JMeter Performance Test Plan Documentation: Demoblaze

Test Plan Name: Demoblaze Performance Tests

Description: This JMeter Test Plan is designed to measure the performance of key user interactions on the Demoblaze e-commerce website, focusing on browsing the "Phones" category and interacting with product listings.

Test Cases Included:

1. User successfully navigates to the "Phones" category Selection
2. Product Listing Interactions (View Product Details)
3. User Verify that clicking on "Next" and "Previous" Button will redirect them to the next and previous page

I. Phones Category Load Test (Thread Group)

- **Purpose:** Simulates multiple concurrent users loading the homepage and then navigating to the "Phones" category page.
- **Configuration:**
  - **Name:** Phones Category Load Test
  - **Number of Threads (users):** 20 (This can be adjusted based on the desired load level)
  - **Ramp-up Period (in seconds):** 10 (JMeter will start 20 users over a period of 10 seconds)
  - **Loop Count:** 5 (Each simulated user will execute the scenario 5 times)
- **Test Steps:**
  1. **Load Homepage (HTTP Request Sampler):**
     - **Name:** Load Homepage
     - **Protocol:** https
     - **Server Name or IP:** www.demoblaze.com
     - **Method:** GET
     - **Path:** /index.html
  2. **Homepage Think Time (Constant Timer):**
     - **Name:** Homepage Think Time
     - **Thread Delay (milliseconds):** 2000 (Simulates a 2-second pause a user might take before navigating)
  3. **Load Phones Category (HTTP Request Sampler):**
     - **Name:** Load Phones Category
     - **Protocol:** https
     - **Server Name or IP:** www.demoblaze.com
     - **Method:** GET
     - **Path:** /bycat?cat=phone

- **Listeners:**
  - **Summary Report:** Provides a table summarizing the test results, including the number of samples, average response time, error rate, and throughput.
  - **Aggregate Report:** Similar to the Summary Report but provides more detailed statistics like median, 90% line, 95% line, min, and max response times.
  - **Graph Results:** Displays response times over time in a graphical format.
- **Key Performance Indicators (KPIs) to Monitor:**
  - Average response time for loading the homepage.
  - Average response time for loading the "Phones" category page.
  - Throughput (requests per second) for both requests.
  - Error rate for both requests (should ideally be 0%).

II. View Product Details Test (Thread Group)

- **Purpose:** Simulates users loading the "Phones" category page and then viewing details of individual products.
- **Configuration:**
  - **Name:** View Product Details Test
  - **Number of Threads (users):** 15
  - **Ramp-up Period (in seconds):** 5
  - **Loop Count:** 3
- **Test Steps:**
  1. **Load Phones Category (HTTP Request Sampler):**
     - **Name:** Load Phones Category
     - **Protocol:** https
     - **Server Name or IP:** www.demoblaze.com
     - **Method:** GET
     - **Path:** /bycat?cat=phone
  2. **Extract Product IDs (Regular Expression Extractor):**
     - **Name:** Extract Product IDs
     - **Apply to:** Main sample only
     - **Response Field to Check:** Body
     - **Regular Expression:** `href="/prod.html?idp_=(.*?)"` (This regex extracts the product ID from the product links on the "Phones" category page. **Note:** You might need to adjust this based on the actual HTML structure.)
     - **Template:** `$1$` (Uses the first captured group)
     - **Match No. (0 for Random):** 0 (Selects a random product ID from the matches)

        - **Default Value:** 1 (If no product ID is found, it defaults to '1')
        - **Variable Name to store the result:** `productID`
    3. **Category Page Think Time (Constant Timer):**
        - **Name:** Category Page Think Time
        - **Thread Delay (milliseconds):** 1500 (Simulates a 1.5-second pause before viewing a product)
    4. **View Product Details (HTTP Request Sampler):**
        - **Name:** View Product Details
        - **Protocol:** https
        - **Server Name or IP:** www.demoblaze.com
        - **Method:** GET
        - **Path:** `/prod.html?idp_=${productID}` (Uses the extracted `productID` variable)
- **Listeners:**
    - **Summary Report**
    - **Aggregate Report**
    - **Graph Results**
- **KPIs to Monitor:**
    - Average response time for loading the "Phones" category page.
    - Average response time for viewing product details.
    - Throughput for both requests.
    - Error rate for both requests.

III. Product Pagination Test (Thread Group)

- **Purpose:** Measures the performance of navigating between pages within the "Phones" category using the "Next" and "Previous" buttons.
- **Configuration:**
    - **Name:** Product Pagination Test
    - **Number of Threads (users):** 10
    - **Ramp-up Period (in seconds):** 5
    - **Loop Count:** 2
- **Test Steps:**
    1. **Load Phones Category (HTTP Request Sampler):**
        - **Name:** Load Phones Category
        - **Protocol:** https
        - **Server Name or IP:** www.demoblaze.com
        - **Method:** GET
        - **Path:** /bycat?cat=phone
    2. **Extract Next Button URL (Regular Expression Extractor):**
        - **Name:** Extract Next Button URL
        - **Apply to:** Main sample only

- **Response Field to Check:** Body
- **Regular Expression: You need to inspect the HTML source of the "Phones" category page to find the pattern for the "Next" button's `href` attribute.** It might look something like `<a id="next".*?href="(.*?)"`. Adjust based on the actual HTML.
- **Template:** `$1$`
- **Match No.:** 1 (Assuming the first match is the relevant "Next" button)
- **Default Value:** `/bycat?cat=phone` (If the "Next" button is not found)
- **Variable Name to store the result:** `nextButtonURL`

3. **Category Page Think Time (Constant Timer):**
   - **Name:** Category Page Think Time
   - **Thread Delay (milliseconds):** 1500
4. **Click Next (HTTP Request Sampler):**
   - **Name:** Click Next
   - **Protocol:** https
   - **Server Name or IP:** www.demoblaze.com
   - **Method:** GET
   - **Path:** `${nextButtonURL}` (Uses the extracted URL)
5. **Extract Previous Button URL (Regular Expression Extractor):**
   - **Name:** Extract Previous Button URL
   - **Apply to:** Main sample only
   - **Response Field to Check:** Body
   - **Regular Expression: Inspect the HTML source of the *next* page to find the pattern for the "Previous" button's `href` attribute.** It might look something like `<a id="prev".*?href="(.*?)"`. Adjust based on the actual HTML.
   - **Template:** `$1$`
   - **Match No.:** 1
   - **Default Value:** `/bycat?cat=phone`
   - **Variable Name to store the result:** `previousButtonURL`
6. **Next Page Think Time (Constant Timer):**
   - **Name:** Next Page Think Time
   - **Thread Delay (milliseconds):** 1500
7. **Click Previous (HTTP Request Sampler):**
   - **Name:** Click Previous
   - **Protocol:** https
   - **Server Name or IP:** www.demoblaze.com

        - **Method:** GET
        - **Path:** `${previousButtonURL}`
- **Listeners:**
    - **Summary Report**
    - **Aggregate Report**
    - **Graph Results**
- **KPIs to Monitor:**
    - Response times for loading the "Phones" category page.
    - Response time for clicking the "Next" button.
    - Response time for clicking the "Previous" button.
    - Throughput for all requests.
    - Error rate for all requests.

How to Implement in JMeter:

1. Open JMeter.
2. Create a new Test Plan.
3. For each Test Case, create a separate Thread Group and configure its properties.
4. Within each Thread Group, add the HTTP Request Samplers, Timers, and Regular Expression Extractors as described above.
5. Add the specified Listeners at the Test Plan level (or under each Thread Group for more granular results).

7.Run the Test Plan and analyze the results in the Listeners.