

СОЗДАНИЕ *ANDROID*-ПРИЛОЖЕНИЯ ДЛЯ ДИСПЕТЧЕРСКОГО ОТОБРАЖЕНИЯ ИНФОРМАЦИИ В *SCADA* "СИСТЕЛ"

Филиал «Протвино» университета «Дубна»
Кафедра информационных технологий

Рассматривается распространенность мобильных устройств в современном мире и актуальность создания приложений для работы на таких устройствах, в том числе, в области энергетики. Описывается клиентское приложение, созданное для мобильных устройств в составе *SCADA* «Систел».

На протяжении последних лет все большее развитие получают различные мобильные устройства, по своим функциональным возможностям ставшие полноценной альтернативой традиционным персональным компьютерам. Распространенность таких устройств заставляет учитывать необходимость их интеграции в процесс взаимодействия различных составных частей программных комплексов, в том числе систем *SCADA* (*Supervisory Control And Data Acquisition* — диспетчерское управление и сбор данных), которые представляют собой программные комплексы, предназначенные для обеспечения работы систем сбора, обработки, отображения и архивирования информации об объекте мониторинга или управления. Работа с мобильными устройствами существенно отличается от работы с персональным компьютером — в связи с этим необходимо создавать приложение, которое, выполняя необходимые функции, будет при этом учитывать особенности работы с таким устройством.

Одной из наиболее популярных операционных систем для мобильных устройств является *Android*, которая позволяет создавать на языке *Java* приложения, управляющие устройством через специальные библиотеки. Также, с использованием дополнительных инструментов, можно создавать приложения, написанные на других языках. К основным преимуществам *Android* над другими операционными системами для портативных устройств можно отнести ее открытость, высокую скорость работы, хорошую возможность настройки и простой процесс обновления системы.

SCADA «СИСТЕЛ» является частью программно-технического комплекса «СИСТЕЛ», разработанного на базе предприятия ООО «Системы телемеханики и автоматизации» (г. Протвино) [1]. В состав комплекса входит сервер сбора и обработки телеинформации для систем диспетчерского управления — функционально законченный набор задач, выполняющий услуги по получению данных и управлению ими. Также составной частью комплекса являются автоматизированные рабочие места, являющиеся клиентской частью системы и представляющие собой программное обеспечение, позволяющее просматривать информацию, предоставляемую оперативным информационно-управляющим комплексом, осуществлять управляющие воздействия, контролировать работу комплекса.

Информационный обмен данными между сервером и клиентскими приложениями организован через стандартные сетевые протоколы связи, что позволяет значительно оптимизировать поток информации между составными частями системы.

В рамках выполнения данной работы была поставлена задача разработать, в составе клиентской части программного комплекса *SCADA* «СИСТЕЛ», приложение, работающее на устройствах под управлением операционной системы *Android*. Приложение должно получать от сервера список устройств, подключенных к системе, отображать их на экране, а также предоставлять пользователю возможность выбрать одно из устройств и получать информацию о состоянии всех его сигналов с определенной периодичностью.

Для разработки приложения был выбран язык программирования *Java* [2] с использованием набора инструментов *Android Software Development Kit (SDK)* [3]. *SDK* представляет собой набор инструментов, необходимых при разработке приложений для операционной системы *Android*. В *SDK* входят:

- файл архива *android.jar*, содержащий все классы *SDK Android*, необходимые для создания приложений;
- документация;

- каталог с примерами;
- каталог инструментов, содержащий все инструменты командной строки для создания *Android*-приложений;
- драйверы для подключения среды разработки к поддерживающим *Android* устройствам;
- отладчик;
- эмулятор телефона.

Взаимодействие приложения с сервером происходит с использованием встроенного класса *Socket*, реализующего интерфейс сокетов для обеспечения обмена данными между процессами (в данном случае, выполняющимися на различных ЭВМ). Для обмена информацией между клиентским приложением и сервером в приложении использован формат *XML* (*Extensible Markup Language*) — расширяемый язык разметки документов, фактически представляющий собой свод общих синтаксических правил. *XML* — текстовый формат, предназначенный для хранения структурированных данных, для обмена информацией между программами, а также для создания на его основе более специализированных языков разметки.

Для реализации взаимодействия с созданным клиентским приложением в серверной части системы был создан отдельный поток данных [4]. Данный поток взаимодействует через один и тот же порт со всеми экземплярами приложения, запущенными на разных мобильных устройствах. С установленной периодичностью сервер получает информацию об изменениях, произошедших в состоянии сигналов устройств, и пересылает новые параметры клиентскому приложению. Передача данных осуществляется в виде строки в формате *XML* с использованием стека протоколов *TCP/IP*.

Созданное в рамках выполнения данной работы приложение по своей функциональности является упрощенной версией клиентского приложения, предназначенного для работы на персональном компьютере. Оно отображает список устройств, подключенных к системе, их состояние в данный момент времени и параметры их сигналов.

При запуске приложение посылает серверу сообщение, содержащее слово «*CONNECT*», и получает в ответ *XML*-строку со списком устройств системы и сигналов каждого из них. Каждое устройство имеет один сигнал, отвечающий за само соединение с сервером. При этом параметры самих сигналов на этом этапе не передаются (за исключением сигнала соединения), что позволяет уменьшить объем передаваемой информации.

Полученный список отображается на экране, при этом рядом с каждым устройством существует графическое изображение, отвечающее за состояние сигнала соединения (рис. 1). Этот сигнал (как и все остальные) может иметь состояния: достоверный, недостоверный, на ручном вводе, на автоматическом вводе.

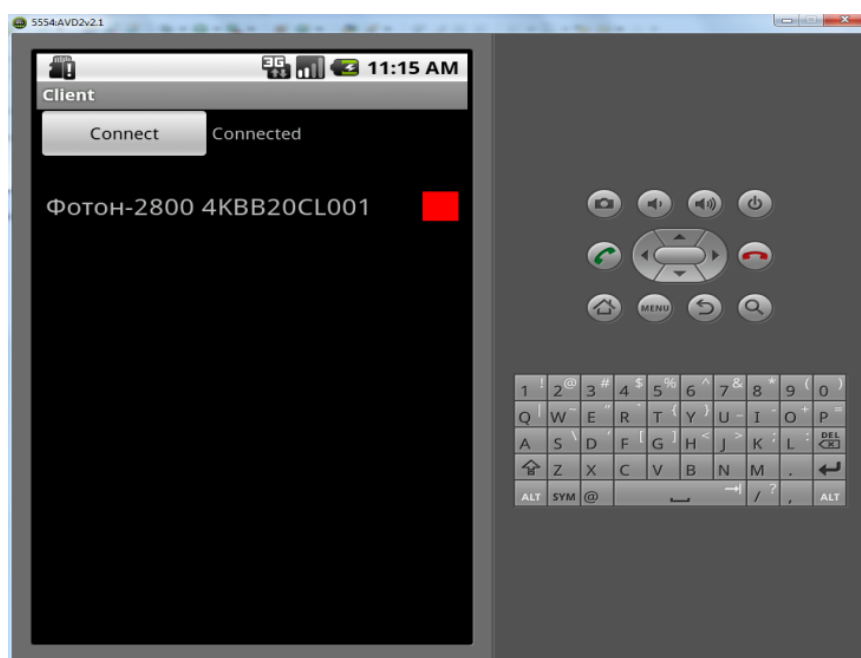


Рис. 1 Внешний вид окна приложения, отображающего список устройств

При выборе пользователем одного из устройств приложение передает серверу его идентификатор. В ответ сервер передает XML-строку, содержащую значения, отвечающие за параметры сигналов выбранного устройства. Список сигналов отображается на экране, при этом рядом с его названием отображается графическое изображение, отвечающее за его состояние, либо, для аналоговых сигналов, значение конкретной величины (рис. 2).

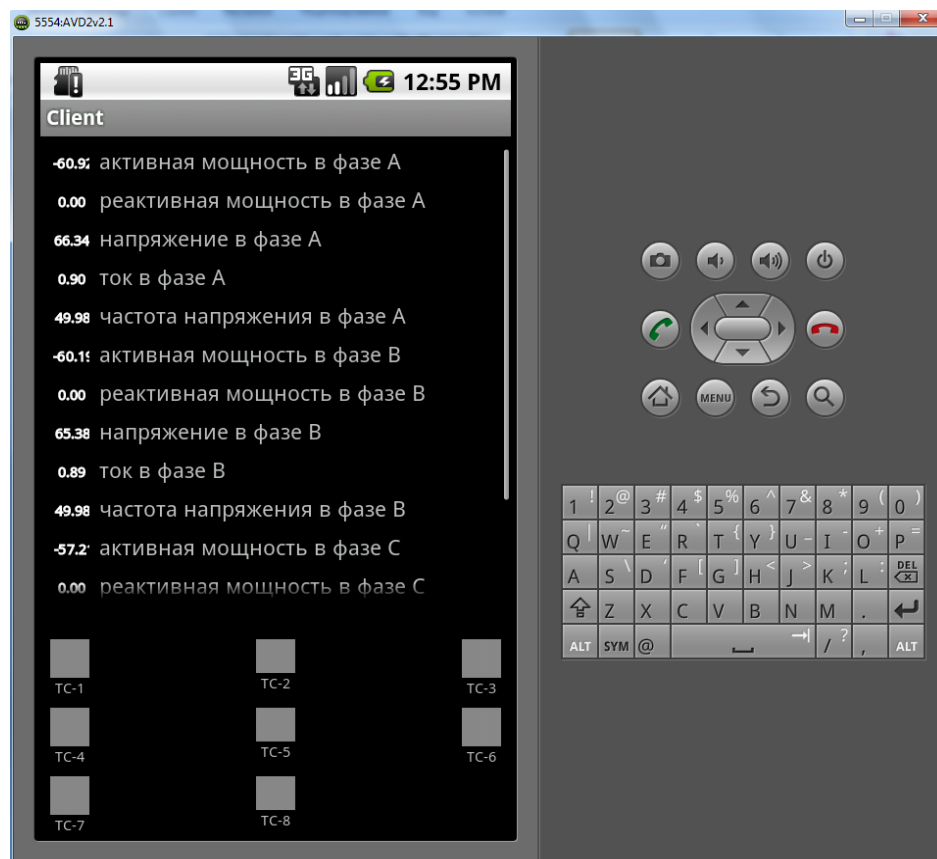


Рис. 2 Внешний вид окна приложения, отображающего состояния сигналов выбранного устройства

Сразу после запуска приложение создает отдельный поток, отслеживающий получение новой информации от сервера. С определенной периодичностью приложение получает информацию о состоянии сигналов и производит необходимые изменения на экране.

При нажатии кнопки возврата к списку устройств приложение отправляет условное сообщение серверу, который заново передает информацию о состоянии только тех сигналов, которые отвечают за соединения с соответствующими устройствами для отображения соответствующего графического изображения рядом с каждым из устройств.

Таким образом, в результате выполнения данной работы в составе SCADA «СИСТЕЛ» было создано клиентское приложение, работающее на мобильных устройствах под управлением операционной системы *Android*. Данное приложение будет внедряться в работу в процессе дальнейшей эксплуатации SCADA «СИСТЕЛ».

Библиографический список

1. Рыкованов, С. Н. Межотраслевой производственно-технический «Автоматизация от А до Я» — / С. Н. Рыкованов, Н. Ю. Кульман., В. И. Ухов — Минск, №1 (32), 2007. С. 9—11.
2. Вебер, Дж. Технология Java. / Дж. Вебер. — СПб. : БХВ-Петербург, 2001. — 1104 с.;
3. Голощапов, А. Л. Google Android: программирование для мобильных устройств. / А. Л. Голощапов — СПб.: БХВ-Петербург, 2011. — 448 с.;
4. Шеферд, Дж. Программирование на Microsoft Visual Studio C++.NET. / Дж. Шеферд — М. : Издательско-торговый дом «Русская Редакция», 2003. — 928 с..