**Automated Testing Suite Design – Banking App (Login & Fund Transfer)**

**1. Tools & Frameworks**

* **Testing Framework**: Selenium / Cypress (Web), Appium (Mobile)
* **Assertion Library**: TestNG / JUnit (Java), Mocha / Chai (JavaScript)
* **CI/CD**: Jenkins / GitHub Actions
* **Load Testing**: JMeter / Locust (for concurrency)
* **Mocking Services**: WireMock (for simulating APIs)

**2. Test Suite Structure**

**BankingAppTestSuite**

**├── LoginTests**

**│ ├── test\_validLogin()**

**│ ├── test\_invalidPin()**

**│ ├── test\_multipleConcurrentLogins()**

**├── FundTransferTests**

**│ ├── test\_validTransfer()**

**│ ├── test\_insufficientBalance()**

**│ ├── test\_invalidAccount()**

**│ ├── test\_transferLimitExceeded()**

**│ ├── test\_concurrentTransfers()**

**3. Key Test Cases**

**Login Functionality**

| **Test Case ID** | **Description** |
| --- | --- |
| LT001 | **Valid Login**: User enters correct credentials and lands on dashboard. |
| LT002 | **Invalid PIN**: User enters wrong PIN 3 times → account lock or error message. |
| LT003 | **Concurrent Login**: Simulate 100 users logging in at once → server should handle without crash or major delays. |
| LT004 | **Session Expiry**: After inactivity, session should expire and redirect to login page. |
| LT005 | **Brute Force Prevention**: After repeated failed login attempts, account should be temporarily locked or CAPTCHA shown. |

**Fund Transfer Functionality**

| **Test Case ID** | **Description** |
| --- | --- |
| FT001 | **Valid Transfer**: Transfer from Account A to B with valid details and sufficient balance. |
| FT002 | **Insufficient Balance**: Transfer amount > available balance → show error. |
| FT003 | **Invalid Recipient Account**: Enter non-existent account → return “account not found”. |
| FT004 | **Transfer Limit Exceeded**: Try exceeding daily/monthly limit → reject transaction. |
| FT005 | **Concurrent Transfers**: Simulate 100 fund transfers at the same time → check for race conditions and data consistency. |
| FT006 | **Fund Transfer with Special Characters in Remarks**: Check for input sanitization to prevent injection attacks. |
| FT007 | **Transfer to Self**: Prevent or allow transfer to own account as per business rules. |

**4. Load/Concurrency Testing Setup**

* **Scenario**: 1000 concurrent logins + 500 simultaneous fund transfers
* **Tool**: JMeter or Locust
* **Metrics Captured**:
  + Response Time (avg/max)
  + Error Rate
  + Server CPU/Memory Usage

**5. Integration into CI/CD**

* **Pre-deployment**: Run unit and integration tests on push
* **Post-deployment (Staging)**: Run full suite including concurrency/load
* **Notifications**: Slack / Email alerts for failures

**Benefits**

* Ensures **robust login security**, **transaction integrity**, and **system resilience under load**.
* Reduces **manual regression effort**.
* Early detection of **race conditions**, **validation bugs**, and **performance bottlenecks**.