*“Optimizing the performance of computer systems has always been an art relegated to a few individuals who happen to have the ‘right skills’.”*

**Amir H. Majidimehr**  
Optimizing Unix for Performance, 1995



**Методики нагрузочного тестирования**

Версия 8.6

Оглавление

[1 История внесения изменений в документ 4](#_Toc158211964)

[2 Лист согласования 5](#_Toc158211965)

[3 Список терминов и сокращений 6](#_Toc158211966)

[4 Введение 7](#_Toc158211967)

[4.1 Назначение документа 7](#_Toc158211968)

[4.2 Объект тестирования 7](#_Toc158211969)

[4.2.1 База данных PostgreSQL 7](#_Toc158211970)

[4.2.2 Сервис для отправки email-сообщений 8](#_Toc158211971)

[4.2.3 UXCrowd-frontend 8](#_Toc158211972)

[4.2.4 UXCrowd-backend 8](#_Toc158211973)

[5 Цели и задачи 9](#_Toc158211974)

[6 Ограничения тестирования 10](#_Toc158211975)

[7 Архитектура системы 11](#_Toc158211976)

[7.1 Конфигурация серверов продуктивного стенда 11](#_Toc158211977)

[8 Тестовый стенд 12](#_Toc158211978)

[8.1 Архитектура тестового стенда 12](#_Toc158211979)

[8.2 Конфигурация тестового стенда 12](#_Toc158211980)

[8.3 Конфигурация ПО 12](#_Toc158211981)

[8.4 Тестовые данные для средств НТ 13](#_Toc158211982)

[9 Взаимодействие с внешними системами 14](#_Toc158211983)

[9.1 Эмулятор сервиса для отправки email-сообщений 14](#_Toc158211984)

[10 Стратегия тестирования 15](#_Toc158211985)

[10.1 Этапы тестирования 15](#_Toc158211986)

[11 Моделирование нагрузки 16](#_Toc158211987)

[12 Анализ статистики 18](#_Toc158211988)

[12.1 Наиболее часто используемые операции 18](#_Toc158211989)

[13 Профили нагрузки 19](#_Toc158211990)

[14 Сценарии пользования 20](#_Toc158211991)

[15 Наполнение БД 22](#_Toc158211992)

[16 Планируемые тесты 23](#_Toc158211993)

[16.1 Планируемые тесты 23](#_Toc158211994)

[16.2 Критерии успешности проведенного теста 23](#_Toc158211995)

[17 Требования к производительности 24](#_Toc158211996)

[18 Мониторинг производительности 25](#_Toc158211997)

[18.1 Метрики производительности 25](#_Toc158211998)

[18.2 Способы мониторинга показателей производительности 26](#_Toc158211999)

[19 Риски проекта 27](#_Toc158212000)

[20 Требования к заказчику 28](#_Toc158212001)

[21 Материалы, подлежащие сдаче 29](#_Toc158212002)

[22 Контакты 30](#_Toc158212003)

# История внесения изменений в документ

|  |  |  |  |
| --- | --- | --- | --- |
| Дата | Версия | Описание | Автор |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Лист согласования

|  |  |  |  |
| --- | --- | --- | --- |
| ФИО | Должность | Подпись | Дата |
| Матвеев Александр Павлович |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Список терминов и сокращений

|  |  |
| --- | --- |
| Термин | Полное наименование |
| НТ | Нагрузочное тестирование |
| БД, DB | База данных |
|  |  |
|  |  |
|  |  |

# Введение

## Назначение документа

Методика НТ представляет собой подробное описание технологии нагрузочного тестирования систем сервиса «UXCrowd».

Документ преследует следующие цели:

* описать стратегию тестирования производительности систем;
* описать планируемые этапы работ;
* описать объект исследования и конфигурацию тестового стенда;
* описать порядок передачи результатов проекта;
* описать рамки и ограничения тестирования;
* описать методики тестирования.

Методика НТ предназначена для специалистов бизнес-подразделений, менеджеров и технических специалистов Заказчика, а также будет использоваться при проектировании и проведении тестов специалистами «Перформанс Лаб».

## Объект тестирования

Объектом тестирования является облачный сервис для удаленных немодерируемых тестирований на пользователях. Сервис состоит из:

* БД PostgreSQL;
* Сервис для отправки email-сообщений;
* UXCrowd-frontend;
* UXCrowd-backtend;

### База данных PostgreSQL

БД PostgreSQL обеспечивает:

* хранение информации о пользователях, тестировщиках и т.д.;
* хранение информации о платежах, ценах и заказах;
* хранение информации о записанных тестировщиками видео;
* хранение информации о проведенных тестах.

### Сервис для отправки email-сообщений

Сервис предназначен для оповещения пользователей и тестировщиков через электронную почту. В ходе работы сервис выполняет следующие задачи:

* отправка паролей на электронную почту зарегистрированным пользователям;
* оповещение пользователей и тестировщиков о деталях оплаты;
* информирование пользователей о важных нововведениях;
* рассылка рекламы.

### UXCrowd-frontend

Frontend часть сервиса для удаленных немодерируемых тестирований на пользователях UXCrowd может выполняет следующие задачи:

* предоставление пользователю интерфейса для регистрации и авторизации на платформе UXCrowd;
* предоставляет пользователям доступные тестирования и возможность выбора интересующих их заданий;
* обеспечивает возможность заполнения и отправки результатов тестирования;
* отображает статус выполнения тестовых заданий и результатов прошлых тестирований;
* предоставляет интерфейс для взаимодействия с другими функциональными компонентами сервиса UXCrowd, такими как система обратной связи и аналитический модуль;

отображает уведомления и важные сообщения от сервиса.

### UXCrowd-backend

UXCrowd-backend отвечает за внутреннюю логику сервиса и выполняет следующие функции:

* регистрация и аутентификация пользователей, управление информацией о профилях пользователей и тестировщиков;
* управление базой данных тестовых заданий, включая создание, обновление и удаление заданий;
* обработка запросов от фронтенда, включая принятие и валидацию данных, выполнение бизнес-логики и обновление базы данных;
* взаимодействие с другими компонентами системы, такими как Сервис отправки email‑сообщений;
* обеспечение безопасности данных, защиту от атак и обеспечение конфиденциальность.

# Цели и задачи

Нагрузочное тестирование преследует следующие цели:

1. Определение максимальной нагрузки, которую система может выдержать без значительного снижения производительности.
2. Оценка надежности и устойчивости системы при ограниченной или повышенной нагрузке.
3. Выявление узких мест, проблем производительности, сбоев или неожиданного поведения системы под нагрузкой.
4. Измерение временных характеристик процессов (например, время отклика) при различных уровнях нагрузки.
5. Оценка поведения системы при изменении количества одновременных пользователей или нагрузки.
6. Проверка масштабируемости системы и ее способности обрабатывать растущую нагрузку.
7. Анализ существующих проблем тестируемой среды. Предоставление рекомендаций по оптимизации.
8. Выявление возможных рисков, которые могут повлиять корректность результата.
9. Тестирование системы на стабильность.

К основным задачам нагрузочного тестирования относятся:

* Определение профилей нагрузки, исходя из типовых операций, выполняемых пользователями.
* Анализ интеграционных связей, с целью определения систем и операций, которые нужно эмулировать в процессе нагрузочного тестирования.
* Составление существенных операций и процессов, которые будут включены в профиль нагрузки.
* Определение границ и предельных значений нагрузки для проверки производительности и надежности системы.
* Описание структуры стенда нагрузочного тестирования.
* Проведение испытаний в тестовой среде.
* Подготовка отчетов по результатам тестов.
* Проведение сравнительного анализа результатов тестирования нового релиза с результатами тестирования предыдущих релизов информационной системы.
* Анализ результатов нагрузочного тестирования и выявление проблемных мест для дальнейшей оптимизации.

# Ограничения тестирования

В рамках проводимого нагрузочного тестирования следует отметить следующие ограничения:

* Данное тестирование не является функциональным и не служит для выявления функциональных дефектов, в то же время, обнаруженные в ходе проведения работ дефекты регистрируются и передаются Заказчику.
* Для прохождения успешного прохождения некоторых тестов будут внесены изменения в БД.
* Тестирование не направлено на выявление дефектов в аппаратной части стенда.
* Перед проведением тестирования на этапе создания нагрузочных скриптов версии компонент информационной системы фиксируются и не изменяются до окончания тестирования, за исключением случаев устранения ошибок, мешающих дальнейшему проведению работ по тестированию.
* Тестирование производится с учетом того, что со стороны Заказчика не предоставлены доступы и контакты серверов продуктивного стенда.
* Для корректного поведения сценариев нагрузочного тестирования в них вносятся изменения, позволяющие миновать ошибки со стороны продуктивного стенда путем прямого взаимодействия с базой данных.
* Объем данных в БД на котором производится тестирование не соответствует промышленному серверу из-за отсутствия доступа к промышленному серверу. Пред началом тестирования была проведена частичная чистка таблиц.

# Архитектура системы

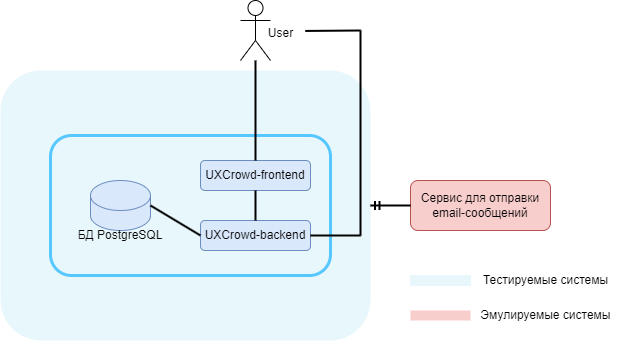


Рисунок – Компонентная архитектура собственного процессинга

## Конфигурация серверов продуктивного стенда

Заказчик не предоставил доступы и контакты серверов продуктивного стенда.

# Тестовый стенд

## Архитектура тестового стенда

Архитектура тестового стенда включает в себя нагрузочную станцию c JMeter, развернутую на виртуальной машине.

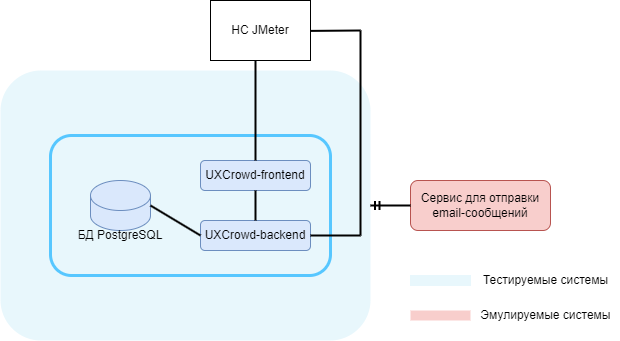


Рисунок – Архитектурная схема тестового стенда

## Конфигурация тестового стенда

Таблица 1 – Конфигурация тестового стенда

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Тип/имя сервера | Конфигурация | Кол-во в прод. | Кол-во для нагрузочного тестирования | ОС |
| Сервер БД и сервер приложений  Dev-lt.uxcrowd.ru | VMware Virtual Platform –  Intel Xeon E5620 1xCore,  5 CPU count,  7.6 Gb RAM,  195 Gb HDD | 1 | Windows Server 2012 R2 | Сервер БД и сервер приложений  Dev-lt.uxcrowd.ru |

## Конфигурация ПО

Тестирование проводится на системе Linux, база данных является копией промышленной. Тестирование будет произведено на среде, наполненной до целевых объемов.

## Тестовые данные для средств НТ

Генерация тестовых данных со стороны Performance Lab производится с помощью скриптов в Apache JMeter. Генерируемые данные включают в себя информацию о:

* пользователях разработчиках/клиентах;
* тестовых сценариях;
* видео проведенного теста.

# Взаимодействие с внешними системами

## Эмулятор сервиса для отправки email-сообщений

Для эмуляции взаимодействия системы с сервисом для отправки сообщений используется дополнительная логика в сценарии Apache JMeter.

# Стратегия тестирования

## Этапы тестирования

Проект проведения нагрузочного тестирования делится на следующие этапы:

* создание методики (данный документ);
* разработка средств, эмулирующих нагрузку на процессинг и подготовка среды тестирования;
* проведение тестирования;
* системный анализ;
* подготовка отчета.

Критерии успешного завершения тестирования:

* достигнута повторяемость результатов тестов;
* выполнены все запланированных тестов;
* получены данные мониторинга;
* измерены времена отклика интерфейса;
* указываются дополнительные критерии успешности тестирования.

# Моделирование нагрузки

Для проведения тестирования будут разработаны средства нагрузочного тестирования (НТ). В данном разделе описаны требования к средствам НТ.

Средства НТ разрабатываются с использованием ПО Apache JMeter 5.6.3, предназначенного для создания тестов и проведения тестирования.

Эмулятор системы разрабатываются на JDK 7 и запускаются в виде JAVA приложения.

Для подачи нагрузки используются нагрузочные скрипты. Все сценарии нагрузочных скриптов запускается одновременно.

При записи скриптов воспроизводятся статические элементы (.svg, .js, .css).

Интенсивность выполнения сценария каждым пользователем зависит от сценария, времен отклика системы и величины задержки между двумя последовательными итерациями (между началами выполнения двух последовательных сценариев). В процессе тестирования время выполнения сценария и время отклика системы ограничивается сверху некоторым порогом и считаются постоянными, а изменение интенсивности выполнения сценария будет производиться посредством изменения количества потоков и изменением задержек между итерациями.

Таким образом, если сумма времени отклика системы и времени выполнения сценария не превышает задержку между двумя последовательными итерациями, то увеличение количества виртуальных пользователей (потоков), выполняющих сценарии одновременно, повлечет пропорциональное увеличение суммарной интенсивности.

В процессе тестирования изменение суммарной интенсивности выполнения сценариев всеми виртуальными пользователями (моделируемая нагрузка на ИС) будет производиться путем изменения количества виртуальных пользователей, выполняющих сценарии, и величины задержки между последовательными итерациями.

Величина задержки и количество виртуальных пользователей, выполняющих различные сценарии, рассчитываются с использованием Excel шаблона на этапе подготовки стенда и средств НТ после написания скриптов и определения времени их работы в ИС, не испытывающей нагрузку.

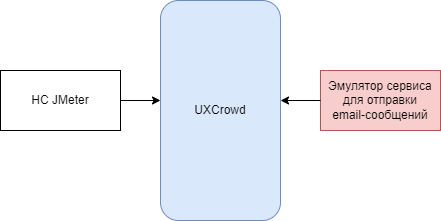


Рисунок - Схема генерации нагрузки

# Анализ статистики

## Наиболее часто используемые операции

На основании проведенного анализа данных БД PostgresSQL в период с 1 декабря по 31 декабря 2023 года (исходя из требований Заказчика), был выбран час пиковой нагрузки для наиболее часто используемых операций.

В таблице ниже представлен список наиболее часто используемых операций

Таблица 2 – Список операций с разбивкой по типам по типам в ЧПН

|  |  |  |
| --- | --- | --- |
| **№** | **НАЗВАНИЯ ОПЕРАЦИЙ** | **КОЛИЧЕСТВО** |
| 1 | Создание нового теста | 2787 |
| 2 | Загрузка видео | 1588 |
| 3 | Регистрация тестировщика | 1585 |
| 4 | Регистрация клиента | 770 |
|  | **Общий итог** | **6730** |

# Профили нагрузки

Профиль нагрузки включает в себя операции регистрации пользователей, создания нового теста и загрузки видео проведения теста. Выполнение операций производится единовременно. Описание операций приведено в таблице 3.

Таблица 3 – Список операций нагрузочного профиля

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Название операции | Распределение % | Средняя нагрузка | Пиковая нагрузка в час |
| 1 | Создание нового теста | 41% | 1394 | 2787 |
| 2 | Загрузка видео | 24% | 794 | 1588 |
| 3 | Регистрация тестировщика | 24% | 792 | 1585 |
| 4 | Регистрация клиента | 11% | 385 | 770 |
|  | **Итого** | **100,0%** | **3365** | **6730** |

# Сценарии пользования

Для эмуляции процесса записаны скрипты, которые симулируют выполнение следующих пользовательских операций. Каждый скрипт разбит по операциям на отдельные транзакции.

Таблица 4 – Перечень эмулируемых операций

|  |  |  |
| --- | --- | --- |
| ID | Название операции | Название скрипта |
| UC01 | Регистрация тестировщика | UC\_01\_tester\_registration |
|  | Открытие главной страницы https://dev-lt.uxcrowd.ru | UC\_01\_TR\_01\_website\_opening |
|  | Нажатие на кнопку “Войти” | UC\_01\_TR\_02\_signin\_click |
|  | Регистрация тестировщика | UC\_01\_TR\_03\_registration\_as\_tester |
| UC02 | Регистрация клиента | UC\_02\_customer\_registration |
|  | Открытие главной страницы https://dev-lt.uxcrowd.ru | UC\_02\_TR\_01\_website\_opening |
|  | Нажатие на кнопку “Войти” | UC\_02\_TR\_02\_Signin\_click |
|  | Регистрация клиента | UC\_02\_TR\_03\_Registration\_as\_  customer |
| UC03 | Создание новых тестов | UC\_03\_new\_test\_creating |
|  | Открытие главной страницы https://dev-lt.uxcrowd.ru | UC\_03\_TR\_01\_website\_opening |
|  | Нажатие на кнопку “Войти” | UC\_03\_TR\_02\_button\_login |
|  | Заполнение полей почты и пароля | UC\_03\_TR\_03\_login |
|  | Создание нового теста. Выбор типа теста “Немодерируемое юзабилити-тестирование” | UC\_03\_TR\_04\_test\_creating |
|  | Описание теста | UC\_03\_TR\_05\_test\_creating - description\_of\_test |
|  | Составление задания | UC\_03\_TR\_06\_test\_creating - drawing\_up\_task |
|  | Выбор аудитории | UC\_03\_TR\_07\_test\_creating - audience\_selection |
|  | Запуск теста | UC\_03\_TR\_08\_test\_creating - running\_test |
|  | Выход из аккаунта | UC\_03\_TR\_09\_logout |
| UC04 | Прохождение новых тестов | UC\_04\_new\_test\_pass |
|  | Открытие главной страницы https://dev-lt.uxcrowd.ru | UC\_04\_TR\_01\_website\_opening |
|  | Нажатие на кнопку “Войти” | UC\_04\_TR\_02\_button\_login |
|  | Заполнение полей почты и пароля | UC\_04\_TR\_03\_login |
|  | Заполнение данных профиля | UC\_04\_TR\_04\_profile\_data |
|  | Нажатие на кнопку “Пройти” для прохождение теста | UC\_04\_TR\_05\_button\_pass |
|  | Нажатие на кнопку “Настроить запись экрана” | UC\_04\_TR\_06\_button\_set\_up\_screen\_recording |
|  | Выбор части экрана | UC\_04\_TR\_07\_button\_screen\_selection |
|  | Нажатие на кнопку “Начать тест” | UC\_04\_TR\_08\_button\_start\_test |
|  | Ознакомление с описанием теста, нажатие на кнопку «Продолжить» | UC\_04\_TR\_09\_button\_ continue |
|  | Выполнения задания и нажатие на кнопку закончить | UC\_04\_TR\_10\_button\_finish\_test |
|  | Отправление записанного видео | UC\_04\_TR\_11\_sending\_video |
|  | Подтверждение уведомления о загрузки видео, нажатие на кнопку “ОК” | UC\_04\_TR\_12\_download\_confirmation |
|  | Выход из аккаунта | UC\_04\_TR\_13\_logout |

# Наполнение БД

Перед началом тестирования была проведена частичная очистка таблиц БД. Ниже приведена таблица с наиболее объемных таблиц БД после чистки.

Таблица 5 – Перечень наиболее объемных таблиц БД

|  |  |  |  |
| --- | --- | --- | --- |
| Имя таблицы | Общий таблицы | Общий индекса | Общий объем |
| users | 151 MB | 101 MB | 252 MB |
| interview\_audio | 110 MB | 11 MB | 120 MB |
| test\_pass\_stats | 45 MB | 16 MB | 61 MB |
| payment | 27 MB | 21 MB | 48 MB |
| step | 37 MB | 11 MB | 47 MB |
| customer\_init\_wish | 27 MB | 5416 | 33 MB |
| shedlock | 31 MB | 1072 | 32 MB |

Наполнение БД POSTGRES сущностями производится нагрузочными скриптами. В сценарии UC\_01 производится создание пользователя-тестировщика. UC\_02‑ создание пользователя - клиента. UC\_3‑составление тестового сценария, для прохождения его в сценарии UC\_4. UC\_04 - добавление информации о видео проведения теста.

# Планируемые тесты

## Планируемые тесты

Таблица 6 – Перечень планируемых тестов

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Тест | Используемые профили | Длительность теста | Кол-во пусков | Описание теста |
| 1 | Определение максимальной производительности | 2%(от профиля Заказчика) + 2% на каждую ступень | 10 мин выход на каждую ступень + 10 минут подтверждения ступени | 1 | Определение максимальных рабочих параметров системы и ее компонентов. |
| 2 | Подтверждение максимальной производительности | 100% (от найденного максимума) | 60 мин | 1 | Определение способности системы поддерживать высокий уровень производительности. |
| 3 | Определение надежности | 70% (от найденного максимума) | 11 часов | 1 | Определение способности системы надежно работать без сбоев или ошибок. |
| 4 | Тестирование при 150% профиле | 150% (от найденного максимума) | 60 минут | 1 | Оценка реакции системы на увеличение нагрузки сверх предельного значения. |
| 5 | Тестирование при отказе БД | 75% (от найденного максимума) | 60 минут | 1 | Оценка работоспособности системы при отказе работы БД. |
| 6 | Тестирование сбоя | 75% (от найденного максимума) | 60 минут | 1 | Оценка работоспособности и отказоустойчивости системы после ее сбоя. |

## Критерии успешности проведенного теста

Тест считается успешным, если:

* в процессе тестирования запросы выполнялись с частотой, соответствующей профилю тестирования (в процессе тестирования возникло не более 5% ошибок);
* в процессе тестирования общая загрузка CPU составляет не более 70%;
* в процессе тестирования общая загрузка RAM составляет не более 80%;
* в процессе тестирования время выполнения каждого http - запроса составляет не более 5 сек.

# Требования к производительности

Таблица 7 – Требования к времени отклика по типам операций

|  |  |
| --- | --- |
| ОПЕРАЦИЯ | ПРОИЗВОДИТЕЛЬНОСТЬ (ОП/Ч) |
| Операционная деятельность | |
| Регистрация клиента | 770 |
| Регистрация тестировщика | 1585 |
| Создание нового теста | 2787 |
| Загрузка видео | 1588 |

Также в качестве требований к производительности системы определяются:

* утилизация процессорных мощностей системы не должна превышать 70% (рекомендуемый максимум);
* утилизация оперативной памяти системы не должна превышать 80% (рекомендуемый максимум);
* система должна быть способна обработать операции с интенсивностью, указанной в профиле нагрузки;
* средняя утилизация дисковой подсистемы должна не превышать 90%, время отклика дисковой подсистемы должно быть не более 15мс;
* время выполнения каждой транзакции должно составлять не более 5 сек.
* количество возникших в процессе тестирования ошибок не должно превышать 5%.

# Мониторинг производительности

Мониторинг производительности серверов будет включать наблюдение за системными параметрами и будет осуществляться при помощи таких утилит как Influx, Telegraf, Prometheus, Grafana. Интервал сбора статистики по метрикам мониторинга будет составлять 5 секунд.

## Метрики производительности

Таблица 8 – Метрики производительности

|  |  |  |  |
| --- | --- | --- | --- |
| № | МЕТРИКА | ЕДИНИЦА ИЗМЕРЕНИЯ | ОПИСАНИЕ |
| 1 | cpu | % | Показатель процессорного использования сервера. |
| 2 | disk\_iops | io/s | Число операций ввода/вывода, выполняемое дисковой подсистемой в секунду. |
| 3 | disk\_read | MB/s | Количество операций чтения данных с жесткого диска. |
| 4 | disk\_write | MB/s | Объем данных, записанных на диск в определенный период времени. |
| 5 | memory | GB;% | Показатель использования оперативной памяти сервера. |
| 6 | network\_recieve | MB/s | Скорость приема данных (трафика) через сетевой интерфейс сервера. |
| 7 | network\_transmit | MB/s | Количество данных, передающихся по сети. |
| 8 | Transaction response time | s | Время выполнения http-запроса |

Все показатели снимаются и анализируются за репрезентативные периоды выполнения тестов:

* за периоды плановой нагрузки на систему (без учета периодов теста, на которых происходит выход на нагрузку);
* за периоды насыщения системы и деградации производительности, если таковая будет иметь место.

## Способы мониторинга показателей производительности

Для снятия показателей производительности будут использоваться следующие инструменты мониторинга:

Таблица 9 – Типы метрик производительности

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **ТИП МЕТРИКИ** | **СПОСОБ СНЯТИЯ** | **СПОСОБ ПОЛУЧЕНИЯ** |
| 1 | Аппаратные ресурсы | Prometheus, Grafana (с использованием ОС семейства Windows)  Telegraf, Grafana (с использованием ОС семейства Windows) | Предоставляются специалистами заказчика по запросу исполнителя.  Фиксируются исполнителем. |
| 2 | Бизнес-показатели | Influx, Grafana (с использованием ОС семейства Windows) | Фиксируются исполнителем. |

# Риски проекта

* Методика тестирования может быть пересмотрена и скорректирована (по результатам получения актуальной информации, актуальных целей и т.д.). Все изменения предварительно согласовываются с Заказчиком в письменном виде.
* В случае недоступности тестового стенда на протяжении 4-х и более часов, сроки проведения Работ Исполнителем могут измениться.
* Результат выполнения Работ зависим от объема БД на период проведения тестирования
* Интерпретация результатов выполненных работ по отношению к продуктовому контуру будет осуществляться полностью на стороне Заказчика.

# Требования к заказчику

Заказчик обязуется:

* Предоставить Исполнителю доступ к тестируемой системе на уровне, достаточном для проведения Исполнителем Работ;
* Предоставить Исполнителю профиль нагрузки;
* Предоставить исполнителю прогнозы роста нагрузки на систему, если таковые имеются;
* Заниматься исправлением блокирующих дефектов, обнаруженных Исполнителем, таким образом, чтобы это не приводило к простою (частичному или полному) представителей Исполнителя, занятых выполнением Работ;
* Снабжать Исполнителя лицензиями ПО, необходимыми Исполнителю для выполнения Работ.

.

# Материалы, подлежащие сдаче

Таблица 10 – Типы метрик производительности

|  |  |  |
| --- | --- | --- |
| № | Артефакт | Формат предоставления |
| 1 | Исходные коды скриптов нагрузочного тестирования | Apache Jmeter |
| 2 | Отчет по результатам тестирования | MS Word |
| 3 | Результаты всех проведенных тестов | MS Office |

# Контакты

ООО «Перфоманс Лаб»

121087 Москва, ул. Барклая, 6, стр.5, офис 511

Телефон: +7 495 780 9228

Факс: +7 495 780 9228

[http://performance-lab.ru](http://performance-lab.ru/)

Генеральный директор: Кутузов Максим Юрьевич

Документ подготовили:

Минин Владимир Евгеньевич (minin.vladimir.90@gmail.com, 8-925-658-12-05)

Гайбович Алексей Николаевич (alexyng7@gmail.com, 8-922-684-43-21)

Нельзин Илья Юрьевич (illobikol@gmail.com, 8-982-990-76-72)