

УДК 004.771

## ПОДХОДЫ К РАЗРАБОТКЕ ИНТЕРФЕЙСА ПРИЛОЖЕНИЯ ДЛЯ УДАЛЕННОГО УПРАВЛЕНИЯ КОМПЬЮТЕРОМ ПРИ ПОМОЩИ ANDROID УСТРОЙСТВА

*И.А. ШУМЯНЦЕВ**(Представлено: канд. техн. наук, доц. А.Ф.ОСЬКИН)*

*В статье представлен практический способ создания интерфейса приложения для удаленного управления компьютером при помощи android устройства. Проведён анализ технологий, наиболее подходящих, для разработки данного приложения. Прототип интерфейса реализован на основе Windows Presentation Foundation.*

**Введение.** На сегодняшний день, любое приложение должно быть основано на общеподдерживаемых стандартах разработки, а также обладать понятным для пользователя интерфейсом. Разрабатываемое приложение для удаленного управления компьютером при помощи android устройства должно позволять производить работу с компьютером посредством wi-fi или любой другой сети при помощи android устройства.

В данном проекте процесс создания интерфейса приложения служит для прямого взаимодействия приложения с сетью, сочетающей в себе всю необходимую информацию для вывода непосредственно в приложение. Интерфейс создаётся путем добавления необходимых компонентов и последующей их настройки. В интерфейсе любого приложения есть компоненты, обеспечивающие взаимодействие программы с пользователем.

**Выбор технологий разработки интерфейса.** Для разработки данного приложения выбран язык программирования C#.

C# – объектно-ориентированный язык программирования. Разработан в 1998-2001 годах группой инженеров под руководством Андерса Хейлсберга в компании Microsoft как язык разработки приложений для платформы Microsoft .NET Framework и впоследствии был стандартизирован как ECMA-334 и ISO/IEC 23270 [1].

Для разработки Windows-версии нами была использована система для построения клиентских приложений Windows Presentation Foundation (WPF) – система для построения клиентских приложений Windows. Достоинства системы — высокая визуальная привлекательность. Графическая (презентационная) подсистема входит в состав .NET Framework (начиная с версии 3.0) и использует язык XAML [2].

Для разработки приложения была выбрана среда разработки Visual Studio 2017, т.к. она поддерживает синтаксис большинства языков программирования и множество плагинов для них. Кроме того, это одна из лучших сред для разработки десктопных приложений.


Visual Studio включает в себя редактор исходного кода с поддержкой технологии IntelliSense и возможностью простейшего рефакторинга кода. Встроенный отладчик может работать как отладчик уровня исходного кода, так и отладчик машинного уровня. Остальные встраиваемые инструменты включают в себя редактор форм для упрощения создания графического интерфейса приложения, веб-редактор, дизайнер классов и дизайнер схемы базы данных. Visual Studio позволяет создавать и подключать сторонние дополнения (плагины) для расширения функциональности практически на каждом уровне, включая добавление поддержки систем контроля версий исходного кода (как, например, Subversion и VisualSourceSafe), добавление новых наборов инструментов (например, для редактирования и визуального проектирования кода на предметно-ориентированных языках программирования) или инструментов для прочих аспектов процесса разработки программного обеспечения (например, клиент Team Explorer для работы с Team Foundation Server).

Visual Studio построена на архитектуре, поддерживающей возможность использования встраиваемых дополнений – плагинов от сторонних разработчиков, что позволяет расширять возможности среды разработки [3].

**Проектирование интерфейса.** Расположение элементов пользовательского интерфейса не должно предоставлять затруднений при работе с приложением. Для создания удобного интерфейса необходимо придерживаться чёткого визуального разделения областей экрана.

Было принято решение о вынесении всех пунктов управления в так называемые Grid-блоки с последующим распределением в Text-блоки и Button-блоки. Таким образом, перед пользователем наглядно будет отображаться вся необходимая информация, а элементы управления приложением будут отображаться понятно и корректно.

При запуске программы на компьютере должно появиться окно, с помощью которого можно будет запускать, и останавливать работу сервера. Макет представлен на рисунке 1.

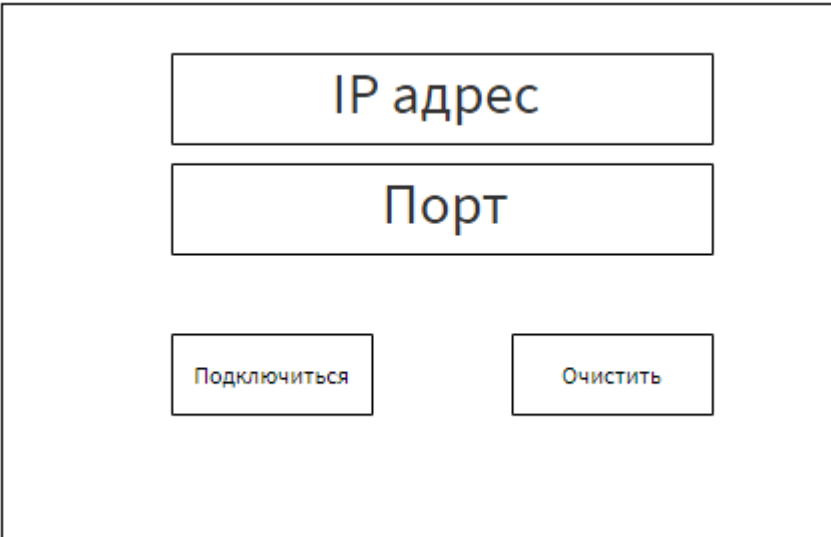


Макет серверного приложения. В центре экрана расположены три текстовых поля: «IP адрес», «Порт» и «Статус». Под полями «IP адрес» и «Порт» находятся две кнопки: «Старт» и «Отмена». Поле «Статус» расположено под кнопками.

**Рисунок 1. – Макет серверного приложения**

Поле «IP адрес» содержит в себе информацию о IP. Поле «Порт» содержит случайно сгенерированный порт. Поле «Статус» содержит в себе информацию об активности сервера. Кнопка «Старт» создает новые потоки, в которых реализованы функции для отправки и приема данных. Кнопка «Отмена» закрывает все запущенные потоки (кроме главного) и сокеты.

В клиентской части проекта есть два окна. В первом происходит ввод необходимой информации для подключения к серверу (рисунок 2), а во втором окне происходит само управление (рисунок 3).



Макет меню подключения к серверу. В центре экрана расположены три текстовых поля: «IP адрес», «Порт» и «Подключиться». Под полями «IP адрес» и «Порт» находятся две кнопки: «Подключиться» и «Очистить». Поле «Подключиться» расположено под кнопками.

**Рисунок 2. – Макет меню подключения к серверу**

В поле «IP адрес» необходимо ввести IP сервера, к которому нужно подключиться. В поле «Порт» необходимо ввести порт, заданный на сервере. При нажатии на кнопку «Очистить» происходит очистка вышеперечисленных полей. При нажатии на кнопку «Подключиться» на сервер отправляются данные о телефоне, а именно его IP адрес.

В это окно выводится изображение рабочего стола. Левая область экрана, которая занимает 10% от всей ширины экрана, осуществляет функции стандартных кнопок мыши. Управлять курсором можно при помощи перемещения пальца по любой части экрана.

90%	10%
Изображение	Левая кнопка мыши
	Правая кнопка мыши

Рисунок 3. – Макет окна управления

**Заключение.** Правильным подходом при разработке интерфейса приложения является использование современных технологий. Именно поэтому было принято решение использовать WPF. Это, в первую очередь, экономит время при разработке интерфейса, а также ресурсы на обработку данных. Это является очень актуальным, при большом количестве информации, которая подлежит постоянному изменению.

Рассмотрены технологии WPF и XAML, при взаимодействии, позволяют создать адаптивный интерфейс, любого уровня сложности, при этом сохраняя его функциональные возможности и привлекательный внешний вид. По итогам данной работы, была описана возможность их применения к построению интерфейса для удобной работы пользователя.

#### ЛИТЕРАТУРА

1. METANIT.COM Сайт о программировании [Электронный ресурс]. Режим доступа: <https://metanit.com/sharp/>. Дата доступа: 23.09.2019.
2. Microsoft.com. Xamarin [Электронный ресурс]. Режим доступа: <https://docs.microsoft.com/ru-ru/xamarin/>. Дата доступа: 23.09.2019.
3. Microsoft.com — свободной энциклопедии. Visual Studio [Электронный ресурс]. Режим доступа: <https://visualstudio.microsoft.com/ru/?rr=https%3A%2F%2Fwww.google.com%2F>. – Дата доступа: 23.09.2019.