Задание для студентов корпоративного института.

# Подготовка методики нагрузочного тестирования (МНТ).

**Задание:**

1. Вписать «Название системы» и «версия системы» на первой странице.
2. Заполнить «Лист согласования», (Нагрузочное тестирование, Инженер-тестировщик, Куликов В.В.).
3. Заполнить таблицу «История изменений».
4. Перечислить перечень проводимых испытаний в нашем случае это:
   1. «Тест поиска максимума»
   2. «Тест подтверждения максимальной производительности»
   3. «Тест надёжности»
5. Актуализировать раздел «Введение».
6. Заполнить раздел «Цели тестирования».
7. Заполнить раздел «Ограничения тестирования», если по вашему мнению они есть.
8. Заполнить раздел «Объект тестирования» и его подразделы. В данном разделе должна присутствовать таблица с характеристиками хоста на котором проводится тестирование (ОС, версия ОС, характеристики железа, CPU, количество процессоров и ядер, объем оперативной памяти и т.д.). Так же должно присутствовать описание тестируемого приложения (см. Шаблон МНТ).
9. Ознакомится с разделом «Стратегия тестирования», актуализировать раздел согласно вашей стратегии тестирования и привести форматирование к общему виду.
10. Ознакомится с разделом «Моделирование нагрузки», актуализировать раздел согласно вашему моделированию нагрузки и привести форматирование к общему виду.
    1. Ознакомится с подразделом «Обзор», актуализировать раздел.
    2. Заполнить подраздел «Профиль нагрузки» информацией из таблицы расчета профиля (раздел заполняется после разработки скриптов и подсчета профиля).
    3. Заполнить подраздел «Сценарии использования» информацией о юзер кейсах.
11. Заполнить раздел «Планируемые тесты».
12. Заполнить раздел «Мониторинг».
13. Заполнить раздел «Материалы подлежащие сдачи». В столбце «дата готовности» указать дату сдачи Итогового домашнего задания.
14. Ознакомится с остальными разделами и привести форматирование к общему виду.

Комментарии: все делается на курсе «Введение в НТ» кроме п. 7.2 Профиль НТ.

# Разработка средств нагрузочного тестирования СНТ

## Создание юзер кейсов.

**Задание: создать 5 юзер кейса.**

Пример юзер кейса поиска и оформления билета:

1. Зайти на страницу логина, ввести учетные данные пользователя, нажать кнопку «Login»
2. Нажать кнопку «Flights»
3. Выполнить поиск рейса, нажать кнопку «Continue»
4. Выбрать один из рейсов, нажать кнопку «Continue».
5. Заполнить данные о платеже, нажать кнопку «Сontinue».
6. Нажать кнопку «Itinerary»
7. Нажать кнопку Sign Off

Комментарии: делается на курсе «Введение в НТ».

## Разработка скриптов нагрузочного тестирования (Vusers script)

**Задание: разработать 5 скриптов в соответствии с юзер кейсами из предыдущего пункта.**

1. **Оформление билета.**
2. **Поиск билетов**
3. **Логин**
4. **…**

Краткая инструкция по созданию скрипта:

1. WebTours должен быть уже запущен.
2. Запустить Vugen.
3. Создать новый скрипт.
4. Проверить настройки записи (корректные можно посмотреть в презентации по курсу).
5. Начать запись скрипта.
6. Всю бизнес операцию заключаем в транзакцию 01\_BuyTicket
7. Каждый шаг из юзер кейса заключить в транзакцию с соответствующим именем.

* Не забывайте закрывать транзакции.

1. Выполнит корреляцию скрипта, таким образом, чтобы запросы выполнялись успешно. Например скоррелировать userSession, что бы пользователь смог залогиниться в системе.
2. Параметризовать скрипт так, чтобы при каждой итерации заполнялись разные поля на странице.
3. Выставить think\_time (5 секунд).

## Подготовка Профиля для проведения НТ.

**Задание: посчитать профиль используя шаблон.**

1. Заполнить столбец «Операция» названиями скриптов.
2. Заполнить столбец «max. Время выполнения одного скрипта». Что бы получить данную величину нужно прогнать скрипт 3 раза (запустить без учета think\_time, в runtime\_settings -> Think time -> ignore think time) и выбрать наибольшее время выполнения (view -> output -> в самом конце ”Duration” ).



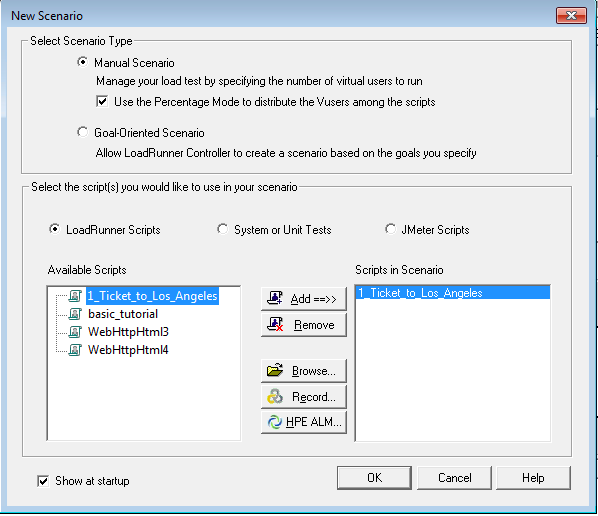
1. Заполнить столбец «Think Time». Что бы получить данную величину нужно прогнать скрипт 3 раза (запустить с учетом think\_time, в runtime\_settings -> Think time -> Replay think time as recorded) выбрать наибольшую и величину выполнения (view -> output -> в самом конце ”Duration” ) и вычесть из нее «max. время выполнения скрипта» полученное шагом ранее Результат это и есть Think time!.
2. Заполнить столбец «Количество VU». Распределить пользователей между скриптами самостоятельно, учитывая, что максимальное количество пользователей на первой ступени 10.
3. Вписать формулу в желтый столбец интенсивность, что бы рассчиталась интенсивность для остальных ступеней.

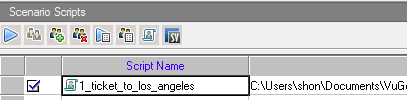
# Проведение итерации нагрузочного тестирования.

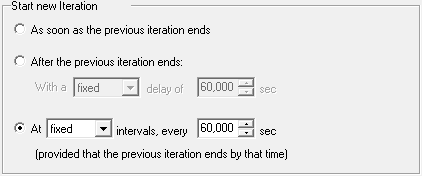
## Создание сценария тестов в Controller

**Задание: создать сценарии теста поиска максимальной производительности и теста надежности.**

Краткая инструкция по созданию сценария:

1. Открыть Controller
2. Создать новый сценарий File -> New Scenario
3. Настройки нового сценария должны соответствовать скриншоту ниже
4. Выставить Run-time Settings:

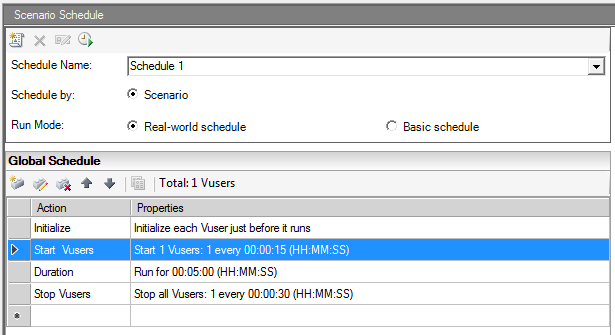


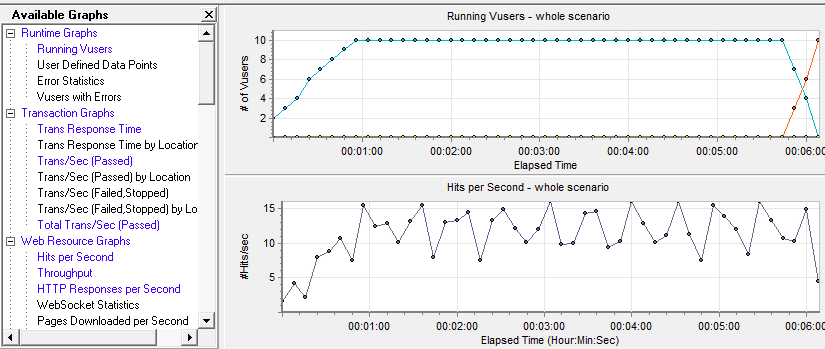
* 1. General -> Run Logic, количество итераций 1
  2. General -> Pacing, значение выставить из профиля. 
  3. General -> Think Time, Replay think time -> As recorded

1. Настроить Global Schedule:

Обратите внимание, что для теста максимума на каждой ступени должен быть интервал стабильной нагрузки, не считая времени входа пользователей согласно значениям, указанным в профиле (20 минут), Duration лучше указывать с небольшим запасом, 2-4 минуты.

Для теста максимума должен быть также интервал стабильной нагрузки длительностью 2 часа, не считая времени входа пользователей.



1. Настроить графики на вкладке Run 
   1. Добавить на график «Windows Recourses» следующие метрики:

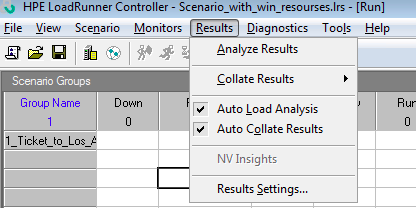
|  |
| --- |
| Windows - основные счётчики процессора |
| * % Processor Time * [% User Time](http://devopswiki.net/index.php/%25_Processor_Time_Counter_(%D1%81%D1%87%D1%91%D1%82%D1%87%D0%B8%D0%BA_%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B6%D0%B5%D0%BD%D0%BD%D0%BE%D1%81%D1%82%D0%B8_%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D1%80%D0%B0)) * [Processor Queue Length](http://devopswiki.net/index.php/Processor_Queue_Length_Counter_(%D1%81%D1%87%D1%91%D1%82%D1%87%D0%B8%D0%BA_%D0%B4%D0%BB%D0%B8%D0%BD%D1%8B_%D0%BE%D1%87%D0%B5%D1%80%D0%B5%D0%B4%D0%B8_%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D1%80%D0%B0)) |
| Windows - основные счётчики памяти |
| * Available MBytes * [Committed Bytes In Use](http://devopswiki.net/index.php/Committed_Bytes_Counter_(%D1%81%D1%87%D1%91%D1%82%D1%87%D0%B8%D0%BA_%D0%B2%D1%8B%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%BD%D1%8B%D1%85_%D0%B1%D0%B0%D0%B9%D1%82_%D0%B2%D0%B8%D1%80%D1%82%D1%83%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B9_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D0%B8)) * [Pages Faults/sec](http://devopswiki.net/index.php/Pages/sec_Counter_(%D1%81%D1%87%D1%91%D1%82%D1%87%D0%B8%D0%BA_%D0%BF%D0%BE%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%87%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BE%D0%B1%D0%BC%D0%B5%D0%BD%D0%B0_%D0%B2_%D1%81%D0%B5%D0%BA%D1%83%D0%BD%D0%B4%D1%83)) * Pages/sec |
| Windows - основные счётчики ввода/вывода |
| * Avg. Disk secs/Read * Avg. Disk secs/Write * Avg. Disk Queue Length * Current Disk Queue Length * % [Disk Time](http://devopswiki.net/index.php/Disk_Transfers/sec_(%D0%BE%D0%B1%D1%80%D0%B0%D1%89%D0%B5%D0%BD%D0%B8%D0%B9_%D0%BA_%D0%B4%D0%B8%D1%81%D0%BA%D1%83/%D1%81%D0%B5%D0%BA)) |
| Windows - основные сетевые счетчики |
| * Bytes Total/sec * BytesReceived/sec (в Grafana) * BytesSent/sec (в Grafana) |

Нажать пкм на график -> выбрать «add Measurements» -> добавить необходимое.

1. Сохранить сценарий File -> Save As

## Проведение тестов

**Задание:**

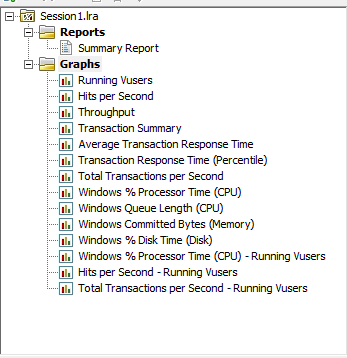
1. Провести тест поиска максимальной производительности. Сценарий теста и нагрузка должна соответствовать профилю. Провести краткий анализ, на основе которого определить нагрузку для теста надежности/подтверждения.
2. Провести тест надежности/подтверждения. Нагрузка должна соответствовать разделу 6.1.2 МНТ пункту первому «тестирование целевых требований системы». В нашем случае тест надежности будет длится 2 часа.
3. Что бы collate и analysis выполнился автоматически по завершении теста, выставить следующие настройки в Controller:

## Анализ результатов НТ

**Задание: проанализировать 2 сессии, тест поиска максимума и тест надежности.**

Тест поиска максимума:

1. Применить фильтр для SummaryReport по времени за ступень определенного максимума.
2. Применить фильтр для SummaryReport по необходимым названиям транзакций.
3. Построить следующие графики:



1. Применить минимальную гранулярность.
2. Сохранить проанализированную сессию.

Тест надежности/подтверждения:

1. Применить фильтр для SummaryReport по времени за стабильный интервал нагрузки.
2. Выполнить пункты 2-5 из инструкции для теста максимума.

## Подготовка отчета о проведении НТ.

**Задание: подготовить отчет о проведенном тестировании.**

1. Заполнить раздел 2 в соответствии с МНТ.
2. Заполнить раздел 3 «Выводы» подробнее в шаблоне отчета.
3. Заполнить раздел 4 «Графики и таблицы» подробнее в шаблоне отчета.

Как получить количество успешных запросов для выводов: суммировать значения столбца Pass из SummaryReport.