

# Нагрузочное тестирование

Практический курс по нагрузочному тестированию

Длительность курса: 88 академических часов

## 1 Введение в нагрузочное тестирование (НТ)

- НТ: цели, события, показатели, основные этапы**

объяснить, какие цели ставятся перед нагрузочным тестированием и какие события инициируют НТ;  
объяснить, что такое показатели производительности системы и как их считать;  
планировать нагрузочное тестирование и знать его основные этапы.
- Профиль нагрузочного тестирования**

проанализировать статистику использования системы и прогнозировать нагрузку;  
составить профиль нагрузочного тестирования.  
изучить способы составления профиля нагрузки и применения его к сценариям.

Домашние задания

  - Профиль нагрузочного тестирования  
Цель: 1) Анализ статистики и составление профиля нагрузочного тестирования.  
Требуется составить профиль НТ. Материалы - файл со статистикой (статистика\_дз.xlsx) и файл автоанализа (Профиль\_НТ\_1.5\_пустой.xlsx) прилагаются к материалам урока.  
2) Расчёт параметров сценария нагрузочного тестирования  
Требуется рассчитать параметры (VU и Pacing) сценария НТ.  
Параметры:  
\* Список и интенсивности операций берутся из результатов задания 1

\* Предполагается, что все операции выполняются не более 5 секунд  
Параметры сценария НТ:  
\* Разгон до 200%, после чего 5 ступенек по 20% каждая (220%, 240%, 260%, 280%, 300%).

Профиль НТ:

- Вставить статистику в файл анализа
- Отобрать дни, подходящие для составления профиля НТ
- Отобрать часы, для которых будет составляться профиль НТ
- Скорректировать список операций, попадающих в профиля
- Оценить итоговый профиль

Сценарий НТ:

Необходимо рассчитать параметры сценария НТ:

- \* Количество VU
- \* Pacing

Результаты предоставляются в XLSX-файле

---

### 3 **Методика и отчёт по НТ**

описать основные разделы методики нагрузочного тестирования;  
определить типы отчётов и что можно в них указывать.

1 **Скрипты и сценарии НТ - 1: MF Performance center (PC) и Vugen часть 1/3** разработать скрипт web/http (web сайт).

---

2 **Скрипты и сценарии НТ - 1: MF Performance center (PC) и Vugen часть 2/3** создать сценарий нагрузки; планировать и запускать тесты.

---

3 **Скрипты и сценарии НТ - 1: MF Performance center (PC) и Vugen часть 3/3** настроить LR Analysis; добавить, объединить и настроить отображение графиков; произвести расчёт производительности; провести анализ ошибок.

### Домашние задания

1 Разработка скриптов тестирования и проведение нагрузочного теста web-сайта

Цель: В этом ДЗ вы разработаете 3 скрипта нагрузочного тестирования, составите сценарий нагрузки, проведёте тестовое испытание, соберёте результаты тестов и подготовите краткий отчёт по ним. Выполнив это задание вы научитесь писать скрипты нагрузочного тестирования в LoadRunner VuGen, создавать сценарий тестирования в MF Performance Center и запускать нагрузочное испытание, анализировать результаты теста и составлять отчёт о результатах испытаний.

Пункты со \* не обязательны для получения зачёта по ДЗ.

1. Создайте новый web/http скрипт в LoadRunner VuGen.
2. Запишите скрипт из двух действий Login и Logout на сайте MobileShop (разверните на выделенных виртуальных серверах).
3. Выполните корреляцию токена (необходимо для успешного прохождения скрипта).
4. Выполните параметризацию логина и пароля.
5. Расставьте транзакции для замера времени и успешности прохождения действий: открытие сайта, вход пользователя, выход пользователя + общая транзакция для всех действий.
6. Сохраните полученный результат в свой проект в ALM по адресу <http://194.87.239.51:8080/qcbin/> с вашим логином паролем и названием проекта.
7. Создайте новый проект web/http в LoadRunner VuGen.
8. Запишите скрипт из двух действий (регистрация нового пользователя и выход пользователя) на сайте MobileShop (разверните на выделенных виртуальных серверах).
9. Оцените и выполните необходимые действия по параметризации, корреляции и расстановки транзакций в данном скрипте.

10. Сохраните полученный результат в свой проект в ALM по адресу <http://194.87.239.51:8080/qcbin/> с вашим логином паролем и названием проекта.

\* Создать новый web/http скрипт в LoadRunner VuGen.

\* Записать скрипт из трёх действий: вход пользователя, заказ телефона, выход пользователя (кнопка выхода на сайте отсутствует, нужно добавить свой web\_custom\_request в скрипт, который будет выполнять запрос на выход пользователя). На сайте MobileShop (разверните на выделенных виртуальных серверах).

\* Оцените и выполните необходимые действия по параметризации, корреляции и расстановки транзакций в данном скрипте.

\* Сохраните полученный результат в свой проект в ALM по адресу <http://194.87.239.51:8080/qcbin/> с вашим логином паролем и названием проекта.

11. Откройте MF Performance Center по адресу <http://45.143.93.37/Loadtest/psx/login> и авторизуйтесь под своей учётной записью в своём проекте.

12. Откройте в главном меню Test Runs создайте папку с именем Scenario и добавьте в неё новый сценарий типа Real-World By Test (дайте имя вашему тесту, TestSet можно присвоить позднее или сразу создать новый со свободным именем).

13. Добавьте в сценарий скрипты которые вы создали в шагах выше.

\* Добавьте в сценарий скрипт для оформления заказа на сайте (из задания со звёздочкой).

14. Создайте сценарий нагрузки для поиска максимальной производительности с шагом нагрузки 50% от профиля:

LoginLogout -- 18000 оп./час

RegistrationLogout -- 3600 оп./час

LoginBuyLogout -- 1200 оп./час

Продолжительность теста не более 4х часов, время стабилизации 20-30 минут.

Среднее время выполнения транзакции (для расчёта пэйсинга) вы можете посмотреть в output-логе VUGen'a запустив ваш скрипт на выполнение в 10 итераций и посмотрев время выполнения последних 2-3х транзакций.

\* Установите на выделенный сервер связку Telegraf + InfluxDB + Grafana.

\* Настройте в Grafana Dashboard для отображения метрик утилизации системных ресурсов (CPU used/idle, memory free, disk r/w byte, network send/recieved byte/sec).

15. Проведите тестовое испытание.

16. После завершения теста, перейдите в MF Performance Center. Главное меню Test Runs. Выберите ваш сценарий и нажмите Runs. Выберите в списке ваш тест и нажмите View Results. Выберите вкладку Results, выберите из списка Raw Results и нажмите Download.

17. Распакуйте полученный файл (во избежании ошибок распаковки рекомендуется использовать стандартное приложение Windows).

18. Запустите .lra файл и дождитесь пока Load Runner Analysis выполнить подготовку и анализ результатов.

19. Добавьте графики Avg. Transaction Response Time (с наложенным графиком Running Vusers), Transaction per seconds (с наложенным графиком Running Vusers).

20. Оцените уровень найденной максимальной производительности и уровень пиковой производительности (если она была достигнута).

21. Подготовьте краткий отчёт по шаблону из урока. При

подготовке отчёта используйте глобальные фильтры для оценки максимальной производительности.  
22. Сохраните рабочую сессию в Load Runner Analysis, сожмите в zip архив вместе с файлом отчёта, и отправьте ссылку на GoogleDrive в чат с преподавателем.

---

4 **Скрипты и сценарии НТ - 2: Jmeter часть 1/3**

---

5 **Скрипты и сценарии НТ - 2: Jmeter часть 2/3**

Домашние задания

1 Разработка скрипта и проведение теста в Jmeter

Цель: Домашнее задание делится на 2 части:

- Разработка скриптов
  - Проведение теста и оценка результата.
- ДЗ построено так, что бы вы могли выполнять их независимо.

Разработка скриптов:

- разработать скрипт нагрузочного тестирования в Jmeter
- Проведение теста и оценка результатов:
- провести мини-тест
  - оценить результаты

Домашнее задание считается успешно выполненным в случае, если:

- Разработка скрипта: после выполнения скрипта вы видите в веб-интерфейсе сайта результаты работы скрипта.
- Проведение теста и оценка результата: Вы получили графики, отображающие интенсивности и времена отклика. Интенсивность совпадает с целевой.

```
=====  
=====  
=====Разработка  
скрипта=====
```

```
=====
```

Для выполнения скрипта необходимо зарегистрировать персонального пользователя на сайте  
<http://www.load-test.ru:1080/webtours>  
(альтернативный адрес)  
<http://79.164.106.94:1080/webtours>

Необходимо разработать скрипт, выполняющий следующие действия:

- Открытие страницы <http://www.load-test.ru:1080/webtours>
- Вход в систему под вашим пользователем.
- Переход на страницу "Flights", выбор случайного города отправления и прибытия.
- В половине случаев необходимо купить билет "туда-обратно"

- Выбор случайного рейса по данному направлению из 4х предложенных
- Покупка билета
- Переход на корневую страницу
- Элементы Jmeter, необходимые для выполнения задания:-----
- ThreadGroup
- Cookie manager
- Post Processor => Boundary extractor ИЛИ RegExp extractor
- Использование \${переменных}
- Throughput Controller или If controller (для покупки билета туда-обратно в половине случаев)

Задачи повышенной сложности:

- Обеспечить, что бы при покупке билета гарантированно не был куплен билет, где город отправления и прибытия совпадает.
- Разработать скрипт, переходящий на страничку с купленными билетами и отменяющий первый из них (важно учесть, скрипт должен работать вне зависимости от количества купленных билетов)

-----Элементы Jmeter, необходимые для выполнения задания:-----

- jsr223 Sampler
- Опционально: ForEach Controller
- while controller

=====  
 =====  
 =====Проведение  
 теста=====

- Необходимо создать 2 тред-группы, содержащей по одному http-семплеру:  
 Первая открывает ya.ru  
 Вторая открывает google.com
- Необходимо подать следующую нагрузку:  
 В течение 5 минут разгоняемся до 100% профиля  
 В течение 10 минут подаём равномерную нагрузку в 100% профиля  
 В течение 5 минут разгоняемся до 120% профиля  
 В течение 10 минут подаём 120% профиля

100% профиля выглядит так:  
 ya.ru = 60 запросов в минуту  
 google.com = 30 запросов в минуту

По итогам необходимо прислать графики:

- Количество операций в минуту в течение всего теста
- Времена отклика в ходе теста
- цифрами написать, какие были времена отклика и сколько операций было выполнено на последней ступени теста (120%)

7	<b>Мониторинг: ClickHouse и Docker</b>	описать основы мониторинга; настроить Докер; поднять кликхаус; записать метрики из Jmeter в ClickHouse.
8	<b>Автоматизация НТ: CI / CD и quality gate</b>	определить критерии успешности тестов (Quality gate); создать и запустить Job и Pipeline; использовать Jenkins-plugin для запуска сценариев MF Performance Center (PC); параметризовать запуск Pipeline; подключить Pipeline из Git; создать чек-лист запуска тестов через Pipeline; добавить кастомные скрипты для сбора и анализа результатов теста в Job.
9	<b>Скрипты и сценарии НТ - 3: Gatling часть 1/2</b>	объяснить базовые действия: установка, базовые компоненты, вывод результатов в файл, типы сценариев; рассмотреть Throttling, запись в influxdb и вывод графиков в Grafana.
10	<b>Скрипты и сценарии НТ - 3: Gatling часть 2/2</b>	использовать шаринг данных, собирать docker образ, запускать тесты в container'e; объяснить использование кастомных плагинов: нагрузка grpc сервиса.

#### Домашние задания

##### 1 Разработка скрипта и проведение теста в Gatling

Цель: Разрабатываем скрипт в средстве НТ, подготавливаем сценарий, проводим тест и анализируем результаты

У ДЗ 2 части:

Есть сайт - sushishop. <http://185.177.93.224:3000/>

Необходимо реализовать 3 метода

1. Получение списка заказов (/api/orders get)
2. Оформление заказа (/api/orders post)
3. Открытие сайта (/)

Целевые интенсивности - 100% нагрузки:

1. 50 запросов в секунду
2. 20 запросов в секунду
3. 100 запросов в секунду

Необходимо:

1. Реализовать ступенчатый тест от 100 - 200% с шагом в 20% используя Gatling(длительность теста любая, лучше не меньше 20 минут). Сборщик любой. В результате должен быть код сценария + html отчет о тестах.(или ссылка на grafana)
2. Реализовать тест надежности от 80% от базовой нагрузки в течение часа.
3. Нагрузить GRPC сервис(proto файл в материалах урока, адрес 185.233.0.230:50051 может измениться, проверяйте

11 **Скрипты и  
сценарии НТ - 3:  
кб.ио часть 1/2**

объяснить базовые действия: установка, компоненты, главная идея инструмента, вывод результатов, типы сценариев; записать в influxdb и вывести данные в Grafana.

---

12 **Скрипты и  
сценарии НТ - 3:  
кб.ио часть 2/2**

настроить и запустить нагрузочные тесты через GitlabCI и Github Actions вместе со сборкой микросервиса.

Домашние задания

1 Разработка скрипта и проведение теста в кб.ио

Цель: Разрабатываем скрипт в средстве НТ, подготавливаем сценарий, проводим тест и анализируем результаты

1. Разработать нагрузочные тесты на кб и провести тестирование сушишоп локально -> Результаты в Influx, ссылку на Grafana
2. Разработать workflow на Github Actions и запустить 1 из тестов на сушишоп из Actions -> Результаты ссылкой на воркфлоу



## 3 Эмуляторы, мониторинг, регрессионное НТ

- 1 Разработка базовых эмуляторов внешних систем**

разработать эмулятор REST web-сервиса, на wiremock.  
развернуть эмулятор.

---
- 2 Мониторинг и узкие места: Сервера приложений и очереди**

настроить мониторинг JVM;  
настроить мониторинг MQ (очереди);  
проанализировать полученные результаты.

Домашние задания

  - 1 Настройка мониторинга в Grafana + JVM

Цель: Применим знания по работе с InfluxDB, ElasticSearch, Grafana, Kibana на практике

  1. Используя прошлые наработки по настройке InfluxDB и Grafana - проведите любой из предыдущих тестов в Gatling и выведите результаты в Grafana.
  2. Поднимите ELK Stack, проведите любой из предыдущих тестов в Jmeter и выведите файловый лог Jmeter'a в ElasticSearch, чтобы можно было из Kibana по нему делать запросы

---
- 3 Мониторинг и узкие места: СУБД**

объяснить основные БД (MS SQL, Oracle, PostgreSQL) и их особенности для проведения НТ;  
настроить мониторинг БД;  
проанализировать узкие места по БД.

---
- 4 Критерии качества НТ, регрессионное НТ и риски**

определить слабые места текущего процесса по НТ и что можно улучшить;  
определить критичные тесты для регрессионного НТ, в том числе при нехватке времени;  
оценить риски производительности до начала НТ / по результатам НТ.

Домашние задания

  - 1 Подготовка заключения об НТ

Цель: Провести сравнение результатов 2-х версий системы, подготовить заключение о регрессионном тестировании

  1. Проведите эталонный тест сайта, расположенного на вашей машине (по-умолчанию адрес `https://ВАШ_IP/Account/Login`)
  2. Установите новый релиз сайта! :) Для этого внести ограничение по `gpc` для сайта (конфиги и инструкция по изменению настроек `nginx` в материалах к уроку)
  3. Проведите регрессионный тест сайта
  4. Подготовьте заключение о релизе сайта, по шаблону, дайте ваши выводы, рекомендации.

\*по желанию можете подготовить мини-отчёт или  
экспресс-отчет - ведь весь мониторинг в ваших руках!

- |   |   |  |
|---|---|--|
| 1 | <b>Выбор темы и организация проектной работы</b>    | выбор темы и организация проектной работы.<br>Домашние задания   |
|   |   | <p>1 Проект</p> <p>Цель: В проекте будет разработана методика и профиль нагрузочного тестирования, скрипты и сценарии, проведены тесты и подготовлен отчет о нагрузочном тестировании</p> <p>См. презентацию к уроку</p> <hr/> |
| 2 | <b>Консультация по проектам и домашним заданиям</b> | получить ответы на вопросы по проекту, ДЗ и по курсу.  |
| 3 | <b>Защита проектных работ</b>                       | защитить проект и получить рекомендации экспертов.   |