API Testing Flow in REAL Companies

1. **Requirement**
    1. API Documentation (If they are Good Company)
        1. Google Doc
        2. Google Excel
        3. Confluence
        4. Swagger UI - httpbin.org/
        5. Postman Documentation - apidocs.imgur.com/#c85c9dfc-7487-4de2-9ecd-66f727cf3139
        6. HTML page (URL)
            1. docs.github.com/en/rest/repos?apiVersion=2022-11-28
            2. restful-booker.herokuapp.com/apidoc/index.html#api-Booking-DeleteBooking
            3. apidocs.imgur.com/
2. **No Documentation**
        1. Create your Own Documentation
            1. UI is ready ->
                1. docs.google.com/document/d/1OVxw2b9XHutE_trecdlP1lvWrn5u_yw3/edit#heading=h.gjdgxs
                2.
            2. No UI - **Curl** Way (Mech) - Create your Own Req. -
                1. Ask the **Developer for the Curl Request**
                2. via AI Generation - chatgpt.com/share/67c29482-9290-8009-a8f5-e551ebc2874d
                3. you.com/search?q=Generate+a+API+Documentation+in+this+format+for+this+request+%0A%0AExample+%3A++%22Objective+-+This+API+is...&cid=c1_a9f29fa2-3c63-4878-9826-340d6e1c72c4&tbm=youchat

3. **Test Plan (Get it reviewed by the QA Lead/Product Manager)**
    1. Google Doc
    2. JIRA - Zephyr
4. **Test Cases**
    1. docs.google.com/spreadsheets/d/1EH1UJ9Qezgx_aZ0xim3KcVJUCEeR7A-7/edit?usp=sharing&ouid=104755920778477387077&rtpof=true&sd=true
    2. LIVE Zephyr - > prnt.sc/1LO-8X5Bq0EV
5. **Test Environment - Setup**
    1. Max - **DevOps (Person) -> SRE, Deployment Application)**
    2. Dev
    3. QA (Browser Stack, Tekion)
6. **Test Execution**
    1. Manual
        1. **Excel Sheet** -> Execute Your Testcase where? API (No UI) - Tools
            1. Postman - Big Daddy
            2. Swagger, SOAP UI, Advanced Rest Client, Soapy, Katalon Studio, JMeter...
    2. Automation -> (Test Case Stable)
        1. Postman - Automation -> We don't use it. (Maintenance is Very High, Code Reusability Issues, Duplicate Issues).
        2. **Rest Assured** - Lib Java API Automation

What to Test in an API?

- Verify the Parse of JSON or XML response (i.e. Body and Response)
- HTTP Status Code
- Verify the Headers response with Negative or Invalid headers
- Max/ Min Key values
- Missing Keys
- Response Time
- Cookies
- JSON Schema - Parameters
- Authorization
- CRUD Operations
- Empty Updates or POST call handling
- Chaining request verification
- Verify how the APIs' error codes are handled
- Duplicate entries, Error message
- Testing Tech - Boundary Values

Extra:

- Test nulls and empty strings; these have a habit of causing issues.
- Test with Unicode characters, ensure they are correctly saved to the database.
- Test mandatory and optional parameters, in particular, test with just mandatory values.
- Ensure correct status codes are returned; you will find false positives, e.g. a 500 being returned as a 200.
- Functionality Bugs: The test looks for missing functionality bugs.
- Reliability Bugs: API testing helps identify bugs in integration across different systems.

- Performance Bugs: The tests help determine how much traffic the system can handle before being overloaded, and how to expand infrastructure to meet rising demand at the core, but also are very effective at pinpointing weak points in the API.
- Security Bugs.

Common Bugs in API?

- Missing Keys
- Duplicate Keys
- Incorrect HTTP Code
- HTTP Methods not Handled Properly
- Performance Issues
- Security Issues
- Validation Errors
- API Monitoring Issues