

Кросикова Татьяна Ивановна

магистрант

Рашидова Елена Викторовна

доцент

Институт сферы обслуживания
и предпринимательства (филиал)

ФГБОУ ВО «Донской государственный
технический университет»

г. Шахты, Ростовская область

ПРОБЛЕМЫ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ И СПОСОБЫ ИХ РЕШЕНИЯ

***Аннотация:** сегодня все больше людей активно используют мобильные устройства: смартфоны и планшеты уверенно вытесняют настольные компьютеры. Число мобильных приложений сегодня достигает около полутора миллионов программ из App Store и одного миллиона шестисот тысяч на Google Play. Авторы данной статьи рассматривают ошибки мобильных приложений и способы их решений.*

***Ключевые слова:** мобильные приложения, разработка, проблема, решение, практичность.*

Избежать ошибок и проблем при разработке своего приложения и его интерфейса – непростая задача. В данной работе я хочу рассмотреть некоторые из этих проблем, и попытаюсь найти решения.

1. Проблема производительности приложения.

Чем популярнее ваше приложение, тем больше пользователей постоянно используют его, непрерывно посылая запросы и загружая свои данные. Конечно если юзеров десять или даже сто, проблема практически не заметна. Но если их количество переваливает за тысячи, приложение может «зависать», при этом запрошенная информация выдается не полностью или с опозданием.

Решение проблемы.

С этой проблемой можно справиться, повысив производительность. Например, можно преобразовать запросы мобильных пользователей в обычные запросы, получающие в ответ контент «на ходу». В компании Яндекс, в частности, с 2012 года поисковые приложения дают готовые ответы на запросы пользователей: они позволяют мгновенно получить нужную информацию, не загружая новую страницу. По статистике Яндекса, 15% запросов с мобильных телефонов связаны с ориентированием на местности, поиском изображений, новостей и прогноза погоды. Именно по этим тематикам чаще всего даются готовые ответы. На все остальные запросы приложения дают обычные ответы, которые тоже оптимизированы для восприятия с экрана мобильного.

Кроме того, некоторые данные можно кэшировать. Кэшировать нужно данные, которые медленно генерируются и часто запрашиваются. При наличии высоких нагрузок кеширование просто необходимо. Оно позволяет обслуживать больше клиентов, с теми же ресурсами, потому что поставщики данных больше отдыхают. Однако чрезмерное использование кэша может негативно повлиять на гибкость приложения.

2. Проблема хранения больших объектов в базе данных.

Разнообразие приложений растет, а с ним растет количество информации, хранимой на сервере. Аудио, Видео контент и другие тяжелые объекты при хранении в базе данных могут привести к перегрузке серверов, и, как следствие, потере производительности приложения. И чем больше пользователей у приложения, тем серьезнее проявляется эта проблема.

Решение проблемы.

Почти все системы баз данных, которые мы используем для хранения нашей информации, являются реляционными, такие как Oracle, SQL Server, MySQL, Sybase, DB2, TeraData и так далее. Хотя реляционные хранилища и обеспечивают наилучшую смесь простоты, устойчивости, гибкости, производительности их показатели по масштабируемости ниже, чем у аналогичных не реляционных си-

стем. Реляционные БД хорошо масштабируются только в том случае, если располагаются на единственном сервере. Когда ресурсы этого сервера закончатся, вам необходимо будет добавить больше машин и распределить нагрузку между ними. И вот тут сложность реляционных БД начинает играть против масштабируемости, а следовательно, и скорости работы приложений.

Для решения проблемы нужно применять другие типы баз данных, которые обладают более высокой способностью к масштабированию. Примером такого типа бд является хранилище типа ключ-значение (key-value store). Такие системы отлично подходят для облачных сервисов. Благодаря тому, что такие хранилища легко и динамически расширяются, они также пригодятся вендорам, которые предоставляют многопользовательскую веб-платформу хранения данных. Такая база представляет относительно дешевое средство хранения данных с большим потенциалом к масштабируемости.

Периодическая чистка серверов так же может помочь с решением проблемы хранения. Чистить можно те данные, которые никто не затребует в течение длительного времени.

3. Проблема DDoS.

Конкуренция в бизнесе ужесточается с каждым годом, а способы завоевания клиентов становятся всё более беспринципными. Если ваше мобильное приложение работает с интернетом, то есть возможность столкнуться с одним из видов такой борьбы – DDoS атакой.

Для того что бы понять, как приложение можно «поломать» DDoS-ом, необходимо понять, что же такое DDoS.

DDoS – атака, запущенная компьютером или сервером, которому дали цель и приказ к началу. После старта сеть зараженных машин начнет отправлять мертвые пакеты (бесполезную информацию) на роутер цели и таким образом блокирует соединение.

Помимо злоумышленников, которые могут атаковать специально, есть еще вариант того, что приложение «сломает» само себя.

В качестве примера можно взять случай, когда разработчик планирует ввести в работу новую функцию приложения и извещает об этом всех пользователей программы. Это может привести к огромной нагрузке на сервер, в результате чего он просто перестанет нормально функционировать.

Решение проблемы.

Во избежание поломки сервера по вине самого приложения, как в примере выше, необходимо разделить своих пользователей на группы и высылать уведомления последовательно каждой группе, а не всем сразу. Это позволит обеспечить всех пользователей обновлением, но нагрузка на сервер не будет чрезмерной.

Для защиты от сторонних атак существует много путей. Один из них – установить систему защиты от DDoS атак. Сейчас на рынке представлено очень много таких систем. Хотя этот способ весьма эффективен, он так же является весьма затратным. На данный момент менее чем 25% компаний внедрились решения для защиты от DDoS.

4. Проблема безопасность как вопрос первостепенной важности.

Многие компании и стартапы не предусматривают надежных мер безопасности для своих продуктов. В результате данные пользователей подвергаются опасности – их можно без особых проблем «увести», было бы желание.

Решение проблемы.

Сервера с БД и API необходимо держать в актуальном состоянии, регулярно обновляя отдельные компоненты ПО. Это не всегда простая задача, поэтому необходимо планировать технические мероприятия заранее.

1. Не нужно использовать пароли для доступа к удаленным серверам, вместо этого лучше использовать SSH.

2. Необходимо документировать запланированные и реализованные меры безопасности. Убедитесь в том, что меры, которые вы принимаете, действительно обеспечат безопасность данных и самого приложения.

3. Публичные, частные и защищенные данные храните отдельно, используя различные уровни доступа для каждой группы данных.

4. Хэшируйте и «солите» ваши пароли, шифруйте такие личные данные пользователей, как номера телефонов, или же не сохраняйте их, если это не является необходимостью.

5. Дайте пользователям возможность удалять свои данные из системы.

Список литературы

1. Аналитический портал веб-разработок [Электронный ресурс]. – Режим доступа: <http://www.cmsmagazine.ru/library/items/mobile/pain-mobile-developers> (дата обращения: 27.01.2016).

2. Проблемы «отзывчивого» дизайна и способы их решения [Электронный ресурс]. – Режим доступа: <https://vc.ru/p/responsive> (дата обращения: 28.01.2016).

3. Как ускорить android смартфон или планшет [Электронный ресурс]. – Режим доступа: <http://apptoday.ru/f-a-q/item/108> (дата обращения: 28.01.2016).

4. Мобильный поиск: моменты, которые имеют значение [Электронный ресурс]. – Режим доступа: <http://www.therunet.com/articles/734-mobilnyy-poisk-momenty-kotorye-imeyut-znachenie> (дата обращения: 28.01.2016).

5. Кэширование данных [Электронный ресурс]. – Режим доступа: <http://ruhighload.com/index.php/2009/06/23> (дата обращения: 28.01.2016).

6. Реляционные базы данных обречены? [Электронный ресурс]. – Режим доступа: <https://habrahabr.ru/post/103021/> (дата обращения: 28.01.2016).

7. DDOS сервера самого себя? Или что это? [Электронный ресурс]. – Режим доступа: <http://searchengines.guru/showthread> (дата обращения: 28.01.2016).

8. Обзор решения Juniper Networks по защите против DDOS-атак на уровне приложений [Электронный ресурс]. – Режим доступа: <https://habrahabr.ru/company/muk/blog/261657/> (дата обращения: 28.01.2016).

9. Ошибки приложений на Android [Электронный ресурс]. – Режим доступа: <http://devicehit.ru/publication/oshibki-prilozhenij-na-android/> (дата обращения: 28.01.2016).