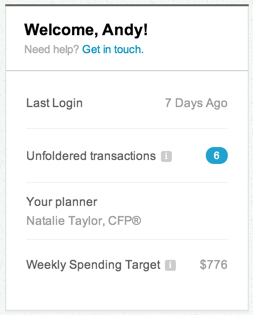
**Scenario**: You are given the task to test out a new area of the website that gives the user a snapshot of important information once they login. You are given both an image of the design and acceptance criteria / requirements. Please review them below:



**Acceptance Criteria / Requirements:**

- Users should see a welcome module in which they are presented with the following information:

* A welcome message which includes their first name
* Information about the last time they logged in
* Number of bank/credit card transactions they have which need to be manually categorized by the user (categories are normally assigned automatically via an algorithm. Example category might be rent, gas, restaurants, etc. In the case where it can’t automatically be categorized, it will be unfoldered).
  + Clicking on this link takes you to a transactions screen
* The name of your planner
* How much money you are allowed to spend every week

**Task 1**: Write a list of questions or uncertainties you may have that you would ask a designer or stakeholder concerning the image or acceptance criteria/requirements.

* Expected response time SLA of login transaction.
* Get in touch Volume and response time.
* Expected User volume.
* Expected response time SLA of Transactions screen.
* Acceptable Sub transaction response times:

1. unfolded transactions

2. Your planner

3. Weekly Spending target.

4. Mobile? / Desktop?

* Target transaction volume per hour (hits/sec, throughput).
* Percentage of transaction failures accepted for each test.
* Application architecture artifacts.
* Test environment stats vs production.
* Entry criteria:

Weather the application is functionally stable or not

**Task 2** : Write a list of data or users in a certain state that you may need to properly test this.

* Is the test data reusable or created?
* No.of Test user IDs
* Any user lock outs?
* User count split Mobile/Desktop?
* Average no.of unfolded transactions
* Transaction (Business critical) volume for each userID.
* Are there any test data dependencies for the clicks to go to transaction screen?
* Allowed money limit?

**Task 3** : Write a list of high-level test scenarios that you can think of based only on the design and acceptance criteria you are given. After you list them out, number them in terms of priority (1 - must test, 2 - should test, 3 - test only if there is time).

Example:

**3** - Verify that users who have not logged in for more than a year display the correct

**1** - Verify that clicking on the transactions link takes you to the transactions section

* Priority1: a) Verify that all the users login successfully with respect to data.

b) Verify that clicking on the transactions link takes you to the transactions section.

c) Execute an endurance test to verify the application has no memory leaks or thread related bottlenecks.

d) Verify load balancing algorithm (failover test).

* Priority2: Verify weather application is capable of handling unexpected load (Stress test, Spike test).
* Priority3: a) Identify users with highest number of unfolded transactions and execute with the volumes.

b) Verify that users who have not logged in for more than a year display the correct.

c) Webservice testing if required.

d) Test Get in touch if you have time.

**Task 4:** Imagine that you find a defect while testing the above. Write a sample defect report below:

**Defect:**

While running an average load test expected SLA is met, but when running an endurance (long running) I have identified transaction response time discrepancies. Transaction response times were elevated for login transactions after two hours in to test execution.

**Root cause (Possibly Memory leak):**

Identified consistent increase in heap utilization (Old gen), also saw tooth pattern was not seen during the test.

**Work around/ recommendation:**

1.Revisit on JVM/ Heap configuration.

2.GC algorithm.

3. Identify the method where the objects are kept reference without use.

**Expected outcome / Post defect fix:**

Successful test run without any memory leaks.

**Task 5:** What scenarios would you consider valid for automation? Why?

1. Login using credentials
2. Add text assertions, value assertions and color assertions to validate the texts, numbers and colors in the page after the user login. Below is the screenshot with the text assertion possibilities.
3. Automating these scenarios will save time and application functionality can we validated without any functional defects.

