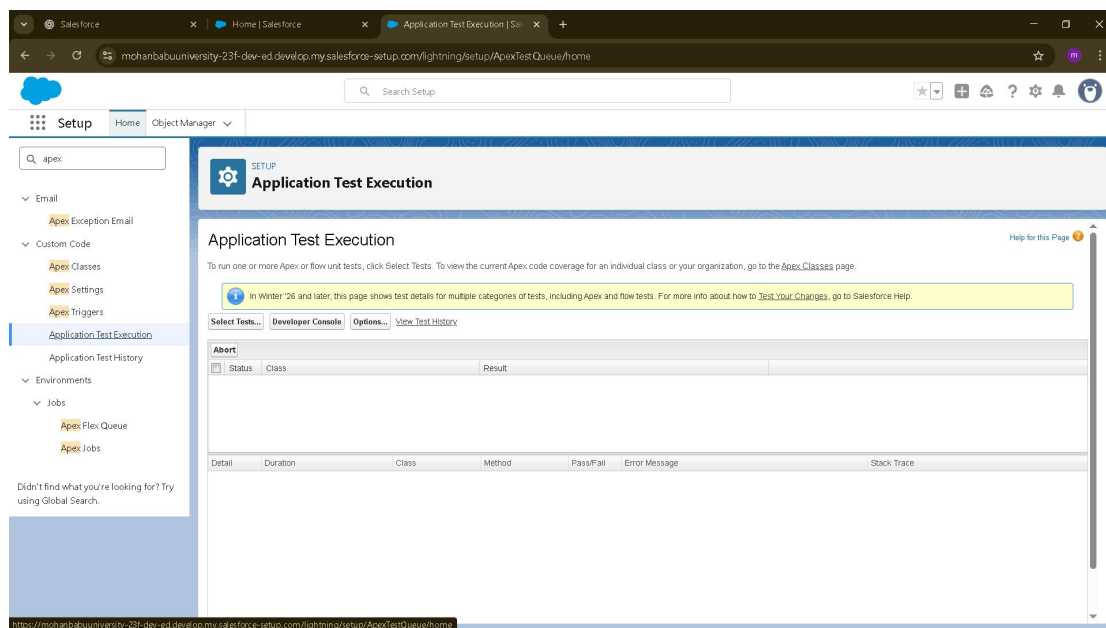
Log errors to Sentiment_Log__c.

8. Test Classes

Requirement: **≥75% coverage** for deployment.

SentimentServiceTest.cls → Mocks API response, asserts sentiment field update.

CaseTriggerTest.cls → Inserts test case, verifies trigger execution.



Screenshot: VS Code with Apex class, trigger, and test files