Django Signals Assessment Documentation

Objective

The assessment aimed to evaluate the default behaviour of Django signals by addressing three main questions:

1. **Are Django signals executed synchronously or asynchronously?**
2. **Do Django signals run in the same thread as the caller?**
3. **Do Django signals run in the same database transaction as the caller?**

Approach

A Django project was created with:

- **Model (MyModel):** A simple model to trigger post_save signals.
- **Signal Handler:** A function that listens for post_save events, testing synchronous behaviour, threading, and transaction rollback.
- **Unit Tests:** Designed to verify the default signal behaviour across the three key aspects.

Key Findings:

1. **Synchronous Execution:**
   The signal handler contained a time.sleep(5) delay, and the test confirmed that signals execute synchronously by measuring the total runtime.
2. **Same Thread:**
   The test compared the signal's thread with the main thread (MainThread), verifying that the signal runs in the same thread as the caller.
3. **Same Transaction:**
   A ValidationError was raised in the signal to test transaction behaviour. The test confirmed that the signal participates in the same transaction, causing a rollback if an error is raised.

Test Results :

```
(base) PS C:\Learning\Projects\Django_Projects\signals_assessment> python manage.py test
Found 3 test(s).
Creating test database for alias 'default'...
System check identified no issues (0 silenced).

[TEST] Verifying if the signal runs in the same database transaction as the caller...
Signal received for FailTransaction in thread: MainThread
[RESULT] Signal caused a rollback, and no object was created, verifying it's part of the same transaction.


.
[TEST] Verifying if the signal runs in the same thread as the caller...
Signal received for TestThread in thread: MainThread
Signal processing completed.
[RESULT] Signal ran in the same thread ('MainThread') as expected.


.
[TEST] Verifying if the signal runs synchronously...
Signal received for TestSync in thread: MainThread
Signal processing completed.
[RESULT] Signal ran synchronously as expected.


.
----------------------------------------------------------------------
Ran 3 tests in 5.012s

OK
Destroying test database for alias 'default'...
```

Conclusion :

- Django signals are executed synchronously by default.
- They run in the same thread as the caller.
- They participate in the same database transaction as the caller, capable of triggering rollbacks.

These behaviours were verified through comprehensive test cases.