

# РАЗРАБОТКА МОБИЛЬНОЙ СИСТЕМЫ МОНИТОРИНГА ТРАНСПОРТА

С.А. Грос<sup>1</sup>, Т.А. Онуфриева<sup>1</sup>

<sup>1</sup> Калужский филиал МГТУ им. Н.Э. Баумана, г. Калуга

В статье описывается мобильное приложение, предназначенное для мониторинга транспорта в городе с целью контроля бизнеса. Основной информацией для определения местоположения являются данные, получаемые с датчиков, расположенных на отслеживаемом транспорте. Полученные данные находятся на сервере. Данные с сервера передаются на устройство в формате JSON и выносятся на карту. Соединение с сервером и получение данных на мобильном устройстве осуществляется путем использования соответствующих классов из стандартных библиотек Android и Java.

Мобильное приложение реализует два основных этапа работы – сбор и обработка информации об объектах от датчиков, расположенных на отслеживаемых объектах и визуализация движения транспорта на основе этой информации.

Этап сбора информации с сервера предусматривает использование стандартных технологий обмена информацией между мобильным устройством и сервером. Как следствие, максимальная эффективность работы приложения может быть достигнута только в случае стабильного интернет соединения.

Этап визуализации движения для обычного пользователя является основным и позволяет достаточно легко отслеживать необходимые транспортные средства. Формализация этого этапа и адаптация его реализации в виде прогнозируемой анимации для обычного пользователя смартфона предполагается в последующих версиях мобильного приложения. Разработанное мобильное приложение позволяет пользователю эффективно отслеживать интересующие его события. Кроме того, в случае возникновения важных событий, это приложение позволяет получать мгновенные уведомления, что существенно повышает качество контроля бизнеса. В настоящее время осуществляется улучшение пользовательского интерфейса мобильного приложения, разработка алгоритма для реализации прогнозируемой анимации и алгоритма построения оптимального маршрута. Достижение поставленных целей планируется осуществить за счет использования соответствующих интерфейсных библиотек Android и математических методов регрессии и оптимизации. Приложение предназначено для работы на смартфонах Android, поскольку данная платформа имеет высокую популярность среди пользователей.

*Ключевые слова:* мобильное приложение, android, картографический сервис, транспортная логистика.

## ВВЕДЕНИЕ

Информационные технологии в настоящее время проникают во все сферы общественной деятельности. Современная логистика также не обходится без активного использования информационных технологий. На рынке систем мониторинга транспорта существует множество сервисов, но большинство из них предлагает решение в виде web-приложения. Применение мобильных приложений для транспортных и логистических компаний позволило бы повысить эффективность бизнеса, поскольку администраторы транспортных компаний смогли бы получать уведомления на свои мобильные устройства и своевременно реагировать на них.

В настоящее время популярные картографические порталы, такие как Google Maps, Яндекс.Карты и ряд других, предоставляют API (Application Programming Interface – интерфейс программирования приложений), которые позволяют строить и добавлять карты в свои приложения. API представляет собой набор средств, дающих возможность получить доступ к какому-либо сервису и запросить у него данные. Разработчик может воспользоваться им для получения доступа к функционалу программы, библиотеки, модуля [1].

При разработке мобильных приложений для отслеживания объектов необходимо использование карт, но многие картографические ресурсы являются платными, поэтому существует необходимость соотносить стоимость и функциональность картографического сервиса. Анализ сервисов ориентирован на платформу Android, так как это самая популярная мобильная платформа в мире. Согласно послед-

ним опросам, в мире свыше миллиарда активных Android-устройств, и их количество продолжает стремительно расти.

## АНАЛИЗ КАРТОГРАФИЧЕСКИХ СЕРВИСОВ

Картографический сервис – это специализированная информационная система, предоставляющая пространственные данные в виде интерактивной карты. Картографический сервис обеспечивает доступ к картографической информации на основе интерфейсов прикладного программирования. Рассмотрим наиболее популярные сервисы с точки зрения предоставляемых ими возможностей для создания собственных карт, их настройки, размещения на них необходимой информации и тарифные планы.

GoogleMaps – один из самых известных картографических сервисов. Существует с 2005 года. С помощью Maps SDK для Android можно добавлять карты на основе данных Google Maps в свое приложение. API автоматически обрабатывает доступ к серверам Google карт, загружает данные, отображает карты и позволяет обрабатывать жесты пользователя. Вы также можете использовать API для добавления маркеров, полигонов и слоев к основной карте, и изменить вид в определенной области карты. Подобные объекты предоставляют дополнительные сведения о местоположениях на карте и позволяют пользователю взаимодействовать с картой. API позволяет добавить следующие графические изображения на карту.

- Изображения, привязанные к определенным точкам на карте (Маркеры)
- Наборы линейных сегментов (Полилинии)
- Замкнутые сегменты (Полигоны)

- Растровые изображения, привязанные к определенным точкам на карте

- Наборы изображений, которые отображаются поверх основной карты

Для использования Google карт в своем приложении существует два тарифных плана: стандартная и премиум версии.

Стандартная версия позволяет бесплатно использовать карты, как в бесплатных, так и в платных приложениях для Android и iOS, если плата за использование взимается через GooglePlay или AppStore. Если плата взимается другими способами необходимо выбрать премиум версию. Также при использовании Google карт в приложениях для отслеживания людей и объектов необходимо приобретать премиум версию. Для уточнения стоимости использования премиум версии API необходимо связаться со службой поддержки Google.

MapKit – это программная библиотека, которая позволяет использовать картографические данные и технологии Яндекса в мобильных приложениях. MapKit позволяет создать приложение с картами Яндекса для операционных систем iOS и Android.

В таком приложении можно использовать поиск по организациям и топонимам, строить автомобильные и пешеходные маршруты с учётом актуальной дорожной ситуации, а также отображать информацию о пробках или панорамы. На созданной с помощью MapKit карте будут появляться строящиеся дома, новые дороги и развязки, недавно открытые организации. На ней будут доступны все те изменения, которые вносятся на Яндекс.Карты.

Картографический сервис MapKit возможно использовать как бесплатно, так и на коммерческой основе. Ниже перечислены бесплатные условия использования MapKit.

- Нельзя использовать MapKit в закрытых и платных приложениях

- Нельзя использовать MapKit для мониторинга и диспетчеризации

- Нельзя сохранять и изменять данные, полученные с помощью MapKit

- Нельзя превышать суточный лимит запросов (25000 запросов)

Коммерческая лицензия MapKit предназначена для приложений, которым не подходят условия использования бесплатной версии.

Лицензия ограничивает количество запросов в день согласно выбранному тарифу. При превышении лимита выставляется отдельный счет за превышение. В таблице 1 представлены тарифы компании Яндекс, для использования карт в коммерческих приложениях.

Табл. 1. Тарифные планы MapKit.

Максимальное количество запросов в сутки	Стоимость в год	Стоимость каждой тысячи запросов сверх лимита
До 1000 запросов	120 000 руб.	120 руб.
До 10 000 запросов	360 000 руб.	36 руб.
До 25 000 запросов	600 000 руб.	24 руб.
До 50 000 запросов	850 000 руб.	17 руб.
До 100 000 запросов	1 000 000 руб.	11 руб.

Число запросов считается по формуле 1.

$$q = sq + qr + 5 \times qp (1)$$

Где  $q$  – суточное число запросов,  $sq$  – число поисковых запросов,  $qr$  – число запросов на построение маршрута,  $qp$  – число запросов к панорамам. Количество запросов сверх лимита округляется до тысячи в большую сторону. Например, 400 запросов сверх лимита округляются до 1000.

Mapbox – компания, которая предоставляет геоинформационную платформу для разработчиков и дизайнеров. Mapbox Maps SDK для Android – это набор инструментов с открытым исходным кодом для отображения карт внутри приложения Android.

Maps SDK для Android позволяет полностью настраивать внешний вид карты. Вы можете выбрать один из основных стилей Mapbox или создать пользовательский стиль карты, настроив цвета, знаки и шрифты карты в соответствии с пользовательским интерфейсом приложения. API также предоставляет несколько различных способов добавления меток на карту, создания кругов, добавления линии между несколькими точками или рисования многоугольников. Часто эти объекты рисуются либо поверх карты, либо в некоторых случаях внутри самой карты. Библиотека также позволяет отслеживать местоположение устройства пользователя, использование офлайн карт, отслеживание жестов, введенных пользователем и многое другое.

Mapbox API является бесплатным в использовании до определенного количества запросов. В таблице 2 представлены тарифные планы компании Mapbox.

Табл. 2. Тарифные планы компании MapBox.

	Бесплатный лимит	При превышении лимита 0.5\$ за каждые
Количество активных пользователей в месяц	50 000	500
Количество запросов геоданных в месяц	50 000	1 000
Количество запросов построения маршрута в месяц	50 000	1 000

При использовании библиотеки в приложениях для отслеживания объектов необходимо связаться со службой поддержки, для уточнения условий использования.

Для решения поставленной задачи наиболее оптимальным является выбор картографического сервиса Mapbox, поскольку он имеет широкий функционал и удовлетворительную себестоимость. В приложении были использованы следующие средства разработки: язык программирования Java и AndroidStudio.

## СБОР И ОБРАБОТКА ИНФОРМАЦИИ

Разрабатываемое приложение является частью информационной системы, которая предполагает наличие у отслеживаемых объектов датчиков, сообщающих информацию о состоянии объекта [2-6]. Так как в разрабатываемом приложении на карту выносятся маркеры с координатами, соответствующими координатам транспорта в реальном времени, то необходимо определиться в каком формате следует получать данные о текущем местоположении объекта от сервера. В настоящее время в качестве формата дан-

ных в интернете широко используются две технологии: XML и JSON.

Для реализации приложения был выбран формат JSON, так как сообщения в данном формате легко подготавливать и обрабатывать. Формат является независимым от языка и может быть использован практически с любым языком программирования. Для множеств языков существуют готовые библиотеки, предназначенные для создания и обработки данных в формате JSON. JSON-текст может представлять собой (в закодированном виде) одну из двух структур:

- Набор пар ключ – значение. В разных языках это реализовано как: объект, запись, структура, словарь, хэш-таблица, список с ключом или ассоциативный массив. Ключом может быть только строка (регистрозависимая, имена с буквами в разных регистрах считаются разными), значением – любая форма.

- Упорядоченный набор значений. Во многих языках это реализовано как массив, вектор, список или последовательность.

Приведенные структуры данных являются универсальными: обычно, любой современный язык программирования их имеет поддержку в той или иной форме. Так как они используются для обмена данными между различными языками программирования, они легли в основу JSON. В AndroidSDK предусмотрены готовые классы для работы с JSON файлами.

Функционирование мобильного приложения можно разделить на два этапа: предварительный сбор информации и обработка полученной информации, рис. 1.

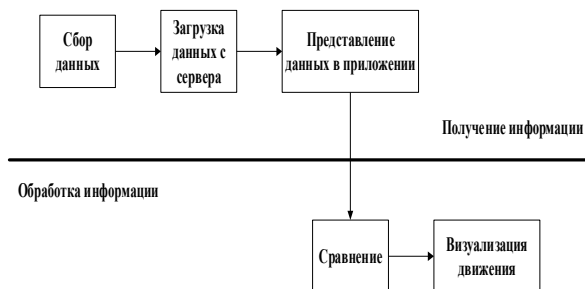


Рис. 1. Этапы работы приложения

Обмен данными предусматривает доступ к сети интернет на мобильном устройстве. На сегодняшний день такая возможность имеется на большинстве Android-устройств, а большое количество android-приложений взаимодействует со средой интернет. Для получения данных с определенного интернет ресурса в модели приложения была использована библиотека OkHttp [7-9].

Полученная в результате запроса строка содержит в себе объект с массивом устройств, которые необходимо отобразить на карте. Для извлечения информации из строки необходимо преобразовать ее к классу JSONObject из библиотеки Android и извлечь из данного объекта массив устройств путем создания экземпляра класса JSONArray и присвоением этому экземпляру результат выполнения метода JSONObject.getJSONArray().

Далее необходимо записать данные из полученного массива в специально созданные классы для работы с данными о транспортных средствах. Запись данных происходит в цикле, который проходит по заданному массиву. В конце каждой итерации цикла происходит запись в список машин, который в дальнейшем будет вынесен на карту.

Загрузка данных с сервера осуществляется с помощью отдельного класса AsyncTask. Данный класс предназначен для выполнения сложных задач и передачи результатов в UI-поток. Его использование обусловлено тем, что в случае выполнения тяжелой задачи в основном потоке приложение перестанет отвечать и аварийно завершится. AsyncTask имеет три метода: doInBackground, onPreExecute, onPostExecute.

Метод doInBackground выполняется в новом потоке, именно в этом методе происходит выполнение тяжелых задач и так как поток не основной, он не имеет доступа к UI. Метод onPreExecute выполняется перед doInBackground, имеет доступ к UI. Метод onPostExecute выполняется после doInBackground (не срабатывает в случае, если AsyncTask был отменен), имеет доступ к UI [10]. В данном проекте были использованы два метода doInBackground (загрузка данных с сервера) и onPostExecute (вынесение полученных объектов на карту).

В разрабатываемом приложении добавление маркеров на карту осуществляется путем использования класса SymbolManager из библиотеки Mapbox. Среди недостатков использования данного класса можно выявить то, что на данный момент он не поддерживает кластеризацию, что приводит к затруднению отображения большого количества маркеров на карте.

## ИНТЕРФЕЙС ПРИЛОЖЕНИЯ

Рассмотрим реализацию интерфейса разрабатываемого приложения. На рисунке 6 представлена главная активность приложения. Активностью называется оконная форма в Android-приложении. На главной активности приложения располагается карта с отслеживаемым транспортом.

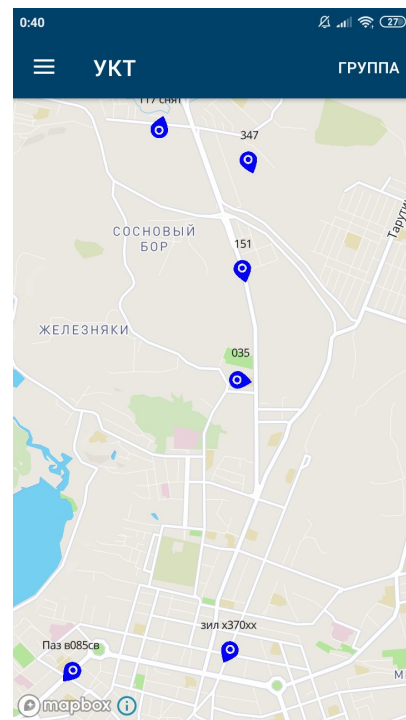


Рис. 2. Карта с объектами приложения

Предполагается, что отслеживание (мониторинг) осуществляется по нескольким группам пользователей и каждая

группа имеет свой набор транспорта, с соответствующими номерами объектов. Интерфейс меню приложения формируется в результате имеющихся групп у пользователя и представлен на рис. 7. Например, в случае, когда у пользователя имеется одна транспортная группа, в меню отсутствует выбор группы машин.

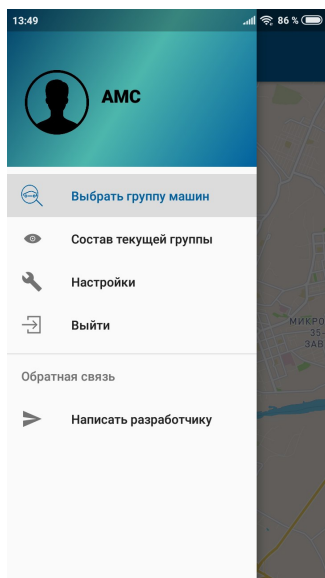


Рис. 3. Меню приложения

Рассмотрим каждый пункт меню. Выбор группы машин переключает пользователя на активность со списком доступных ему групп. Активность выбора транспортной группы представлена на рисунке 8.

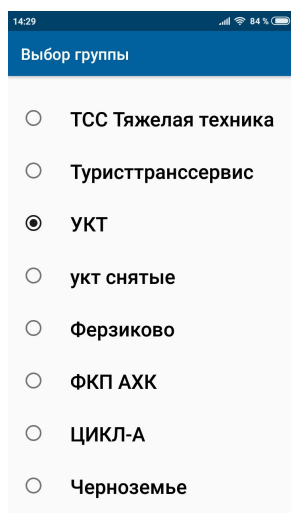


Рис. 4. Активность выбора группы машин

В результате изменения выбора транспортной группы на главной активности будут изменены отображаемые транспортные средства. При выборе пользователем пункта меню, отвечающего за состав текущей группы, пользователю будет представлена возможность ознакомиться со списком машин, соответствующих текущей группе. Активность, соответствующая данному пункту меню, представлена на рисунке 9.

078 ep	20 л.	
1 АТЭ х635хх40	108 л.	
197	141 л.	
200	286 л.	
222	278 л.	
320	477 л.	
325 с12.01.18	223 л.	
326	522 л.	
332	578 л.	
334	240 л.	
400му	388 л.	
420	259 л.	
425	209 л.	
428	274 л.	

Рис. 5. Активность текущей группы

При выборе транспортного средства из представленного списка, пользователь вернется на главную активность, на которой вместе с картой, в центре которой будет отображаться выбранное устройство с информационным фрагментом о состоянии автомобиля. Реализация информационного фрагмента представлена на рисунке 10.

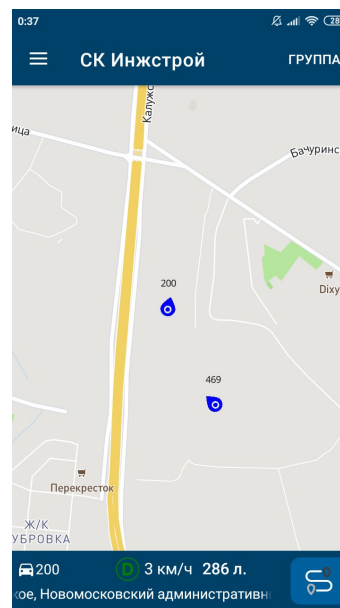


Рис. 6. Информационный фрагмент

Данный фрагмент содержит в себе наиболее важную для пользователя информацию. Также данную информацию можно получить при нажатии на маркер.

### ЗАКЛЮЧЕНИЕ

По результатам анализа картографических сервисов была выбрана библиотека Марбокс, приведена структура и описана модель мобильного приложения для мониторинга транспортных средств, использующая клиент-серверную архитектуру. Приведенная модель обеспечивает хорошую

производительность для решения задач мониторинга транспорта, не загружая процессор устройства. Среди недостатков приведенной модели можно выявить отсутствие прогнозируемой анимации движения транспорта.

Модернизация приложения предполагает реализацию прогнозируемой анимации, а также реализацию алгоритма для построения оптимального маршрута, что позволит представить пользователю удобный интерфейс и предлагать ему более экономные пути, с точки зрения времени и расхода топлива.

#### СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Операционная система Android учебное пособие / М.А. Дмитриев [и др.]. Москва: НИЯУ МИФИ, 2012. – 64 с.
2. Онуфриева Т.А., Щавелев Л.А., Подход к разработке системы мониторинга объектов, Электромагнитные волны и электронные системы. 2017. Т. 22. №3. С. 17-21.
3. Онуфриева Т.А., Зайцева А.А., Нахождение оптимальных маршрутов методом линейного программирования в АСУ ГУ, Электромагнитные волны и электронные системы. 2017. Т. 22. №3. С. 47-51.
4. Винокуров И.В., Мобильное приложение для распознавания фрагментов изображений, Электронный журнал: наука, техника и образование. 2018. №3(21). С. 35-42.
5. Зайцева А.А., Онуфриева Т.А., алгоритмы локального позиционирования объектов, Вестник Калужского университета. 2017. №2. С. 56-58.
6. Борсук Н.А., Подход к разработке интерфейсной части мобильного кроссплатформенного бизнес-приложения, Электромагнитные волны и электронные системы. 2018. Т. 23. №3. С. 32-38.
7. Соколова, В. В. Вычислительная техника и информационные технологии. Разработка мобильных приложений: учеб. пособие для прикладного бакалавриата / В. В. Соколова. – М. : Издательство Юрайт, 2019. – 175 с.
8. Семенчук, В. Мобильное приложение как инструмент бизнеса [Электронный ресурс] / В. Семенчук. – Электрон. дан. – Москва : Альпина Паблишер, 2017. – 240 с. – Режим доступа: <https://e.lanbook.com/book/101108>. – Загл. с экрана.
9. Бауэр, К. Java Persistence API и Hibernate [Электронный ресурс] / К. Бауэр, Г. Кинг, Г. Грегори ; под ред. Киселева А.Н. ; пер. с англ. Зинкевич Д.А.. – Электрон. дан. – Москва : ДМК Пресс, 2017. – 632 с. – Режим доступа: <https://e.lanbook.com/book/111435>. – Загл. с экрана.
10. Агеев, Е.Ю. Основы компьютерных сетевых технологий [Электронный ресурс] / Е.Ю. Агеев. – Электрон. дан. – Москва : ТУСУР, 2011. – 83 с. – Режим доступа: <https://e.lanbook.com/book/11484>. – Загл. с экрана.

*Грос Сергей Анатольевич – студент кафедры «Информационные системы и сети», Московский государственный технический университет им. Н.Э. Баумана (филиал), КФ МГТУ тел. +7(910)5468956, e-mail: sergey.gros@gmail.com.*

*Онуфриева Татьяна Александровна – к.т.н, доцент кафедры «Информационные системы и сети», Московский государственный технический университет им. Н.Э. Баумана (филиал), КФ МГТУ тел. +7(910)9109003, e-mail: onufrievata@mail.ru.*

## DEVELOPMENT OF A MOBILE TRANSPORT MONITORING SYSTEM

**S.A. Gros, T.A. Onufrieva**

*Moscow State Technical University, Kaluga*

The article notes the features of the structure of a mobile application for monitoring vehicles. Cartographic services are analyzed for use in the developed application. An example implementation is given, which shows the main stages of working with a mobile monitoring application.

Index terms: mobile application, android, map service, transport logistics.

*Gros Sergey Anatolievich – student, department of information systems and networks, Moscow State Technical University N.E. Bauman (branch), +7(910)5468956, e-mail: sergey.gros@gmail.com*

*Onufrieva Tatyana Aleksandrovna – Ph.D., associate professor of the department “information systems and networks”, Moscow State Technical University N.E. Bauman (branch), +7(910)9109003, e-mail: onufrievata@mail.ru*