



**Software Engineering**  
**IT314: Final Submission**  
**Group 22**  
**Non Functional Testing**

**Note:** We are going to test our system (for load balance testing) on JMeter which is best for non-functional testing.

### 1. Load Balance Testing

- Testing Login Module (POST request to PATH '/profile')

**Scenario:** We are going to log in with the same user from multiple hosts (500) and will be measuring load balance and throughput of the system.

The screenshot shows the JMeter interface with the following details:

- Project Structure:** The left sidebar shows a tree structure for "Placement Cell Testing" with a "Student" node expanded, containing "HTTP Request", "View Results Tree", "Summary Report", "HTTP Header Manager", and "HTTP Cookie Manager".
- Thread Group Configuration:**
  - Name: Student
  - Comments: [empty]
  - Action to be taken after a Sampler error: Continue (radio button selected)
- Thread Properties:**
  - Number of Threads (users): 500
  - Ramp-up period (seconds): 1
  - Loop Count: Infinite (checkbox checked)
  - Same user on each iteration (checkbox checked)
  - Delay Thread creation until needed (checkbox unchecked)
  - Specify Thread lifetime (checkbox unchecked)
  - Duration (seconds): [empty]
  - Startup delay (seconds): [empty]
- Log Output:** At the bottom, a terminal-like window displays JMeter log entries. The last few entries are:
 

```

991 2023-04-26 19:42:58,689 INFO o.a.j.t.JMeterThread: Thread is done; Student 1-499
992 2023-04-26 19:42:58,689 INFO o.a.j.t.JMeterThread: Thread finished; Student 1-499
993 2023-04-26 19:42:58,696 INFO o.a.j.t.JMeterThread: Thread is done; Student 1-498
994 2023-04-26 19:42:58,696 INFO o.a.j.t.JMeterThread: Thread finished; Student 1-498
995 2023-04-26 19:42:58,868 INFO o.a.j.t.JMeterThread: Thread is done; Student 1-435
996 2023-04-26 19:42:58,868 INFO o.a.j.t.JMeterThread: Thread finished; Student 1-435
997 2023-04-26 19:42:58,882 INFO o.a.j.t.JMeterThread: Thread is done; Student 1-376
998 2023-04-26 19:42:58,882 INFO o.a.j.t.JMeterThread: Thread finished; Student 1-376
999 2023-04-26 19:42:58,885 INFO o.a.j.e.StandardJMeterEngine: Notifying test listeners of end of test
1000 2023-04-26 19:42:58,887 INFO o.a.j.g.u.JMeterMenuBar: setRunning(false, *local*)
      
```

Below screenshot displays the POST request to our deployed website in JMeter screen.

The screenshot shows the JMeter interface with the following details:

- Project Structure:** Placement Cell Testing > Student > HTTP Request.
- HTTP Request Configuration:**
  - Name: HTTP Request
  - Protocol [http]: https
  - Server Name or IP: placement-cell-28yr.onrender.com
  - Port Number: (empty)
  - Method: POST
  - Path: /profile
  - Follow Redirects: checked
  - Use KeepAlive: checked
  - Content encoding: (empty)
  - Parameters (Table):
 

Name:	Value	URL Encode?	Content-Type	Include Equals?
email	202001226@daiict.ac.in	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
password	Jatin@123	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
role	Student	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
- Log View:** Shows a list of log entries from the test execution.

We can see in the below summary report that throughput of login page while we have 500 concurrent user is 3.0 sec.

The screenshot shows the JMeter interface with the following details:

- Project Structure:** Placement Cell Testing > Student > Summary Report.
- Summary Report Configuration:**
  - Name: Summary Report
  - Comments: (empty)
  - Filename: (empty)
  - Buttons: Browse..., Log/Display Only: Errors, Successes, Configure.
- Summary Report Table:**

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/s...	Sent KB/sec	Avg. Bytes
HTTP Request	500	111627	10956	165620	42306.26	20.60%	3.0/sec	14.32	2.10	4879.7
TOTAL	500	111627	10956	165620	42306.26	20.60%	3.0/sec	14.32	2.10	4879.7
- Log View:** Shows a list of log entries from the test execution.

- Testing view company module which access our database to load all list of jobs

The screenshot shows the JMeter interface with a single thread group named "View Company". The "Basic" tab is selected under "Web Server". The "HTTP Request" section shows a GET request to "/viewCompany". The "Parameters" table contains two entries: "id" with value "64495f7bae0bd642d9955fd2" and "role" with value "Student". Below the table are buttons for "Detail", "Add", "Add from Clipboard", "Delete", "Up", and "Down". At the bottom of the screen, a terminal window displays the execution log for 500 requests, starting with:

```

991 2023-04-27 01:04:25,968 INFO o.a.j.t.JMeterThread: Thread is done: Student 1-495
992 2023-04-27 01:04:25,968 INFO o.a.j.t.JMeterThread: Thread finished: Student 1-495
993 2023-04-27 01:04:25,969 INFO o.a.j.t.JMeterThread: Thread is done: Student 1-489
994 2023-04-27 01:04:25,969 INFO o.a.j.t.JMeterThread: Thread finished: Student 1-489
995 2023-04-27 01:04:26,435 INFO o.a.j.t.JMeterThread: Thread is done: Student 1-409
996 2023-04-27 01:04:26,435 INFO o.a.j.t.JMeterThread: Thread finished: Student 1-409
997 2023-04-27 01:04:26,561 INFO o.a.j.t.JMeterThread: Thread is done: Student 1-444
998 2023-04-27 01:04:26,561 INFO o.a.j.t.JMeterThread: Thread finished: Student 1-444
999 2023-04-27 01:04:26,561 INFO o.a.j.e.StandardJMeterEngine: Notifying test listeners of end of test
1000 2023-04-27 01:04:26,562 INFO o.a.j.g.u.JMeterMenuBar: setRunning(false, *local*)
1001

```

## Result of the test (500 requests)

The screenshot shows the JMeter interface with a single thread group named "Summary Report". The "Basic" tab is selected under "Web Server". The "Summary Report" table shows results for "View Company" and "TOTAL" samples. The "Label" column lists "View Company" and "TOTAL". The "# Samples" column shows "500" for both. The "Average" column shows "30311" for "View Company" and "30311" for "TOTAL". The "Min" column shows "25664" for both. The "Max" column shows "32468" for "View Company" and "32468" for "TOTAL". The "Std. Dev." column shows "1066.24" for both. The "Error %" column shows "0.00%" for both. The "Throughput" column shows "15.0/sec" for both. The "Received KB/s..." column shows "217.71" for both. The "Sent KB/sec" column shows "2.40" for both. The "Avg. Bytes" column shows "14873.0" for both. Below the table are checkboxes for "Include group name in label?", "Save Table Data", and "Save Table Header". At the bottom of the screen, a terminal window displays the execution log for 500 requests, identical to the one in the previous screenshot.

## 2. Security Testing

Authentication and Authorization: We have also used JWT authentication for authorizing the users while accessing all paths. It means no one can see the inside details without having login into the system.

Data Encryption: We have used bcrypt for encryption of the password. We have added 10 salt rounds to the encryption of the password. As we have used encryption on the password, we as database admin cannot see the password of any users.

Session Management: We have used cookies for session management, using this we have assured that user can re-login within the 24 hour after logging in. This condition applies even if we close the browser.

Role-Based Access Controls: We have also ensured that company's hr cannot view the details of the other companies that are coming for placement. Also companies cannot see the admin statistics and cannot perform the things that admin can do.

## 3. Compatibility Testing

We have ensured that the portal runs on all latest web browsers including safari, chrome, brave.

Safari:

The screenshot shows a web application interface in a Safari browser window. On the left, there is a dark sidebar with a user profile picture of Jatin Ranpariya, a 'Profile' link, an 'Update Password' link, an 'Update Resume' link, a 'Companies' link, and a 'Logout' link. At the bottom of the sidebar is a 'Display a menu' button. The main content area has a teal header bar. Below the header is a 'Filter' section with dropdown menus for 'CPI' (Any), 'Location' (Any), 'Branch' (Any), 'CTC' (Any), and 'Job Title' (Any), followed by a blue 'Apply' button. Underneath the filter is a table titled 'Company' with the following data:

Name	Criteria	CTC	Open For	Job Title	Details	(De)Register
Tekion	6	15 LPA	B.Tech	Associate Software Engineer	<button>Details</button>	<button>Not Started</button>
Google	7	20 LPA	B.Tech , M.Tech	Software Developer	<button>Details</button>	<button>Register</button>
Fintech Global	7	26 LPA		Developer	<button>Details</button>	<button>Branch</button>
Fintech Global	1	40 LPA	B.Tech , M.Tech	System Designer	<button>Details</button>	<button>Register</button>

Chrome:

The screenshot shows a web browser window with the title "placement-cell-28yr.onrender.com/viewCompany". On the left, there is a sidebar with a user profile picture of Jatin Ranpariya and links for Profile, Update Password, Update Resume, Companies, and Logout. The main content area has a "Filter" section at the top with dropdowns for CPI, Location, Branch, CTC, and Job Title, followed by an "Apply" button. Below the filter is a table titled "Company" with columns: Name, Criteria, CTC, Open For, Job Title, Details, and (De)Register. The table lists four companies: Tekion, Google, Fintech Global, and Fintech Global again, each with different details and registration status.

Name	Criteria	CTC	Open For	Job Title	Details	(De)Register
Tekion	6	15 LPA	B.Tech	Associate Software Engineer	<a href="#">Details</a>	<a href="#">Not Started</a>
Google	7	20 LPA	B.Tech , M.Tech	Software Developer	<a href="#">Details</a>	<a href="#">Register</a>
Fintech Global	7	26 LPA		Deloper	<a href="#">Details</a>	<a href="#">Branch</a>
Fintech Global	1	40 LPA	B.Tech , M.Tech	System Designer	<a href="#">Details</a>	<a href="#">Register</a>

Brave:

The screenshot shows a web browser window with the title "localhost:3000/viewCompany". The layout is identical to the Chrome version, featuring a sidebar with the user profile of Jatin Ranpariya and a main content area with a "Filter" section and a company list table. The companies listed are the same as in the Chrome screenshot, with slight variations in the "Branch" and "Job Title" columns for the second and third entries.

Name	Criteria	CTC	Open For	Job Title	Details	(De)Register
Tekion	6	15 LPA	B.Tech	Associate Software Engineer	<a href="#">Details</a>	<a href="#">Not Started</a>
Google	7	20 LPA	B.Tech , M.Tech	Software Developer	<a href="#">Details</a>	<a href="#">Register</a>
Fintech Global	7	26 LPA		Deloper	<a href="#">Details</a>	<a href="#">Branch</a>
Fintech Global	1	40 LPA	B.Tech , M.Tech	System Designer	<a href="#">Details</a>	<a href="#">Not Started</a>

## 4. Reliability testing

This type of testing evaluates the software system's ability to function correctly and reliably over a specific period of time, including testing for failure recovery, fault tolerance, and stability.

The screenshot shows the MongoDB Atlas Cluster Placement Cell Overview page. The cluster is named 'ClusterPlacementCell'. It is a 'Shared Tier Cluster' with three shards across three regions: 'ac-ytqwc4u-shard-00-00.72blkco.mongodb.net:27017', 'ac-ytqwc4u-shard-00-01.72blkco.mongodb.net:27017', and 'ac-ytqwc4u-shard-00-02.72blkco.mongodb.net:27017'. The cluster is in version 6.0.5, located in AWS N. Virginia (us-east-1) with M0 Sandbox (General) tier. The interface includes tabs for Overview, Real Time, Metrics, Collections, Search, Profiler, Performance Advisor, Online Archive, and Cmd Line Tools. There are also sections for NODES, REPLICA SET, and various monitoring and configuration options like Connect, Configuration, and Upgrade.

As we can see from the above screen shot, our portal database is deployed on 3 separate regions. So it gives high fault tolerance. If one of the region fails, then we have other 2 regions which is up and running. So we can have our updated data available for anytime.

## 5. Usability testing

This type of testing evaluates how easy it is to use the software system, including its interface and design. It includes testing for user experience, accessibility, and internationalization.

In our system, Student has very obvious reason why (s)he cannot register for the jobs in which they have no matching criteria. Suppose if a student is of the branch B.Tech and job is opened for only M.Tech students, then register button will be disabled and button will have its name as 'branch' in the orange color. (Reference screenshot of 3. Compatibility testing)

According to the Fitts law, distance between current pointer position and target position should be as minimum as possible. We have used the side bars for this because it decreases the overall distance between users working area and side bar.