

## СОЗДАНИЕ СИСТЕМЫ СЛЕЖЕНИЯ С ИСПОЛЬЗОВАНИЕМ СИСТЕМЫ НАВИГАЦИИ GPS

А. И. БОЛТУНОВ, А. В. ЦЫКАРЕВ

Пермский государственный национальный исследовательский университет, 614990, Пермь, Букирева, 15

На сегодняшний день в мире существует несколько навигационных систем, использующих искусственные спутники Земли, но предлагающими действительно глобальный сервис позиционирования практически в любом месте нашей планеты являются лишь две: российская ГЛОНАСС и американская NAVSTAR. Именно к ним принято относить популярное сокращение GPS (Global Positioning System).

Сегмент потребителей GPS, аналогично сегменту потребителей ГЛОНАСС, состоит из приемников и некоторых дополнительных устройств, таких как антенны, интерфейс с исполнительными устройствами, а также вспомогательного программного обеспечения. В простейшем случае приемник получает от НКА навигационные данные, встроенный вычислитель решает навигационную задачу и выводит на дисплей абсолютные значения координат. Однако для большинства применений столь скромных возможностей недостаточно.

На первом этапе работы мы создаем приложение для мобильных устройств на базе ОС Android. Данное приложение будет «прослушивать» изменение координат устройства и при перемещении на расстояние большее, чем заранее определенная минимальная дистанция, будет формировать сообщение в виде:

```
<ID устройства>;<широта (десятичный формат)>;<долгота (десятичный формат)>
```

Это сообщение будет передаваться как SMS-сообщение на сервер для дальнейшей обработки.

Разработка приложения велась в среде программирования Eclipse и с помощью комплекса средств разработки Android SDK на языке программирования Java.

Приложение содержит класс Autorun, производный от базового класса BroadcastReceiver. По сути, этот класс реализует приемник широковещательных сообщений. Настроим этот приемник, добавив обработчик события BOOT\_COMPLETED. В этом обработчике необходимо вызвать сервис, выполняющий работу с GPS модулем и записать в лог-файл сообщение о запуске сервиса. После запуска устройства и загрузки ОС, создан-

ный нами приемник перехватывает сообщение о событии `BOOT_COMPLETED` и вызывает обработчик.

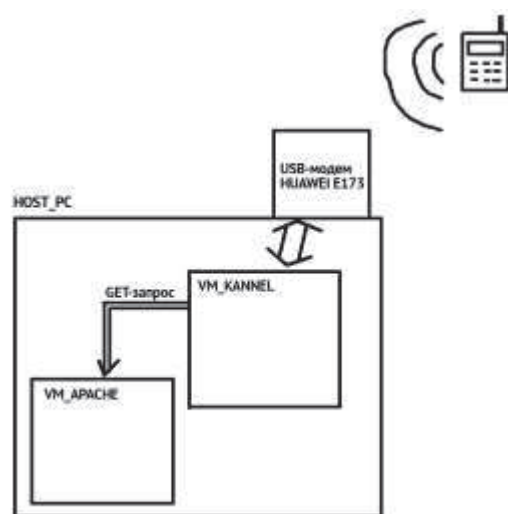
Теперь рассмотрим непосредственно работу с GPS. После запуска сервиса, инициализации потока и получения настроек, приложение запрашивает доступ к системной локационной службе. Получив его, приложение начинает запрашивать информацию о модуле с помощью функции `requestLocationUpdates`. В качестве одного из аргументов указывается минимальная расстояние, при перемещении на которое передается информация о местоположении.

При изменении местоположения вызывается обработчик события `OnLocationChanged`. В этом методе реализованы следующие действия:

- получение уникального идентификатора устройства;
- получение координат, характеризующих местоположение устройства. Используются два метода класса `Location`: `getLatitude()` для получения широты в градусах и `getLongitude()` – для получения долготы. Эти методы извлекают информацию из RMC-сообщений, полученный устройством от GPS модуля по протоколу NMEA-0183;

- формируется сообщение вида:  
<ID устройства>;<широта>;<долгота>;
- приложение получает доступ к функциям телефона, и затем происходит отправка SMS сообщения на сервер.

Следующим этапом работы является построение системы, способной принимать и обрабатывать СМС-сообщения, переданные телефоном-передатчиком. Для построения такой системы нам понадобится обычный ПК и USB-модем. Принципиальная схема взаимодействия передатчика и всех элементов системы приёмника приведена на рис. 1.



**Рис.1** Схема взаимодействия передатчика и всех элементов системы приёмника

На ПК созданы две виртуальные машины: VM\_KANNEL и VM\_APACHE. Виртуальной машине VM\_KANNEL предоставлен доступ к управлению USB-модемом. При поступлении на модем нового СМС-сообщения производится GET-запрос на машину VM\_APACHE. На машине VM\_APACHE развернут сервер APACHE – при поступлении GET-запроса данные обрабатываются и заносятся в базу данных.

В данном случае на машине VM\_KANNEL установлено ПО, которое играет роль СМС-шлюза. В качестве СМС-шлюза используется KANNEL - простой в использовании и мощный СМС-шлюз с открытым исходным кодом. KANNEL необходимо настроить на работу с нашим USB-девайсом в качестве модема. Так же необходимо указать на какой адрес будет посылаться GET-запрос при поступлении СМС. Сам запрос будет иметь следующий вид:

`http://192.168.1.6/incoming.php?check=321321&mobile=%p&text=%a"`

Где:

- «192.168.1.6» - ip-адрес виртуальной машины VM\_APACHE;
- «%p» - параметр хранящий в себе номер телефона отправителя;
- «%a» - текст СМС сообщения.

После получения GET-запроса виртуальная машина VM\_APACHE должна сохранять полученные данные с целью их последующего использования. Для хранения полученных от VM\_KANNEL было решено использовать базу данных. В нашем случае, база данных состоит из двух таблиц: «users» и «locations».

В таблице «users» хранится информация о пользователях системы. Структура таблицы «users» приведена в табл. 1.

**Таблица 1. Информация о пользователях системы**

Поле	Пояснения
id	Уникальный идентификатор пользователя
login	Логин, используемый для авторизации на сайте
IMEI	IMEI телефона, к которому привязан данный пользователь
phone	Номер телефона пользователя
password	Пароль пользователя(используется для авторизации на сайте)

В таблице «locations» хранится информация о перемещении пользователей. Структура таблицы «locations» приведена в табл. 2.

**Таблица 2. Информация о перемещении пользователей**

Поле	Пояснения
id	Уникальный идентификатор положения
date	Дата в формате UTS
lat	широта
lon	долгота
user_id	Уникальный идентификатор пользователя, которому принадлежит данное положения

Такая структура базы данных позволяет одновременно хранить информацию о перемещениях разных пользователей системы, а также позволяет исключить возможность доступа пользователя к чужим данным.

В дальнейшем полученные данные могут быть использованы для визуализации перемещений.

В перспективе, разработанная в ходе данной работы система может быть усовершенствована. Возможные применения такой системы:

- система слежения за автомобилями таксопарка;
- сервис обнаружения потерянного/украденного телефона.