

vychislitel'naya-tekhnika/library/ispolzovanie-vebinarov-v-u chebном-processe (дата обращения: 22.03.2014).

2. Онлайн-семинар // Википедия URL: <http://ru.wikipedia.org/wiki/%C2%E5%E1%E8%ED%E0%F0> (дата обращения: 25.03.2014).
3. Презентация учебной информации как один из способов успешного обучения // Центр качества профессионального образования URL: http://ckprom.portalspo.ru/konf2_9.php (дата обращения: 22.03.2014).
4. Удаление метаданных в презентации PowerPoint // Microsoft URL: <http://support.microsoft.com/kb/826825> (дата обращения: 20.03.2014).

РАЗРАБОТКА ПРИЛОЖЕНИЙ ДЛЯ МОБИЛЬНЫХ УСТРОЙСТВ ПОД УПРАВЛЕНИЕМ ОПЕРАЦИОННОЙ СИСТЕМЫ ANDROID

Сахипова Э.Ф.

*Научный руководитель: к.п.н., доцент кафедры ИИТ и МОИ Газейкина А.И.
Уральский государственный педагогический университет.*

Аннотация

В статье описывается процесс реализации мобильного приложения, выполняющего задачи геопозиционирования. Приводится описание обоснование выбора операционной системы (в данном случае ОС Android), используемых инструментов. Описывается основной функционал созданного приложения, используемый алгоритм для определения местоположения пользователя. Так же приводится краткое описание программной реализации, в которой упомянуты основные используемые компоненты. В конце описана схема тестирования приложения

Ключевые слова: мобильное приложение, геолокация, позиционирование, android, позиционирование, алгоритм.

ОБОСНОВАНИЕ ВЫБОРА ОПЕРАЦИОННОЙ СИСТЕМЫ

В настоящее время в сфере мобильных операционных систем фигурирует небольшое количество участников: Android, iOS, Windows Phone, BlackBerry OS, Sailfish, Ubuntu ONE и другие.

Многие из этих систем являются хорошими программными комплексами, позволяющими, среди прочего, достаточно эффективно создавать мобильные приложения. Но после тщательного анализа вышеупомянутых систем было решено использовать для разработки мобильную операционную систему Android. Решающими факторами стало следующее:

- большая распространенность;
- удобный в использовании API – программный интерфейс насчитывает более 1000 компонентов, которые способны решить большинство часто встречаемых задач, такие как: создание пользовательского интерфейса, работа с сетью, работа с базами данных, использование аппаратных ресурсов устройства (Bluetooth, Wi-Fi, NFC, камера, аудиоустройства), GPS-навигация и так далее;

- открытость операционной системы, что позволяет разработчикам получить доступ к исходному коду Android и понять, каким образом реализованы свойства и функции приложений;
- наличие достаточно количества инструментария, в том числе IDE, редактора графического пользовательского интерфейса, различное множество эмуляторов;
- удобный и эффективный язык Java, используемый при написании приложений.

ИСПОЛЬЗУЕМЫЕ ИНСТРУМЕНТЫ

В качестве среды для разработки Android-приложений было решено использовать интегрированную среду разработки Eclipse, позволяющую использовать плагин Android Development Tools (ADT) for Eclipse. Данный плагин обеспечивает возможность использования инструментов Android SDK для разработки приложений Android с помощью интегрированной среды разработки Eclipse. Инструментальный набор Android Software Development Kit (SDK) включает средства, используемые для разработки, тестирования и отладки приложений Android, а именно:

- менеджер компонентов, позволяющий установить пакеты для различных версий API;
- Android Virtual Device (AVD) – менеджер виртуальных устройств эмуляторов, позволяющий тестировать созданное приложение с разу на большом количестве устройств, в частности различающихся разрешением и диагональю экрана;
 - средства тестирования;
 - документация;
 - Android Debug Bridge (ADB) – средство позволяющее запускать и отлаживать приложение на физическом устройстве под управление ОС Android, подключенном компьютеру.

При разработке приложений Android используется язык программирования Java, являющийся одним из наиболее распространенных языков. Java является кроссплатформенным языком. Для разработки Android-приложений используется набор Java Development Kit (JDK).

ФУНКЦИОНАЛ ПРИЛОЖЕНИЯ И СХЕМА РАБОТЫ

Реализованное в ходе исследования приложение используется для сигнализирования пользователя о достижении определенного расстояния до указанного остановочного пункта. Сигнализирование производится путем подачи аудио и/или вибросигнала.

После запуска приложения пользователь выбирает нужную ему остановку и необходимое расстояние до остановки. При достижении ранее установленного расстояния, программа сигнализирует об этом пользователю.

Для получения координат целевого местоположения используется ранее заполненная база данных, в которых хранятся пары “остановка” = “координаты”.

Для определения текущего местоположения пользователя используется GPS навигация, либо же, если GPS недоступно, используется ориентация по сотовым вышкам.

Во время работы программы происходит постоянный расчет расстояния между пользователем и целевой точкой, используя алгоритм, описанный ниже. При достижении расстояния указанного ранее значения производится оповещение пользователя.

ИСПОЛЬЗУЕМЫЕ ПРОГРАММНЫЕ КОМПОНЕНТЫ

Для получения данных о местоположении устройства использовался пакет `android.location`, который включает классы и интерфейсы, предназначенные для получения и использования данных о местоположении. Класс `LocationManager` обеспечивает доступ к службам геолокации устройства. Эти службы зависят от используемого оборудования и могут применяться для периодического обновления сведений о местоположении устройства либо для запуска объекта `Intent` в случае, если пользователь направляется в определенный географический регион. В зависимости от используемого устройства поддерживаются несколько провайдеров геолокации. Класс `LocationManager` обеспечивает возможность выбора лучшего провайдера на основании требований приложения, определенных с помощью объекта `Criteria`. Настройки, которые могут быть указаны с помощью `Criteria`, — точность, использование батареи, отслеживание перемещения, скорость, высота и стоимость услуг провайдера. Все запросы отправляются к поставщику данных о местоположении, и передаются в класс `LocationListener`. Обновленные сведения передаются слушателю в качестве объектов `Location`, которые представляют географическое местоположение устройства, — сведения о широте и долготе, время записи данных и (в зависимости от поставщика данных о местоположении) данные о высоте и скорости перемещения (у некоторых устройств просто отсутствуют соответствующие сенсоры). Чтобы определить наличие «захвата» сигнала GPS, то есть приема сигналов GPS с достаточно большого количества спутников, необходимого для точного отслеживания местоположения.

Класс `PowerManager` (пакет `android.os`) обеспечивает приложению возможность управления питанием устройства Android. Приложение, изменяющее настройки питания, может оказать отрицательное влияние на время «жизни» батареи устройства при выполнении приложения, поэтому следует параллельно использовать класс `PowerManager`.

Как только приложение начинает отслеживать устройство, требуется, чтобы приложение фиксировало данные о местоположении, даже если отключен экран устройства. Для этого используется класс `PowerManager` для перехода в режим `WakeLock`, в котором предотвращается переход устройства

в режим сна, в результате чего приложение может продолжать принимать GPS-данные

Информация об остановках города хранится в базе данных SQLite. Классы Activity, используемые в приложении, взаимодействуют с базой данных SQLite с помощью класса утилиты DatabaseConnector. Внутри этого класса использовался вложенный подкласс класса SQLiteOpenHelper, который упрощает создание базы данных и позволяет воспользоваться объектом SQLiteDatabase для работы с содержимым базы данных. Для управления результатами запроса к базе данных используется класс Cursor.

ИСПОЛЬЗУЕМЫЙ АЛГОРИТМ

Данный алгоритм используется для получения текущих координат и для вычисления расстояния между устройством и целевой точкой.

- Предварительный этап: выбирается лучший «провайдер». Провайдером является либо GPS модуль, либо Интернет-соединение сотового оператора. После поиска идентификатор провайдера заносится в объект типа String.

- Получение текущих координат:
 - устанавливается связь между провайдером, менеджером локации (LocationManager) и слушателем посредством вызова метода locationManager.requestLocationUpdates(), в качестве параметров которому передаются объект типа String, содержащая идентификатор провайдера и объект типа MyLocationListener;

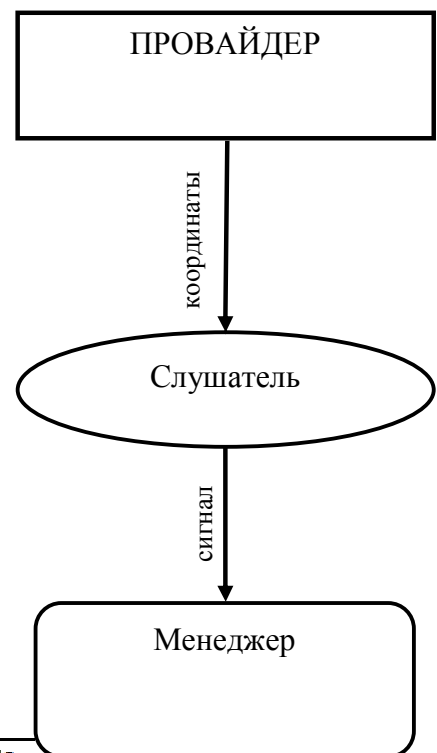
- при поступлении новых координат от провайдера менеджер местоположения вызывает функцию distance(), которая обновляет на экране координаты. Так же рассчитывается расстояние до заданной остановки;

- расчет расстояния между координатами: используется формула гаверсинусов – формула расчета расстояния между двумя точками, заданными координатами, на сфере

$$\Delta\sigma = 2 * \arcsin \left\{ \sqrt{\sin^2 \left(\frac{\varphi_2 - \varphi_1}{2} \right) + \cos\varphi_1 \cos\varphi_2 \sin^2 \left(\frac{\lambda_2 - \lambda_1}{2} \right)} \right\}$$

где λ_1 , φ_1 , λ_2 и φ_2 - широта и долгота двух точек в радианах. Для перевода углового расстояния в метрическое, необходимо угловую разницу умножить на радиус Земли (6372795 метров);

Таким образом, в процессе исследования разработано приложение, с помощью которого выполняется отслеживание перемещения устройства и вычисляется расстояние до нужной точки. Язык программирования Java может успешно применяться для разработки приложений для мобильных устройств, работающих под управлением операционной системы Android. Одна-



ко в процессе разработки необходимо использовать сторонние библиотеки, например, пакет android.location, который был использован при реализации представленного проекта.

Библиографический список

1. Android API Reference [Официальный сайт]. URL: <http://developer.android.com/guide/components/index.html>
2. Laird Dornin, Blake Meike, Masumi Nakamura, "Programming Android", USA, O'Reilly Media, Inc., 2011, 482 с.
3. П. Дейтел, Х. Дейтел, Э. Дейтел, М. Моргано, "Android для программистов: создаем приложение", СПб.: Питер, 2013, 560 с.
4. Зигард Медникс, Лайрд Дорнин, Дж. Блейк Микб Максуми Накамура, "Программирование под Android", СПб, Питер, 2012, 496 с.

ФОРМИРОВАНИЕ ИКТ КОМПЕТЕНТНОСТИ СРЕДСТВАМИ ТЕХНОЛОГИИ WEB-QUEST

Соколова З.И., магистрант УрГПУ

Научный руководитель: профессор Семенова И.Н.

Уральский государственный педагогический университет

Аннотация

В статье обосновывается возможность использования WEB-quest для формирования ИКТ компетентности. Предлагается систематизация типов WEB-quest, выделенная в соответствии с этапами решения информационных задач, составляющих ядро ИКТ-компетентности.

Ключевые слова: WEB-quest, ИКТ-компетентность, информационные задачи, этапы решения информационных задач.

Zinaida I. Sokolova

FORMATION OF ICT COMPETENCE MEANS TECHNOLOGY WEB-QUEST

Abstract

The article substantiates the possibility of the use of the WEB-quests for the formation of ICT competence. It is proposed to systematize types WEB-quest, allocated in accordance with the stages of solving information problems, forming the core of ICT competence.

Keywords: WEB quest, ICT competence, information tasks, stages of solving information problems.

В «Законе об образовании», Концепции модернизации Российского образования до 2020 года декларируются новые требования к выпускнику общеобразовательной школы. Одним из таких требований является формирование ИКТ-компетентности.

Анализируя ИКТ-компетентность, В. Ф. Бурмакина, М. Зелман, И. Н. Фалина отмечают, что правильно построенная программа формирования ИКТ-компетентности не должна сводиться к простому перечню тех знаний и умений, которыми учащиеся должны овладеть (знание устройства компьютера, навыки работы с текстовым редактором, умение искать и находить нужную информацию в Интернете). Авторы определяют ИКТ-компетентность как модель, состоящую из семи этапов, охватывающих решение всевозможных информационных задач, являющихся по существу ядром ИКТ-