

ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – СОФИЯ

**ФАКУЛТЕТ ПО КОМПЮТЪРНИ СИСТЕМИ И УПРАВЛЕНИЕ**

Дипломна работа

Тема:

**Система за позициониране и проследяване на превозни средства и мобилни устройства с GPS модул**

Разработил:

*Елизавета Величкова Каснакова*

*КСТ, ФКСУ*

Факултетен № *121212208*

Научен ръководител: *гл. ас. д-р А. Ташева*

София 2016

# Съдържание

[Съдържание 2](#_Toc453690114)

[Увод 3](#_Toc453690115)

[Глава I. Анализ, цели и задачи 5](#_Toc453690116)

[1. Анализ 5](#_Toc453690117)

[1.1. Prey Anti Theft 5](#_Toc453690118)

[1.2. GPS Tracker Car TK SMS 6](#_Toc453690119)

[1.3. Lost Android 7](#_Toc453690120)

[2. Цели 8](#_Toc453690121)

[3. Задачи 8](#_Toc453690122)

[Глава II. Обзор на използваните технологии 9](#_Toc453690123)

[1. ASP.NET Web API 9](#_Toc453690124)

[1.1. ASP.NET[11] 9](#_Toc453690125)

[1.2. Web API 10](#_Toc453690126)

[2. IIS (Internet Information Services)[13] 10](#_Toc453690127)

[3. MS SQL Server[14] 11](#_Toc453690128)

[4. Entity Framework[15] 11](#_Toc453690129)

[5. Android SDK 13](#_Toc453690130)

[6. GoogleMaps API[18] 13](#_Toc453690131)

[7. GCM 14](#_Toc453690132)

[Глава III. Проектиране на системата 16](#_Toc453690133)

[1. Цялостна идея 16](#_Toc453690134)

[2. Изготвяне на спецификации 16](#_Toc453690135)

[2.1. Функционална спецификация на уеб услугата: 16](#_Toc453690136)

[2.2. Функционална спецификация на мобилното приложение: 17](#_Toc453690137)

[2.3. Функционална спецификация на специалния софтуер за устройства с Андроид операционна система: 19](#_Toc453690138)

[3. Проектиране на цялостната архитектура 19](#_Toc453690139)

[4. Проектиране на базата данни 20](#_Toc453690140)

[5. Създаване на различни екрани 22](#_Toc453690141)

[5.1. Екрани на мобилното приложение 22](#_Toc453690142)

[5.2. Екрани на сцециалния софтуер 28](#_Toc453690143)

[6. Комуникация 29](#_Toc453690144)

[7. Създаване на план за тестване 29](#_Toc453690145)

[8. Описание на функционалностите 30](#_Toc453690146)

[Глава IV. Програмна реализация 32](#_Toc453690147)

[Глава V. Ръководство на потребителя 33](#_Toc453690148)

[Заключение 34](#_Toc453690149)

[Използвана литература 35](#_Toc453690150)

[Приложение 36](#_Toc453690151)

# Увод

Живеем в свят на високите технологии, заобиколени от скъпи коли, лаптопи и смартфони, без които вече даже не можем да си представим да оцелеем. Телефонът ни отдавна не е само телефон([телекомуникационен](https://bg.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BB%D0%B5%D0%BA%D0%BE%D0%BC%D1%83%D0%BD%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F) [апарат](https://bg.wikipedia.org/wiki/%D0%90%D0%BF%D0%B0%D1%80%D0%B0%D1%82), който предава и получава [звук](https://bg.wikipedia.org/wiki/%D0%97%D0%B2%D1%83%D0%BA), най-често човешки глас, на големи разстояния[1]), той обединява множество устройства – комуникационен апарат, калкулатор, таймер, хронометър, аларма, фенер, камера, GPS, MP3 плейър, телевизор, компютър и т.н. Използваме смартофа си от това да се обадим на родителите си, през това да чуем някоя любима песен, да покажем някое видео на приятел, да отговорим на някой служебен имейл или да пишем съобщение във Вайбър на съседа, до това да си платим сметките чрез онлайн банкиране. За добро или за лошо телефона е станал естествено продължение на ръката ни и не само той. Вече трудно можем да си представим живот без пералня, електрическа печка, хладилник, микровълнова и естествено кола. Като всички изброени в днешно време разполагат с множество допълнителни функции. Всичко това е отражение на технологическия прогрес и не може да се отрече, че благодарение на тези устройства спестяваме много усилия и време, а времето е най-ценният ресурс, с който разполагаме.

В същото време обаче ставаме и много зависими. Свикнали сме да излезнем навън, без даже да погледнем прогнозата за времето, защото не ни интересува особено какво е, тъй като се качваме в колата с климатик, свързваме си смартфона към нея, за да може да си поговорим с някого докато сме в задръстване, което обаче ни писва и решаваме да пробваме някой заобиколен път и за тази цел включваме навигацията или от телефона или вградената в модерното ни возило и накрая за немного време паркираме пред офиса. В момента, в който трябва да дадем колата в сервиза за няколко дена или пък си забравим телефона, изпадаме в паника. Как по друг начин освен с колата ще стигнем до офиса? Градски транспорт минава ли до там изобщо? Как да проверим дали минава, като телефонът с интернет не е в нас, а на всичкото отгоре и заваля дъжд, а ние вървим към спирката без чадър.

Тъй като тези устройства вече съдържат твърде много лична информация, те не са просто парче технология, което си върши работата и може да бъде заменено на момента с друго, което притежава същите функции. Хората са готови да вложат много пари, усилие, време и даже да се изложат на опастност, за да намерят изгубения си телефон или открадната кола, вместо да си купят същата[2][3]. И както тези технологии спестяват време и усилие на добросъвестните работещи хора, така и го спестяват на злонамерените крадци. Начините, по които се крадат телефони едва ли са се променили за последните години, но стойността на откраднатите телефони определено се е. Имайки предвид цените на смартфоните, крадците със сигурност са на добра печалба. За колите ситуацията не е много по-различна. От една страна може да кажем, че с новите технологии сигурността се е повишила – аларми, проследяващи устройства, иновативни системи за заключване и т.н., но от друга точно с нови технологии вече може да се открадне кола от разстояние – отваря се шофьорската врата и след това се запалва двигателя, като и едното и другото стават от дистанция[4]. По данни на МВР само за миналата година тези престъпления са скочили с 7%[5]. Също така статистически данни от лондонската полиция сочат, че седемдесет на сто от всички лични престъпление е кражбата на мобилен телефон, както и Службата за национална статистика (ONS) съобщава, че това е единственото престъпление, в Обединеното кралство, което е всъщност във възход[6].

Затова би било чудесно да можем да проследяваме любимите ни устройства: да сме информиране по всяко време за позицията им и да бъдем уведомявани навреме, ако колата ни например изведнъж започне да се движи, при положение, че ключовете са в нас и сме я оставили на паркинга в мола. По този начин можем да реагираме бързо при необходимост. Това не би било полезно само при кражба. Понякога се случва просто да се разсеем, да си оставим телефона някъде и после да не можем да го намерим. Вкъщи ли да го търсим или пък го оставихме в таксито? Също така можем да проследяваме и хора, посредством телефоните им. Дали децата наистина са отишли на курса по английски или на концерта, на който им забранихме да ходят? Или служителите на фирмата ни дали наистина имат поръчка за морето, накъдето са се запътили със служебната кола? Една обща система за проследяване на всичко, в което има вграден или би могъл да се постави GPS модул, може да отговори на всички тези въпроси и да осигури спокойна вечеря, без постоянно надничане през прозореца за проверка дали колата е още там.

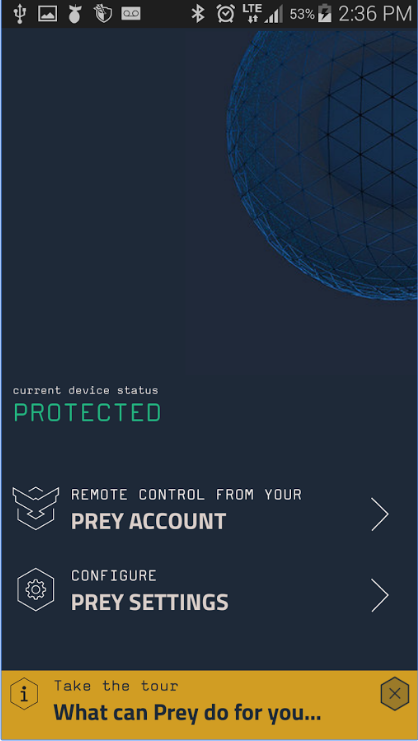
# Глава I. Анализ, цели и задачи

Създаването на система за позициониране и проследяване на превозни средства и мобилни устройства с GPS модул изисква внимателно проектиране и поставяне на крайни цели за постигане, чрез изпълнението на определени задачи.

## Анализ

В днешно време всичко, което бихме могли да измислим вече е измислено отдавна и направено. Чрез бързо проучване в интернет могат да се открият приложения, които изпълняват същите или сходни функции, като разработваната система. Всеизвестно е, че най-разпространената платформа за мобилни устройства е Андроид. В неговия онлайн магазин за приложения – Google Play, се откриват няколко подобни и това са *Prey* , *GPS Tracker Car TK SMS Free* и *Lost Android*.

Фиг. К2 my open courses

* 1. Prey Anti Theft*[7]* – Prey помага да получите своя изгубен или откраднат телефон, таблет и лаптоп. Prey е 100% безплатен и може да защити до 3 устройства с един единствен акаунт. Prey работи на всички известни операционни системи, така че тя ви позволява да следите и да предизвикате действия на всичките си устройства от едно място. Подобно на универсално дистанционно управление, но и за лаптопи и мобилни телефони.

**Функционалности на Prey:**

- Намиране на телефон на картата чрез геолокация, използвайки и GPS и WiFi триангулация.  
- Създаване на доклади със снимки взети с мобилното устройство, скрийншотове и местоположение (ключовата част от данните, които са нужни на полицейските служители, за да предприемат действия).  
- Заключване на устройството.  
- Пускане на силна аларма дистанционно, дори ако телефонът е в режим на без звук.  
- Показване на приспособено предупредително съобщение на екрана.  
- Събиране на информация от мрежата, към която устройството е свързано (за точно позициониране).

Фигура . Prey - скрийншот на приложението

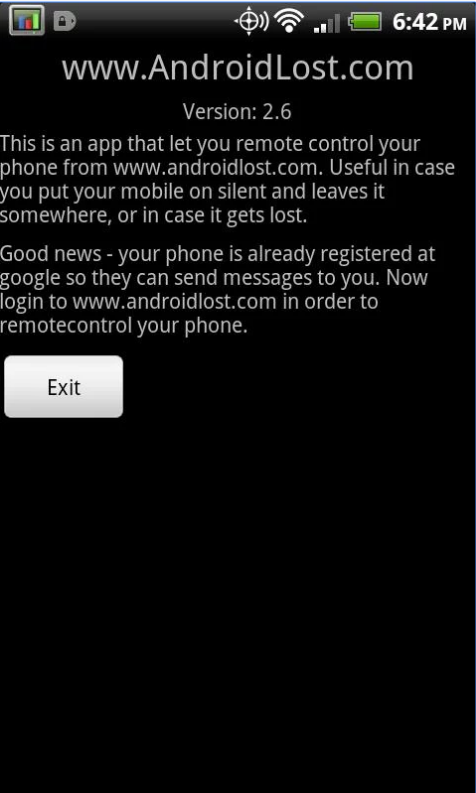
* 1. GPS Tracker Car TK SMS*[8]*  – Austral GPS Tracker помага да се следят и настройват GPS тракери от огромен брой производители. Може да се следят превозни средства (коли, лодки, мотоциклети и т.н.), хора и животни, проследявани от устройствата Xexun, Zclelec, Coban и други, от Андроид смартфон с няколко кликвания на цената на SMS съобщения.

В приложението има четири основни екрана:  
- Dashboard(Табло): където може да се провери текущото местоположение на автомобила, скоростта и разстоянието от смартфона (Андроид). Всичко това в прост и интуитивен екран с карта.  
- Locations(Местоположения): кога и къде автомобила e ходил откакто приложението е инсталирано. Също така показва реалния адрес на това място, не просто географска ширина и дължина.  
- Events (Събития): разглеждане на точните SMS команди, изпратени и получени от устройството в лесно четима форма, която всеки може да разбере.  
- Settings(Настройки): тук може да се изпращат някои команди на устройството като смяна на паролата, промяна на режима на GPS тракера, GeoFence, намаляване на двигателя на превозното средство, определяне на превишена скорост, сигнали за движение и още много.

Фигура . GPS Tracker Car TK SMS - скрийншот на приложението

Може също така да има светлинни и звукови предупреждения, показващи, че превозното средство е преместено или ако алармата е задействана, SOS е натиснат или превозното средство е превишило скоростта, която е предварително дефинирана.

* 1. Lost Android[9] – Дистанционно управление на Андроид телефон от http://www.androidlost.com или чрез SMS.

Функционалности:

- Чете изпратени и получени SMS съобщения

- Изтриване на всичко в телефона

- Заключване на телефона

- Изтриване на SD карта

- Локализира чрез GPS или интернет мрежа

- Стартиране на аларма с мигащ екран

- Изпращане на SMS от уеб страница

- Изскачащи прозорци със съобщения

- Прехвърляне на повикванията

- Статус на телефона: батерия, IMEI, и т.н.

- Старт / стоп на GPS

- Старт / стоп на WIFI

- Пращане на имейл, когато SIM картата е променена

- Снимане с предна или задна камера

- Предизвикване на телефона да говори чрез текст-към-говор

Фигура . Android Lost - скрийншот на приложението

- Изпращане на команди чрез SMS съобщения

- Таймаут при заключване

- Възстановяване на настройките при зареждане

- Записване на звук от микрофона

## Цели

* 1. Да се създаде система за позициониране и проследяване на превозни средства и мобилни устройства с GPS модул като мобилно приложение за Андроид.
  2. Приложението да чете данните от база данни, посредством публична REST услуга.

## Задачи

* 1. Създаване на спецификация за мобилното приложение;
  2. Архитектурно проектиране на цялата система;
  3. Избор на технологиите, с които може да се постигне крайната цел с възможно най-малко трудности и проблеми.
  4. Проектиране и създаване на базата данни;
  5. Проектиране на приложението без връзка с услугата или базата данни;
  6. Създаване на услугата, която да оперира с базата данни;
  7. Свръзване на мобилното приложение с базата данни посредством услугата.
  8. Тестване на мобилното приложение и оправяне възможни проблеми.
  9. Създаване на ръководство за работа.

# Глава II. Обзор на използваните технологии

В разработката на тази дипломна работа са използвани следните технологии: ASP.NET Web API, IIS, MS SQL Server, Entity Framework, Android SDK, Google Maps API, GCM.

## ASP.NET Web API

ASP.NET Web API е уеб рамка(framework), която прави лесно изграждането на HTTP услуги, които достигат до широк кръг от клиенти, включително браузъри и мобилни устройства. ASP.NET Web API е идеална платформа за изграждане на RESTful приложения върху рамката на .NET[10].

### ASP.NET[11]

ASP.NET е следващото поколение уеб рамка, разработена от [Майкрософт](https://bg.wikipedia.org/wiki/%D0%9C%D0%B0%D0%B9%D0%BA%D1%80%D0%BE%D1%81%D0%BE%D1%84%D1%82). За първи път е публикуван през януари 2002 година с версия 1.0 на [.NET Framework](https://bg.wikipedia.org/wiki/.NET_Framework), и е наследник на Microsoft Active Server Pages ([ASP](https://bg.wikipedia.org/wiki/ASP)) технология, но да не се бърка, че не е подобрена версия на ASP. ASP.NET е изградена въз основа на Common Language Runtime ([CLR](https://bg.wikipedia.org/w/index.php?title=CLR&action=edit&redlink=1)), което позволява на програмистите да пишат ASP.NET код като използват .NET език по избор. За първи път е публикуван през януари 2002 година с версия 1.0 на. NET Framework, и е най-успешния наследник на Microsoft Active Server Pages (ASP) технологията.

ASP.NET цели производителност спрямо останалите скрипт-базирани технологии (включително класическия ASP) като компилира сървърно кода в един или повече DLL файлове на Уеб сървъра. Тази компилация става автоматично когато страницата бива заредена за пръв път (което от своя страна означава, че програмистът не трябва да изпълнява отделни компилации за страниците). Тази характеристика осигурява лекота на разработване, предлагана от скриптовите езици, с производителността на бинарните операции. Трябва да се има предвид обаче, че самата компилация може да причини забележимо забавяне при потребителя когато редактираната страница бива изискана за пръв път от Уеб сървъра, но това забавяне не би се появило отново преди следваща промяна.

ASPX и необходимите ресурси се поставят на виртуален хост на [Internet Information Services](https://bg.wikipedia.org/wiki/Internet_Information_Services) сървър (или друг сървър съвместим с ASP.NET). Когато потребителят за пръв път поиска да зареди страница, [.NET Framework](https://bg.wikipedia.org/wiki/.NET_Framework) парсва и компилира файловете в .Net асембли и изпраща отговор; при последвалите зареждания се извикват от DLL файловете. ASP.NET има възможността да компилира цели сайтове на части от по 1000 файла при първо зареждане. Ако забавянето е значително или причинява проблеми, големината на самите части може да бъде променяна.

Програмистите могат също да изберат да компилират предварително своите файлове, преди поставянето им на сървъра, използвайки MS Visual Studio, елиминирайки нуждата от първоначалното компилиране в Уеб средата. Това елиминира и нуждата от качване на изходен код на Уеб сървъра. Той поддържа и предварително компилиран текст.

* 1. Web API[12]

Уеб API(Application Programming Interface) е интерфейс за приложно програмиране или за уеб сървър или за уеб браузър. От страна на сървъра, уеб API е програмен интерфейс, състоящ се от един или повече публично изложени крайни точки на система за дефинирана заявка-отговор система за съобщения, обикновено представени като JSON или XML, който е изложен през интернет мрежата, най-често с помощта на HTTP-базиран уеб сървър. Крайните точки са важни аспекти на взаимодействие със сървърни уеб APIs, тъй като уточняват къде се намират ресурсите, които могат да бъдат достъпени чрез софтуер на трети страни. Обикновено достъпът е чрез URI, към който HTTP заявките са поставени, и от които по този начин се очаква отговор.

## IIS (Internet Information Services)[13]

Това е пакет от сървърни услуги, включващ уеб сървър, пощенски сървър и сървър за споделяне на файлове. Създаден от [Майкрософт](https://bg.wikipedia.org/wiki/%D0%9C%D0%B0%D0%B9%D0%BA%D1%80%D0%BE%D1%81%D0%BE%D1%84%D1%82), той е конкурента на Apache при Unix системите. Той поддържа следните протоколи HTTP, HTTPS, FTP, FTPS, SMTP и NNTP. IIS най-често се използва за съхраняване на динамични сайтове написани на ASP.NET (Web Forms и MVC), също така се използва и за съхранение на различни уеб услуги (WCF Services и ASP.NET Web API). Като основен недостатък се изтъква пряката свързаност с Windows като операционна система, а като основно предимство се изтъква много добрата и надеждна сигурност.

## MS SQL Server[14]

Това е система за управление на релационни бази данни на компанията Майкрософт. SQL Server представлява мощно средство и среда за създаване и функциониране на автоматизирани информационни системи. Системата притежава средства за описание и управление на данните, за дефиниране на връзки между различните обекти, за извличане на информация и представянето й по подходящ за потребителя начин, както и програмен език за реализиране на специфични алгоритми за обработка на данните. Тя е снабдена с множество средства като програми-помощници, генератори на обекти, автоматизирани операции и др., които улесняват значително потребителите при използването на системата.

Основният начин за извличане на данни от базата на SQL Server е чрез заявки към него. Заявката се изразява с помощта на вариант на SQL (Structured Query Language), наречен T-SQL. Той излага ключови думи за операциите, които могат да бъдат извършени на SQL Server, включително създаване и промяна на схеми на бази данни, въвеждане и редактиране на данни, както и мониторинг и управление на самия сървър. Клиентските приложения, които консумират данни или управляват сървъра ще използват функционалността на SQL Server чрез изпращане на T-SQL заявки и команди, които след това се обработват от сървъра и резултатите (или грешките) се връщат на клиентското приложение. Заявката декларативно уточнява какво да бъде извлечено от базата данни. Тя се обработва от процесор за обработка на заявки, който пресмята, последователността от стъпки, които ще бъдат необходими за извличане на исканите данни. Последователността от действия, необходими за изпълнение на заявката, се нарича план на заявката. Може да има няколко начина да се обработи една и съща заявка. Например, за заявка, която съдържа JOIN и SELECT, изпълнението на JOIN между двете таблици и след това изпълнението на SELECT на резултатите ще даде същия резултат като SELECT от всяка таблица и след това изпълнение на JOIN, но плановете за изпълнение ще бъдат различни. В такъв случай, SQL Server избира плана, чрез който ще постигне очакваните резултатите за възможно най-кратко време. Това се нарича оптимизация на заявки и се извършва от самия процесор за обработка на заявки.

## Entity Framework[15]

Това е ORM (Object-Relational Mapping) Framework към .NET. За първи път Entity Framework 1.0 е бил включен в .NET Framework 3.5 SP 1. Версията е силно критикувана и дори официално е отхвърлена от общноста на разработчиците. Втората версия наречена Entity Framework 4.0 е включена като част от .NET Framework 4.0 и при нея всички недостаци на първата версия са отстранени. От версия 6 Entity Framework е отделена от .NET Framework.

Entity Framework е набор от технологии в ADO.NET, които подкрепят развитието на ориентирани към данните софтуерни приложения. Архитекти и разработчици на такива приложения обикновено са се борили с необходимостта от постигане на две много различни цели. Те трябва да моделират обектите, взаимоотношенията и логиката на бизнес проблемите, които решават, и също трябва да работят с бази данни, използвани за съхранение и извличане на данни. Данните могат да обхващат няколко системи за съхранение, всяка със своите протоколи; дори приложения, които работят с една единствена система за съхранение, трябва да балансират между изискванията на системата за съхранение и изискванията за писане на ефективен и лесен за поддържане код.

Entity Framework дава възможност на разработчиците да работят с данни под формата на домейн-специфични обекти и свойства, като клиенти и адреси на клиенти, без да се налага да се занимават с основните таблици и колони на базата, където се съхраняват данните. С Entity Framework разработчиците могат да работят на по-високо ниво на абстракция, когато се справят с данни, и могат да създават и поддържат ориентирани към данни приложения с по-малко код, отколкото в традиционните приложения.

Основните типове данни в Entity Framework се наричат Entity Types, а конкретните им репрезентации Entities. Всеки Entity Type има определени характеристики (Properties), които описват съответните данни. Характеристиките могат да бъдат 2 вида – прост тип и комплексен тип.

Свързания модел се създава благодарение на асоциации. Всеки Entity Type се асоцира със друг Entity Type на база на съответни критерии (Foreign Keys). Entity Framework също така поддържа и съответни реакции при промяна (Update и Delete) като се поддържат каскадни и стандартни реакции (Actions).

## Android SDK

Андроид(Android) е мобилна операционна система (OS), в момента разработвана от Гугъл(Google), базирана на Линукс ядрото и предназначена предимно за сензорни мобилни устройства като смартфони и таблети. От 2013 г. е най-продаваната операционна система за таблети и смартфони. За развитието на Андроид се грижат голям брой софтуерни разработчици, които създават така наречените „apps“ (Applictions) – малки [приложения](https://bg.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD_%D1%81%D0%BE%D1%84%D1%82%D1%83%D0%B5%D1%80), които разширяват функционалността на системата. Приложенията могат да бъдат сваляни от различни сайтове в Интернет или от големи онлайн магазини като [Android Market](https://bg.wikipedia.org/wiki/Android_Market) (впоследствие преименуван на Google Play) – магазинът на Гугъл.[16]

Приложенията обикновено се разработват на езика за програмиране Джава (Java) с помощта на Андроид комплекта за разработване на софтуер (SDK – Software Development Kit), но други среди за разработка също са на разположение. Комплектът за Андроид разработване на софтуер, включва цялостен набор от инструменти за разработка. Те включват дебъгер, библиотеки, емулатор на базата на QEMU, документация, примерен код и уроци. В момента поддържаните платформи за развитие включват компютри, работещи под Линукс (всяка модерна десктоп Линукс дистрибуция), Mac OS X 10.5.8 или по-нова, и Windows XP или по-нов. До около края на 2014 г., официалната интегрирана среда за разработка (IDE - Integrated Development Environment) беше Eclipse с помощта на плъгин за Андроид ADT(Android Development Tools - инструменти за разработка), макар IntelliJ IDEA IDE (всички издания) напълно да поддържа разработването на Андроид приложения, и NetBeans IDE също поддържа разработката за Андроид чрез плъгин. От 2015 г. Android Studio, направено от Гугъл и задвижвано от IntelliJ, е официалната среда за разработка. Въпреки това, разработчиците са свободни да използват други. [17]

## GoogleMaps API[18]

Google Maps е картографска услуга на Гугъл, с която могат да се разглеждат географски карти в произволен уеббраузър. Картите са с променлив мащаб и показват широк спектър от информация за разглеждания регион или местност. Има допълнителен режим за разглеждане на сателитни изображения с или без картографско съдържание, както и теренни карти с изолинии на височините над морското равнище. За територията на България се извършва автоматичен превод на български на населените места и улиците.

Google Maps API позволява да се вграждат Гугъл Карти в собствени уебстраници, десктоп или мобилни приложения. Приложно-програмният интерфейс на Гугъл Карти и Гугъл Земя предоставя определен брой функции за работа с тези карти (така както е в http://maps.google.com и http://code.google.com/apis/earth/), позволяващ да се добави съдържание през различни услуги, и така да се създават пълноценни карти на произволен сайт.

## GCM

Google Cloud Messaging (често наричан GCM) е мобилна услуга, разработена от Гугъл, която позволява на разработчиците на приложения на трети страни да изпращат данни за уведомяване или информация от управлявани сървъри към приложения, насочени към операционната система на Гугъл - Андроид, както и приложения или разширения, разработени за интернет браузъра на Гугъл - Хром. GCM е достъпен за разработчиците безплатно. Той има възможността за изпращане на пуш известия(push notifications), дълбоко-обвързващи команди(deep-linking commands) и данни на приложението. По-големи съобщения могат да се изпращат с до 4 KB полезен товар на данни.

При разрешение на приложението да получава и показва известия, клиентското приложение изпраща заявка за регистрация към GCM интерфейса, за да започне процеса на регистрация. GCM услугата приема и потвърждава заявката и отговаря като дава на устройството за GCM Регистрационно ID, уникален идентификатор, който разработчикът по-късно използва за изпращане на известия до индивидуалното устройство. Идентификаторът се съхранява в устройството и обикновено се изпраща към приложния сървър на разработчика за съхранение. Този идентификатор не съдържа никаква лична или информация за устройството, която може да позволи на разработчиците да открият самоличността на потребителя. Когато разработчикът иска да изпрати известие към дадено устройство, процеса започва с POST заявка към API-то на GCM Authentication услугата. POST заявката включва GCM Регистрационното ID, приоритет, незадължителни стойности и връзки, както и информацията, която трябва да се показва на устройството при пристигането й. При успешна проверка на съответния идентификатор и други идентификационни данни, се връща автентикационен тоукен (authenication token). И двата идентификатори след това се изпращат към GCM услугата да бъдат поставени на опашка и доставени на устройството.

# Глава III. Проектиране на системата

## Цялостна идея

Цялостната идея на системата е да предоставя удобен, бърз и лесен начин за управление и следене на всичко ценно, което притежаваме и би могло да бъде откраднато или изгубено. В тази категория спадат мобилни телефони, таблети, лаптопи, коли, колела, но не сме ограничени само до това. На практика може да се следи всичко, на което може да бъде поставен или вече има вграден GPS модул и евентулано аларма като се инсталира специален софтуер, който да се грижи за комуникацията с уеб услугата. Има съществуваши решения, разгледани по-горе, които предоставят много повече функционалност от тази разработка, но представляват решения на отделни проблеми, докато тази разработка обединява всичко в едно с по-ограничени за момента функционалности, но с присъствието на фундаметалните такива. Също така реализацията на разработката, която е чрез създаването на публична уеб услуга, дава възможност за създаването на множество различни клиенти, през които системата може да бъде достъпена. На този етап в тази дипломна разработка, ще се създаде клиентско приложение за най-широко използваната мобилна операционна система Андроид. То ще бъде обединение между клиент на системата, през който може да се проследяват и управляват желани мобилни устройства/превозни средства, и специален софтуер, който да изпраща координатите на устройството, на което е инсталирано, ако някой го следи.

## Изготвяне на спецификации

Първата и основна стъпка от проектирането на системата е изготвянето на функционална спецификация. В нея се описват всички функции, които системата трябва да изпълнява.

* 1. **Функционална спецификация на уеб услугата**:
     1. Да предоставя услуга за регистрация на потребители.
     2. Да предоставя услуга за влизане в профила на даден потребител с предоставени от него валидни потребителско име и парола.
     3. Да предоставя услуга за излизане от даден потребителски профил.
     4. Да предоставя услуга за регистриране(запазване на уникалния идентификатор в базата данни) на устройство/превозно средство от специалния софтуер, инсталиран на него.
     5. Да предоставя услуга на регистрираните потребители за добавяне, валидиране за съществуване и запазване в базата данни на устройство/превозно средство за проследяване.
     6. Да предоставя услуга на регистрираните потребители за връщане на списък с всички добавени от тях устройства/превозни средства.
     7. Да предоставя услуга на регистрираните потребители за премахване на устройство/превозно средство от списъка им.
     8. Да предоставя услуга на регистрираните потребители за връщане на текущите координати на едно или повече устройства/превозни средства от списъка им.
     9. Да предоставя услуга на регистрираните потребители за връщане на множество координати в хронологичен във времето ред за определ ден и определено устройство/превозно средство от списъка.
     10. Да предоставя услуга на регистрираните потребители за отбелязване, че дадено устройство/превозно средство от списъка не трябва да се движи за определен интервал от време, да го проследява и изпраща известие към клиентско приложение, ако даденото устройство/превозно средство започне да се движи преди да е изтекъл зададения интернал от време.
     11. Да предоставя услуга на регистрираните потребители за постоянно приемане на координати от дадено устройство/превозно средство от списъка.
     12. Да предоставя услуга на регистрираните потребители за приемане на искане за изпращане на сигнал към дадено устройство/превозно средство да си включи звука и пусне алармата.
     13. Да предоставя услуга на регистрирани потребители за включване/изключване на изпращането на координати от дадено устройство/превозно средство от списъка.

### **Функционална спецификация на мобилното приложение**:

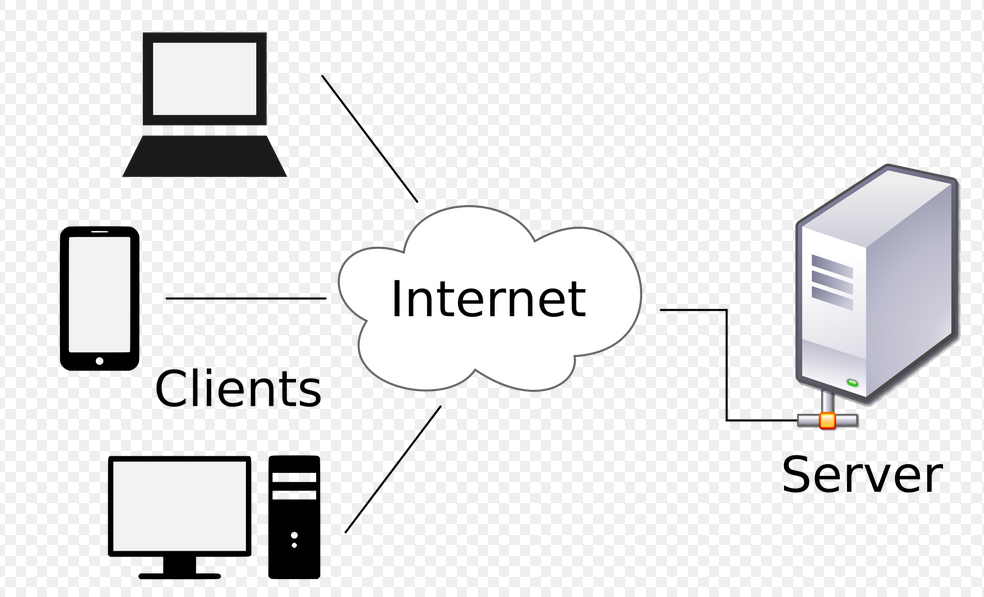
* + 1. Да позволява на потребителите да се регистрират чрез прост и интуитивен потребителски интерфейс.
    2. Да позволява на потребителите да влизат в профила си, предоставяйки потребителско име и парола чрез прост и интуитивен потребителски интерфейс.
    3. Да позволява на потребителите да излизат от профила си чрез прост и интуитивен потребителски интерфейс.
    4. Да позволява на регистрираните потребители да добавят ново устройство/превозно средство за проследяване, предоставяйки необходимата информация за него чрез прост и интуитивен потребителски интерфейс.
    5. Да позволява на регистрираните потребители да виждат списък с всички въведени от тях устройства/превозни средства за проследяване.
    6. Да позволява на регистрираните потребители да премахват устройства/превозни средства от списъка.
    7. Да позволява на регистрираните потребители да виждат текущите позиции на желани от тях устройства/превозни средства от списъка им върху лесна за ориентиране географска карта.
    8. Да позволява на регистрираните потребители да виждат история на посетени места от дадено устройство/превозно средство за избран от тях ден върху лесна за ориентиране географска карта.
    9. Да позволява на регистрираните потребители да отбелязват, че дадено устройство/превозно средство не трябва да се движи за определен от тях интервал от време чрез прост и интуитивен потребителски интерфейс.
    10. Да приема сигнал от услугата когато дадено устройство/превозно средство се движи, а е отбелязано, че не трябва и да създава известие.
    11. Да позволява на регистрираните потребители да поискат включване на звук/аларма на дадено устройство/превозно средство от списъка.
    12. Да позволява на регистрираните потребители да включват/изключват изпращането на координати от дадено устройство/превозно средство от списъка чрез прост и интуитивен потребителски интерфейс.

### **Функционална спецификация на специалния софтуер за устройства с Андроид операционна система**:

* + 1. Да предоставя на потребителите информация за уникалния индентификатор на тяхното мобилно устройство, който им е нужен да въведат, ако искат да следят това устройство.
    2. Да приема сигнал от услугата за включване/изключване на постоянното изпращане на координати от това мобилно устройство.
    3. Да включва GPS на телефона, ако е изключен, и да изпраща координатите на мобилното устройство на заден план без да пречи на нормалната работа на устройството и потребителя.
    4. Да приема сигнал от услугата за включване на звука на съответното мобилно устройство и издаване на звук и да извършва съответните действия.

## Проектиране на цялостната архитектура

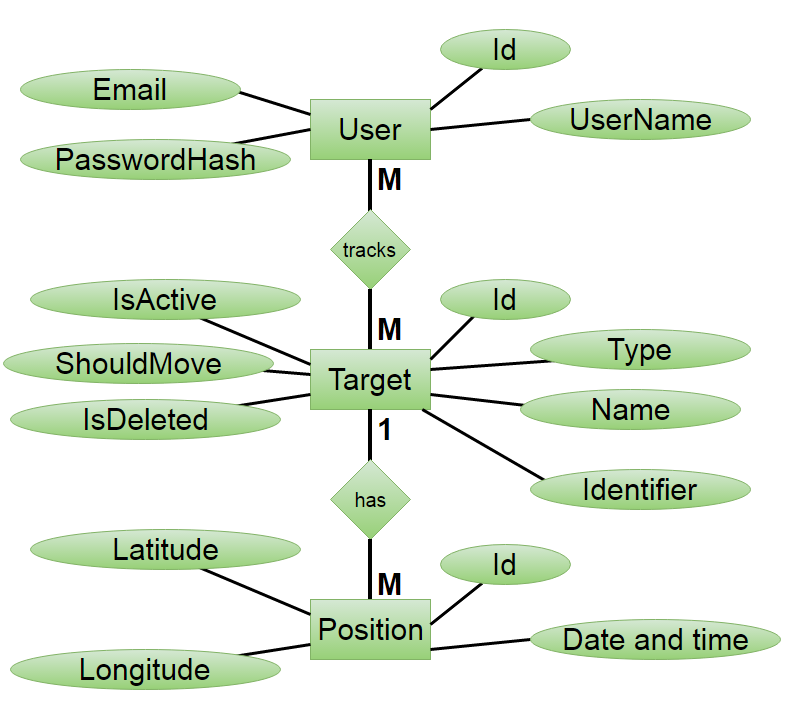
За постигане на горепосочените цели и идеи тази разработка ще използва клиент-сървър архитектура. Сървърът предлага набор от услуги, достъпни през интернет чрез HTTP заявки и осъществява връзката с базата данни. Клиентите предоставят потребителски интерфейс и използват уеб услугите като пращат HTTP заявки и визуализират върнатите данни. По този начин се постига независимост от използваната платформа или език за програмиране на клиента и сървъра. Всичко, което е нужно за комуникация между тях е интернет връзка.



Фигура 4. Клиент-сървър архитектура

## Проектиране на базата данни

В базата данни ще се пази информация за потребителите, устройствата/превозните средства, които те искат да проследяват и съответно история на местоположенията на следените устройства/превозни средства. Приемаме, че един потребител може да проследява множество устройства/превозни средства, както и едно устройство/превозно средство може да бъде проследявано от множество потребители(например всички в семейството се интересуват от това къде се намира семейната кола във всеки един момент, тъй като всички я карат). Описаните обекти и взаимовръзки между тяй може да се видят на следната ER(Entity-Relationship) диаграма:



Фигура 5. ER диаграма на базата данни

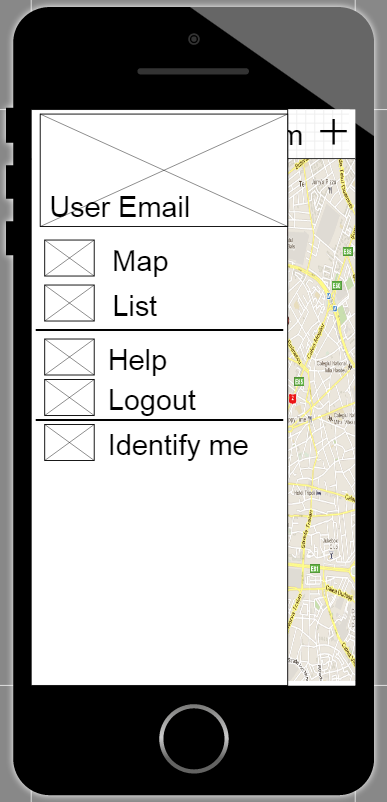
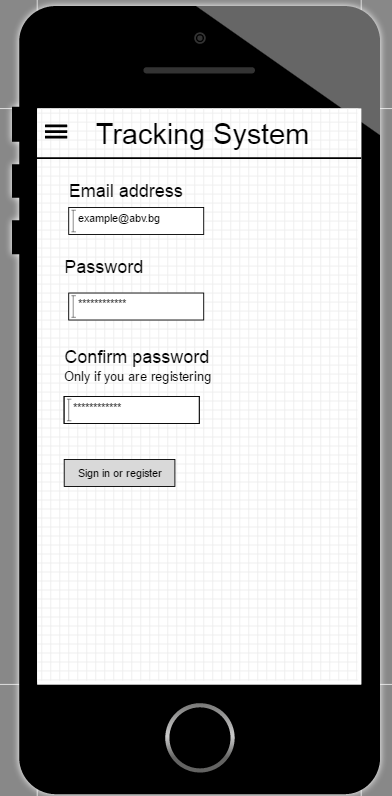
**Фиг. 7 Цялостна архитектура на системата**

## Създаване на различни екрани

Създаване на тъй наречените различни екрани (wireframes) и навигацията помежду им. Те представляват скелета на потребителския интерфейс.

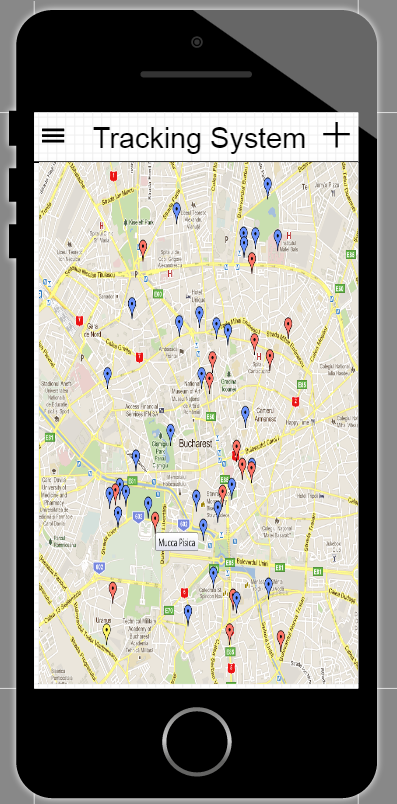
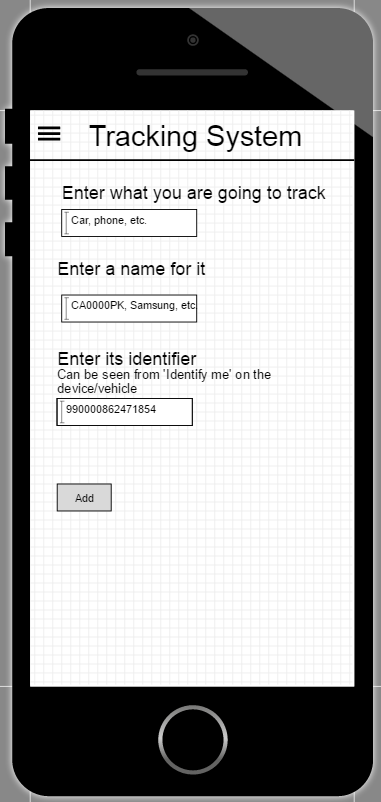
### Екрани на мобилното приложение

Първо ще започнем с екрана за регистрация и влизане в профила на даден потребител. Регистрацията и логина са обединени в един екран като има полета за въвеждане на електронна поща, парола и ако потребителя се регистрира – потвърждение на паролата(Фигура 6). Най-отгоре на всички екрани има навигационна лента, по средата на която има изписан различен текст в зависимост на кой екран се намира потребителя. В лявата страна на лентата винаги има бутон за меню, при чието натискане се появява менюто от Фигура 7, което е само за потребители влезнали в профила си. За потребители, които не са влезнали в профила си менюто ще съдържа само опциите Login, Help и Identify me. Като Help ще представлява обикновен екран с пояснителен текст за това как се използва приложението.

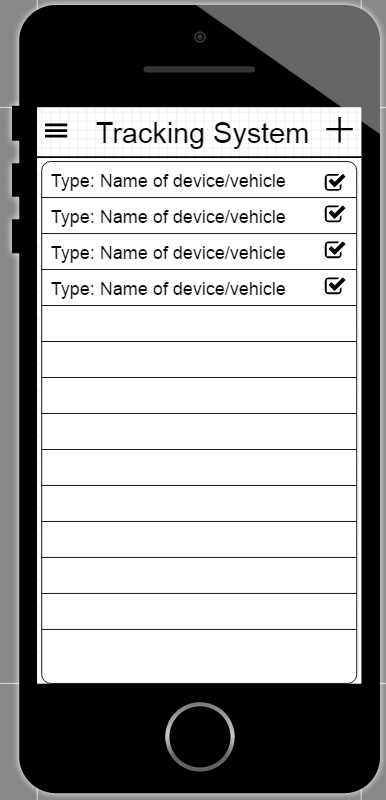
Фигура 7. Екран с менюто

Фигура 6. Екран за регистрация и логин

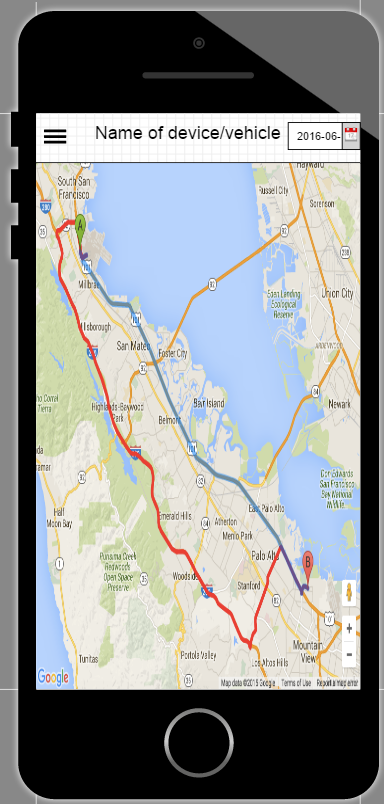
След като потребителя се регистрира и влезе в профила си ще се появява началния екран, който представлява географска карта на Гугъл, на която са отбелязани местоположенията на устройствата/превозните средства, които иска да проследява(Фигура 8). На този екран от дясната страна на навигационната лента има бутон за добавяне на нови устройства/превозни средсства за следене. При натискането му ще се появява нов екран, където трябва да се попълнят тип на устройството/превозното средство(например кола, телефон, таблет и т.н.), име(например регистрационен номер или марка за кола, телефонен номер или марка на телефона и т. н.) и уникален идентификатор, който се взима от специалния софтуер на устройството/превозното средство, което потребителя иска да добави за проследяване(Фигура 9).

Фигура 8. Начален екран с карта

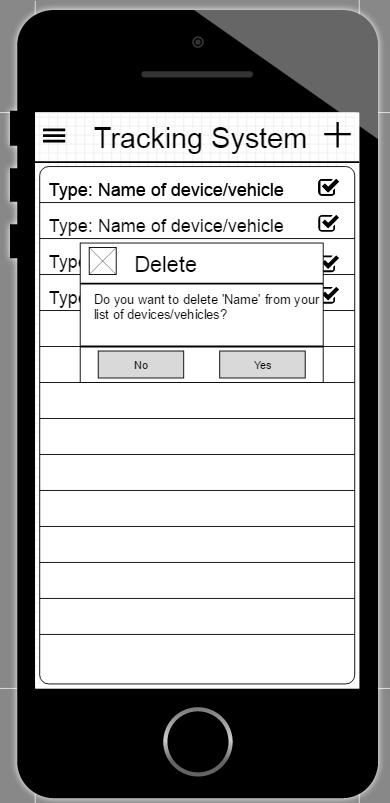
Фигура 9. Екран за добавяне на устройство/превозно средство за проследяване

Когато потребителят плъзне с пръст по екрана от дясно наляво или натисне опцията List от менюто, началния екран с картата ще се смени със списък на всички устройства/превозни средства, които потребителят е добавил за следене. Показва се техния тип, име, зададени от потребитеря при добавянето им, както и чекбокс, индикиращ дали потребителя иска да получава информация за местоположението на това устройство/превозно средство или не(Фигура 10). Ако е маркирано, значи местоположението на съответното устройство/превозно средство ще се отбелязва на картата и то постоянно ще праща координатите си на услугата през определен период от време. Ако не е маркирано, координати не се изпращат и устройстовото/превозното средство не се отбелязва на картата. При натискане на някое устройстовото/превозното средство от списъка се зарежда нов екран с географска карта на Гугъл, на който е отбелязан пътят изминат от съответното устройстовото/превозното средство за текущия ден. От дясната страна на навигационната лента бутона за добавяне се сменя с бутон за смяна на датата. При натискането му потребителя има възможност да избере за кой ден да се отбележи изминятия път на съответното устройстовото/превозното средство(Фигура 11). Ако потребителя е на екрана със списъка то може да се върне към началния екран с картата по два начина: плъзгане с пръст от ляво надясно или чрез опцията Map от менюто.

Фигура 10. Екран със списък на проследявани устройства/превозни средства

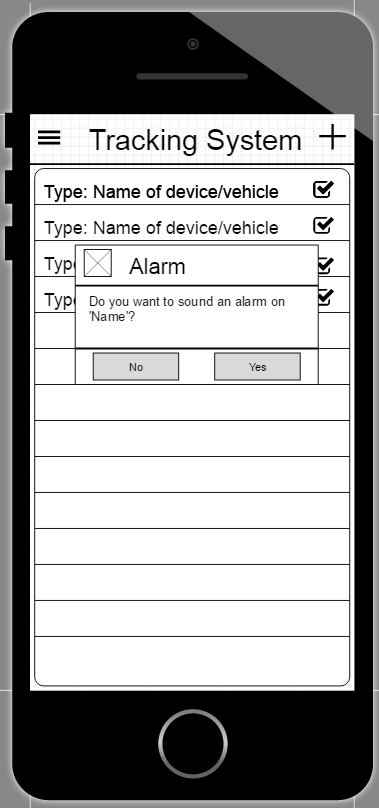
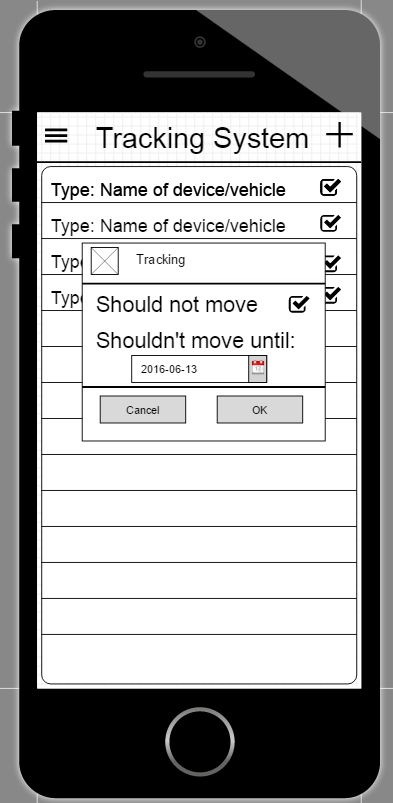
Всеки елемент от списъка разпознава три вида жестове(gestures), при всеки от които се случва определено действие:

Фигура 11. Екран за проследяване пътя на дадено устройство/превозно средство

* Плъзгане от дясно наляво – този жест ще изпълнява триене на елемент от списъка. Ще изскача диалогов прозорец, търсещ потвърждение от потребителя за триене на избраното устройство/превозно средство(Фигура 12).
* Двойно тапване – този жест ще представлява искане от потребителя за пускане на звук/аларма на избраното устройство/превозно средство. Отново ще изскача диалогов прозорец, искащ потвърждение от потребителя(Фигура 13)

Фигура 12. Диалогов прозорец за триене на устройство/превозно средство

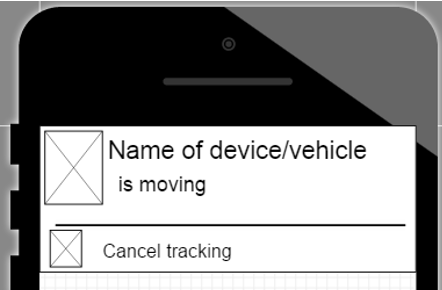
* Продължително задържане - този жест ще означава, че потребителя иска да уточни дали съответното устройство/превозно средство трябва или не трябва да се движи в определен интервал от време. Ще изскача диялогов прозорец, на който може да се отбележи дали даденото устройство/превозно средство трябва да се движи. Ако не трябва, потребителя отбелязва до кога устройството/превозното средство трябва да се следи за движение(Фигура 14).



Фигура 13. Диалогов прозорец за пускане на аларма на избрано устройство/превозно средство

Фигура . Диалогов прозорец за уточняване дали избраното устройство/превозно средство не трябва да се движи за определено време

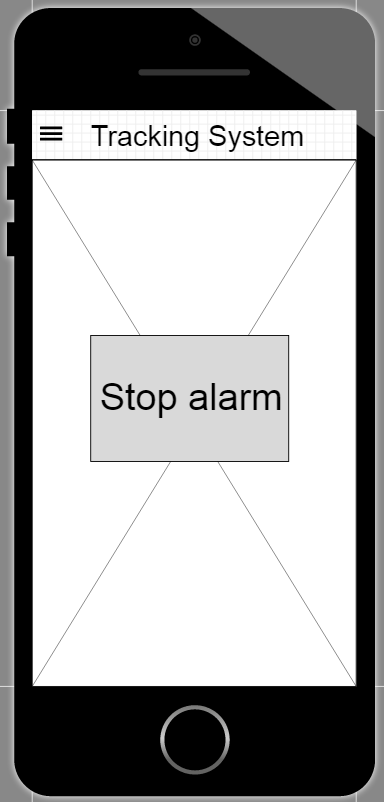
В случай, че потребителя е задал за дадено устройство/превозно средство определен интервал от време, в което не се очаква устройството/превозно средството да се движи, но услугата засече движение, то тя ще изпрати сигнал към мобилното приложение и ще бъде генерирано известие за потребителя(Фигура 15).



Фигура 15. Известие за нередно движение на устройство/превозно средство

### Екрани на сцециалния софтуер

Фигура 16. Диалогов прозорез, покозващ уникалния идентификатор на устройството

Както е споменато по-нагоре, специалния софтуер за Андроид и мобилното приложение, позволяващо проследяване на устройства/превозни средства, са обединени в едно приложение. Затова от менюто на главното приложение всеки потребител, регистриран или не, може да види опцията, която идва от специалния софтуер – Identify me. При избиране на тази опция изскача диалогов прозорец с уникалния идентификатор на съответното устройство, който е нужен за идентифициране това устройство и трябва да бъде въведен от потребителя при добавянето му към списъка със следени устройства/превозни средства, описан по-горе(Фигура 16). За мобилни устройства с Андроид операционна система, за уникален идентификатор се взима IMEI(International Mobile Station Equipment Identity) на устройството.

Фигура 1. Екран за спиране на алармата

Още един екран, идващ от специалния софтуер, е видим за всички, регестрирали се или не, потребители, представляващ празен екран с един бутон за спиране на аларма(Фигура 17). Той се появява, когато някой потребител следящ съответното устройство, е поискал пускане на алармата му, по описания по-горе начин чрез главното приложение. Този екран позволява прекратяване на алармата, тъй кано по подразбиране тя няма да спре преди потребителя изрично да укаже, че иска това.

## Комуникация

За правилно функционариане на системата е нужна интернет връзка на сървъра, където се намира уеб услугата, на мобилното устройство с главното приложение и на всички устройства/превозни средства, които се следят. Комуникацията помежду им се осъществява чрез пращане на HTTP заявки и получаване на оттовори. Всички клиенти на системата, в случая мобилното приложение, и проследявани устройства/превозни средства, чрез специалния им софтуер, могат директно да комуникират с уеб услугата. Комуникацията между две устройства(например при изпращане на сигнал за включване на алармата на проследявано устройство от проследявощото го устройство) не е директна, а става посредством уеб услугата. Тя приема заявката за сигнал от проследяващото устройство и праща сигнал на проследяваното за извършване на съответното действие.

## Създаване на план за тестване

Планът за тестване ще покрие основните функционалности и най-вече по-крайните и специфични случаи и въвеждането на невалидни данни от потребителя. Това ще помогне и за добавянето на задължителни проверки по време на разработването на системата. Ще бъде обърнато особено внимание на следните сценарии:

* Държание на системата при липса на интернет – трябва да се изведе съобщение на потребителя за възникналия проблем и да не се позволяват каквито и да е действия от страна на потребителя.
* Държание на системата при възникване на неочквана грешка – да се съобразят всички възможни грешки, да бъдат прихванати и потребителя да бъде осведомен по подобаващ начин.
* Тестване на правилната навигация от менюто и от екран към екран. Да не се позволява връщане назад към екран, който потребителя не е автентикиран да вижда.
* Тестване на автоматичното програмно включване на GPS и звук – включват ли се наистина, ако потребителя изрично ги е изключил.
* Тестване на изпращането на координати от устройството след рестартирането му – трябва GPS-а да се включва автоматично след рестартиране на устройстовото и да продължат да се пращат координати, ако това се изисква.
* Тестване на обновяването на екрани – картата обновява ли се постоянно, за да показва най-новите местоположения на следените устройства/превозни средства, както и списъка с устройства/превозни средства обновява ли се при успешно добавяне/премахване на елемент.
* Тестване за правилно прихващане на жестовете
* Проверки за автентикация на потребителя и непозволяване на извършване на действия, предназначени само за автентикирани потребители
* Проверки за празни стойности навсякъде, където потребителя може да въвежда текст.
* Проверки при регистрация и извеждане на подходящи съобщения, когато:
  + въведеният имейл адрес не е валиден имейл адрес;
  + вече съществува такъв потребител;
  + паролата не отговаря на минималните изисквания за парола – поне 6 символа;
  + паролата и потвърждението на паролата не съвпадат;
* Проверка, при добавяне на ново устройство/превозно средство за проследяване, дали такова наистина съществува и ако не – извеждане на подходящо съобщение.
* Проверки при въвеждане на дати от потребителя – да не се позволява въвеждане на невалидни дати.

## Описание на функционалностите

Потребителя има възможност да:

* се регистрира;
* влиза и излиза в профила си;
* добява и премахва устройства/превозни средства за проследяване;
* разглежда списък със следените от него устройства/превозни средства;
* следи за местополежението на добавените от него устройства/превозни средства;
* указва на кои добавени от него устройства/превозни средства иска да проследява местоположението;
* разглежда история на пътя изминат от дадено устройство/превозно средство за определен от него ден;
* указва дали дадено устройство/превозно средство не би трябвало да се движи за определен от него интервал от време и да получава известие, ако се засече движение;
* изисква пускането на аларма на определено устройство/превозно средство;
* спира алармата на устройството си, ако е било поискано тя да бъде включена;
* види уникалния идентификатор на мобилното му устройство, чрез който би могъл да го проследява чрез системата;

# Глава IV. Програмна реализация

# Глава V. Ръководство на потребителя

# Заключение

# Използвана литература

1. <https://bg.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BB%D0%B5%D1%84%D0%BE%D0%BD>
2. [http://www.businessinsider.com/smartphone-theft-statistics-2014-5](http://www.businessinsider.com/smartphone-theft-statistics-2014-5%20)
3. <http://www.bbc.com/news/world-europe-35371093>
4. <http://www.banker.bg/finansov-dnevnik/read/kradat-skupite-koli-ot-distanciia>
5. <http://novanews.novatv.bg/news/view/2016/02/05/138603/%D1%81%D0%B5%D1%80%D0%B8%D0%BE%D0%B7%D0%B5%D0%BD-%D1%80%D1%8A%D1%81%D1%82-%D0%B2-%D0%BA%D1%80%D0%B0%D0%B6%D0%B1%D0%B8%D1%82%D0%B5-%D0%BD%D0%B0-%D0%BA%D0%BE%D0%BB%D0%B8/>
6. <http://www.yac.mx/bg/mobile-security/android/how-to-stop-the-rise-of-smartphone-theft.html>
7. <https://play.google.com/store/apps/details?id=com.prey&hl=bg>
8. <https://play.google.com/store/apps/details?id=com.austral.gpscartracker>
9. <https://play.google.com/store/apps/details?id=com.androidlost>
10. <http://www.asp.net/web-api>
11. <https://bg.wikipedia.org/wiki/ASP.NET>
12. <https://en.wikipedia.org/wiki/Web_API>
13. <https://en.wikipedia.org/wiki/Internet_Information_Services>
14. <https://en.wikipedia.org/wiki/Microsoft_SQL_Server>
15. <https://en.wikipedia.org/wiki/Entity_Framework>
16. <https://bg.wikipedia.org/wiki/Android>
17. [https://en.wikipedia.org/wiki/Android\_software\_development#Android\_SDK](https://en.wikipedia.org/wiki/Android_software_development%23Android_SDK)
18. <https://bg.wikipedia.org/wiki/Google_%D0%9A%D0%B0%D1%80%D1%82%D0%B8>
19. <https://en.wikipedia.org/wiki/Google_Cloud_Messaging>

# Приложение