

УДК 004.054

## ***ИССЛЕДОВАНИЕ ОСОБЕННОСТЕЙ НАГРУЗОЧНОГО ТЕСТИРОВАНИЯ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ***

***Бевзенко С. А.***

*Старший разработчик,*

*ООО «Яндекс-Технологии»*

*Москва, Россия*

### **Аннотация**

В данной работе рассматриваются особенности нагрузочного тестирования мобильных приложений. Автор анализирует значимость производительности мобильных приложений для пользователей, выделяя аспекты, которые могут негативно повлиять на удовлетворенность и лояльность пользователей. Дополнительно рассматривается влияние нагрузки на потребление ресурсов и производительность мобильных приложений. Кроме того, в статье проведен анализ наиболее эффективных инструментов и программного обеспечения для проведения нагрузочного тестирования мобильных приложений.

**Ключевые слова:** нагрузочное тестирование, мобильные приложения, информационные технологии, сетевые технологии, программное обеспечение.

## ***RESEARCH OF THE FEATURES OF LOAD TESTING OF MOBILE APPS***

***Bevzenko S. A.***

*senior developer,*

*ООО "Yandex-Technologies"*

*Moscow, Russia*

**Abstract**

This paper discusses the features of load testing of mobile applications. The author analyzes the importance of mobile application performance for users, highlighting aspects that can negatively affect user satisfaction and loyalty. Additionally, the impact of load on resource consumption and performance of mobile applications is considered. In addition, the article analyzes the most effective tools and software for load testing mobile applications.

**Keywords:** load testing, mobile applications, information technologies, network technologies, software.

Нагрузочное тестирование мобильных сервисов имеет важное значение для обеспечения стабильности, производительности и надежности приложений в условиях высоких нагрузок. Оно позволяет выявить узкие места и проблемы, которые могут возникнуть при одновременном использовании приложения большим числом пользователей или при работе с данными больших объемов.

Исследование особенностей нагрузочного тестирования поможет улучшить качество мобильных приложений, повысить их производительность и обеспечить позитивный опыт пользователей систем. Эффективное нагрузочное тестирование предотвращает потенциальные проблемы и сбои, которые могут возникнуть при использовании приложения реальными пользователями в условиях повышенной нагрузки.

Тема нагрузочного тестирования мобильных приложений является актуальной по нескольким причинам. Прежде всего стоит отметить непрерывный рост популярности мобильных приложений, при котором производительное функционирование становится важным условием для решения задач, улучшения потребностей и ожиданий пользователей. Для их привлечения, разработчики должны предлагать стабильные, отзывчивые и производительные приложения. Недостатки в производительности и

Дневник науки | [www.dnevniknauki.ru](http://www.dnevniknauki.ru) | СМИ Эл № ФС 77-68405 ISSN 2541-8327

стабильности могут привести к низким оценкам, отзывам и уходу пользователей к конкурентам. Кроме того, фрагментация мобильного рынка приводит к существованию множества различных устройств и операционных систем. Это усложняет задачу в обеспечении процесса производительности всех устройств, совместимости операционных систем и их взаимодействия. Нагрузочное тестирование помогает выявить и идентифицировать данные проблемы, и дать разработчикам шанс исправить их до внедрения [8].

Научная новизна темы обусловлена постоянно меняющейся технологической средой разработки мобильных приложений, а также поисками новых методов и подходов для эффективного тестирования, и обеспечения высокой производительности мобильных приложений, особенно, которые предъявляют особые требования к производительности и стабильности.

В статье используется аналитический подход. Описываются рекомендации и методы, основанные на опыте и лучших практиках в области информационной безопасности, для проведения нагрузочного тестирования мобильных приложений.

Нагрузочное тестирование в мобильных приложениях имеет свои уникальные особенности, отличающие его от тестирования других видов систем. При использовании мобильных приложений часто происходит множество одновременных событий, создаваемых действиями различных игроков. При проведении тестирования необходимо моделировать реалистичные сценарии игрового процесса с большим количеством одновременных пользователей, чтобы определить, как приложение обрабатывает такую интенсивную нагрузку. Мобильные приложения обычно предлагают сложную графику и визуализацию, что может оказать дополнительное влияние на производительность приложения. Нагрузочные тесты должны учитывать нагрузку на графический процессор устройства и оценивать, как это влияет на общую производительность. Мобильные приложения могут включать онлайн-многопользовательский режим или другие сетевые функции, требующие постоянного взаимодействия с Дневник науки | [www.dnevniknauki.ru](http://www.dnevniknauki.ru) | СМИ Эл № ФС 77-68405 ISSN 2541-8327

серверами. Нагрузочные тесты должны проверять, как приложение работает в условиях нагрузки на сеть и как обрабатываются запросы к серверу [9]. Сценарии игрового процесса могут меняться в зависимости от действий игроков или других факторов. Нагрузочное тестирование должно учитывать такие динамические изменения и адаптироваться к ним для достоверного моделирования реальной нагрузки [3].

При проведении нагрузочного тестирования в мобильных приложениях следует учитывать также такие аспекты как: типы нагрузки, устройства и операционные системы, потребление ресурсов, обработка ошибок и аварийное завершение, тестирование сети, анализ производительности, мониторинг и анализ ресурсов.

Наиболее важной частью нагрузочного тестирования мобильных приложений является анализ влияния нагрузки на потребление ресурсов и производительность. В этом анализе выявляется то, как приложение реагирует на различные уровни нагрузки, как расходуются ресурсы устройства и как это влияет на производительность приложения. Влияние нагрузки на потребление ресурсов и производительность напрямую связано с потреблением памяти, использованием процессора, работой сети, задержки и времени отклика, масштабируемости, ответах на ошибки и восстановлению после сбоев. Проанализируем каждое из них подробнее.

Высокое потребление памяти приводит к замедлению работы приложения, а также к его аварийному завершению из-за нехватки ресурсов. Анализ использования центрального процессора (CPU) является критическим, потому что высокая загрузка CPU может вызывать замедление работы приложения и повышенное потребление энергии, что может привести к уменьшению времени автономной работы устройства [7]. Потребление ресурсов сети тоже играет важную роль. Высокая нагрузка на сеть может вызывать задержки в передаче данных или приводить к использованию большого объема данных, что может повлиять на пользовательский опыт и снизить производительность приложения.

Дневник науки | [www.dnevniknauki.ru](http://www.dnevniknauki.ru) | СМИ Эл № ФС 77-68405 ISSN 2541-8327

Анализ времени отклика приложения при разной нагрузке поможет определить, как быстро приложение реагирует на пользовательские действия или запросы. Долгие задержки могут ухудшить пользовательский опыт и привести к низкой оценке приложения. Масштабируемость означает способность приложения успешно обрабатывать растущее количество пользователей или данных без ухудшения производительности. Ошибки, не обнаруженные и не обработанные приложением, могут привести к аварийному завершению или неправильной работе. Важно оценивать, как приложение восстанавливается после сбоев или перегрузок.

Анализ влияния нагрузки на процедуру потребления ресурсов в процессе использования мобильных систем (игр, софта, приложений) способствует вычленению уязвимых проблем и оптимизирует приложение для сохранения бесперебойной работы даже при нагрузке выше среднего уровня. Это позволяет разработчикам предпринимать необходимые шаги для улучшения производительности, оптимизации ресурсов и повышения качества опыта пользователя продукта.

Для проведения нагрузочного тестирования мобильных приложений, как правило используют инструменты и программное обеспечение, которые способны довести процесс проведения тестов до автоматизированного состояния, моделировать реалистичную нагрузку и проводить качественный анализ результатов [5]. Рассмотрим самые эффективные из них.

На первом месте находится инструмент «JMeter» используемый как для нагрузочного тестирования в целом, так и для мобильных приложений в частности. Он позволяет создавать тестовые сценарии, моделировать различные виды нагрузки и предоставлять статистику по результатам. «JMeter» поддерживает тестирование HTTP/HTTPS, REST, SOAP, JDBC и других протоколов, которые применяются в мобильных системах. «Gatling» является инструментом для производительного тестирования с открытым исходным кодом. Он специально разработан для создания тестовых сценариев с большим Дневник науки | [www.dnevniknauki.ru](http://www.dnevniknauki.ru) | СМИ Эл № ФС 77-68405 ISSN 2541-8327

количеством одновременных пользователей. [7]. «Gatling» поддерживает язык «Scala» для описания сценариев, что делает его более мощным и гибким для сложных сценариев. «LoadRunner» - это инструмент от компании «Micro Focus», который позволяет проводить нагрузочное тестирование приложений с различными протоколами, включая HTTP, WebSockets, SOAP. Он предоставляет широкие возможности для создания разнообразных тестовых сценариев и анализа результатов. «Appium» - это автоматизированный фреймворк с открытым исходным кодом для тестирования мобильных приложений. Он позволяет проводить нагрузочное тестирование мобильных приложений, а также функциональное тестирование, с использованием различных языков программирования. «Calabash» - это еще один фреймворк с открытым исходным кодом для автоматизации тестирования мобильных приложений. Он позволяет проводить нагрузочное тестирование и функциональное тестирование на платформах Android и iOS [7]. Выбор инструментария как правило зависит от требований проекта, характеристик приложения и предпочтений команды разработки и тестирования.

Каждый из перечисленных инструментов имеет свои преимущества и подходит для определенных сценариев тестирования. Однако существует ряд инструментов, которые лучше не использовать для нагрузочной тестировки мобильных систем в связи с ограничениями, недостатками или несовместимостью. Например, «Postman» инструмент для тестирования и отладки API, который может быть полезен при разработке и тестировании веб-сервисов, но не является оптимальным для нагрузочного тестирования мобильных приложений. К общим чертам инструментов, которые не рекомендуется использовать для нагрузочного тестирования мобильных приложений относятся: их непригодность для создания сложных сценариев нагрузки, моделирования реалистичных условий использования и проведения оценки производительности в условиях средней и высокой нагрузки.

Для проведения нагрузочного тестирования лучше использовать только специализированные инструменты, перечисленные выше, так как они имеют спектр возможностей для мобильных приложений и систем.

В заключение, хотелось бы отметить что нагрузочное тестирование является неотъемлемой частью процесса создания мобильных систем и способствует обеспечению высокого качества, производительности и стабильности приложений, что важно для успешной работы на рынке мобильных сервисов.

### Библиографический список

1. Алефиренко В.М. Обзор и классификация информационных систем / В.М. Алефиренко, Н.О. Туровец // Danish Scientific Journal. - 2021. - Vol. 1. - № 55.- С. 52-56.
2. Величневич А.Г., Черепахин А.А., Кудряшов Н.И., Мельникова К.Б., Бычков Б.И. Обзор подходов и механизмов тестирования баз данных// Проблемы науки и образования. 2017. №14. С. 17-20.
3. Гончаренко В. А., Фомин С.И. Оценивание доступности информационных ресурсов компьютерных сетей в условиях дестабилизирующей рабочей нагрузки // Перспективы развития науки и образования: сб. науч. тр. по мат. международной НПК. 28 сентября 2012 г. Ч. 9. Тамбов: ООО «Консалтинговая компания Юком», 2012. С.42-43.
4. Ермакин А.А. Разработка метода построения комплекса нагрузочного тестирования распределенной информационной системы. СПб: Изд-во СПбГУ ИТМО, 2005. 147 с.
5. Криволятева М.С., Гайкова Л.В. Проблемы выборки тест-кейсов для автоматического тестирования ИТ-продуктов. Наука Красноярья. 2020. № 1. С. 83-94.
6. Лепёшкин С.А., Гончаренко В.А., Шульгин А.А. Моделирование и проектирование систем. Моделирование систем: учеб. пособие. СПб.: ВКА имени А. Ф. Можайского, 2016. №1. 244 с.

7. Рудюк Е.П., Соломатин А.А. Оценка производительности сервера базы данных путем применения нагрузочного тестирования // Наука, техника и образование: электрон, 2016. № 4 (9). С. 97-10.
8. Самочадин А.В., Сужаев О.И., Тимофеев Д.А., Рогов П.А. Инструментальные средства нагрузочного тестирования для систем централизованного управления мобильными устройствами// Информатика, телекоммуникации и управление, 2014. №6. С.33-42.
9. Туровец Н.О. Методы тестирования интегрированных информационных систем / Н.О. Туровец, В.М. Алефиренко // Science Time. - 2022. - № 3 (99). - С. 19-27.
10. Уткин Г.С., Башарин А. П. Особенности построения модели нагрузочного тестирования // Лесной вестник. 2009. №6. С.145-147.

*Оригинальность 82%*