

Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Российский государственный профессионально-педагогический университет»

**ЭЛЕКТРОННОЕ УЧЕБНОЕ ПОСОБИЕ «РАЗРАБОТКА WPF  
ПРИЛОЖЕНИЙ В C#»**

Выпускная квалификационная работа  
по направлению подготовки 44.03.04 Профессиональное обучение  
(по отраслям)  
профиля подготовки «Информатика и вычислительная техника»  
специализации «Компьютерные технологии»

Идентификационный номер ВКР: 503

Екатеринбург 2018

Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Российский государственный профессионально-педагогический университет»  
Институт инженерно-педагогического образования  
Кафедра информационных систем и технологий

К ЗАЩИТЕ ДОПУСКАЮ

Заведующая кафедрой ИС

\_\_\_\_\_ Н. С. Толстова

« \_\_\_\_ » \_\_\_\_\_ 2018 г.

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**  
**ЭЛЕКТРОННОЕ УЧЕБНОЕ ПОСОБИЕ «РАЗРАБОТКА WPF**  
**ПРИЛОЖЕНИЙ В C#»**

Исполнитель:

обучающийся группы ЗКТ-401С

А. О. Балакин

Руководитель:

ст. преподаватель

С. Н. Ширева

Нормоконтролер:

Н. В. Хохлова

## АНОТАЦИЯ

Выпускная квалификационная работа состоит из электронного учебного пособия «Разработка WPF приложений в C#» и пояснительной записки на 55 страницах, содержащей 28 рисунков, 31 источник литературы, а так же 1 приложение на 1 странице.

Ключевые слова: ЭЛЕКТРОННОЕ УЧЕБНОЕ ПОСОБИЕ, WPF, C#.

**Балакин, А. О.** «Разработка WPF приложений в C#»: выпускная квалификационная работа / А. О. Балакин; Рос. гос. проф.-пед. ун-т, Ин-т инж.-пед. образования, Каф. информ. систем и технологий. — Екатеринбург, 2018. — 56 с.

В работе рассмотрен процесс создания электронного учебного пособия «Разработка WPF приложений в C#», поскольку существует необходимость в структурировании лекционного материала и лабораторного практикума в одном электронном пособии.

Целью работы является разработать структуру и интерфейс электронного учебного пособия по дисциплине «Алгоритмические языки и системы программирования» на тему «Разработка WPF приложений в C#».

Для достижения цели были проанализированы рабочая программа дисциплины «Алгоритмические языки и системы программирования», литература и различные интернет-источники по теме исследования, проанализированы две учебно-методические разработки по WPF, отобран и структурирован материал, разработаны структура и интерфейс электронного учебного пособия. Пособие реализовано в электронном виде и содержит в себе теоретический материал, лабораторные работы, блок контроля, глоссарий.

С внедрением электронного учебника, у преподавателя появляется больше времени на индивидуальную работу с каждым студентом. Снижается потребность в издании методического материала на бумажных носителях.

# СОДЕРЖАНИЕ

Введение .....	5
1 Теоретический обзор учебно-методических материалов по теме «создание windows presentation foundation приложений в C#» .....	7
1.1 Характеристика рассматриваемой предметной области .....	7
1.1.1 Общая характеристика предметной области.....	7
1.1.2 Основные понятия и определения .....	10
1.1.3 Анализ существующих учебно-методических разработок по windows presentation foundation .....	11
1.1.4 Анализ литературы .....	12
1.1.5 Анализ интернет-источников.....	15
1.2 Педагогический анализ рассматриваемой предметной области.....	16
1.2.1 Характеристика учебного заведения .....	16
1.2.2 Педагогический адрес .....	16
1.2.3 Расширенный педагогический адрес .....	16
1.2.4 Анализ рабочей программы .....	17
1.2.5 Характеристика и специфические особенности текущего состояния педагогического процесса.....	20
1.3 Требования к разрабатываемому педагогическому продукту и выбор средств для его реализации.....	22
1.3.1 Понятие электронного учебного пособия .....	22
1.3.2 Требования к электронным учебным пособиям .....	24
1.3.3 Анализ и выбор средств для реализации продукта.....	25
2 Описание электронного учебного пособия.....	32

2.1 Описание электронного учебного пособия.....	32
2.1.1 Структура электронного учебного пособия.....	32
2.1.2 Описание интерфейса электронного учебного пособия.....	32
2.1.3 Главная страница.....	36
2.1.4 Теоретический блок.....	37
2.1.5 Практический блок.....	39
2.1.6 Блок контроля.....	44
2.1.7 Глоссарий.....	47
2.2 Методические указания по использованию электронного учебного пособия в учебном процессе.....	48
2.2.1 Инструкции по работе с электронным учебным пособием.....	49
2.2.2 Инструкция по контролю знаний.....	50
Заключение.....	51
Список использованных источников.....	53
Приложение.....	56

## ВВЕДЕНИЕ

В связи со стремительно растущей ИТ (Information Technology) индустрией Microsoft не перестает разрабатывать все более новые и совершенные программные продукты для Windows, а также средства для их разработки. Если раньше создание калькулятора и или любого другого маленького приложения занимало большое количество времени, то с появлением платформы .NET все изменилось.

В данном электронном учебном пособии будет описано использование платформы WPF (Windows Presentation Foundation) для разработки программ среды Microsoft Windows. Эти программы могут быть как автоматизированными приложениями Windows, так и интерфейсными модулями для распределенных приложений. Windows Presentation Foundation является основным интерфейсом прикладного программирования API (Application Programming Interface) для системы Microsoft Windows.

Хотя Windows Presentation Foundation используется для разработки программ, называемых «приложениями», они имеют мало общего с Windows-приложениями прошлого поколения. Windows Presentation Foundation значит новый программный интерфейс, новое визуальное оформление и новые средства разработки графического интерфейса (анимация и 3D-графика).

Программы Windows Presentation Foundation могут быть написаны на любом объектно-ориентированном языке программирования (C#, C+ и др.), компилируемом с правилами .NET CLS (Common Langue Specification).

В среде средств обучения, для студентов особую важность имеют электронные учебные пособия (ЭУП).

Электронное учебное пособие представляет собой совокупность теоретического, справочно-информационного, практического материала и заданий для тренинга, контроля и оценки качества усвоения знаний. Электронное

учебное пособие можно постоянно дополнять широким представлением изучаемого материала в виде презентаций, лабораторных работ, а также возможностью проверки усвоения знаний путем прохождения различных тестов и практических заданий. Электронное учебное пособие формируется с помощью специальных программ, позволяющих представить информацию в виде текстового, графического изображения, а также мультимедийных видео и звуковых эффектов.

**Объект исследования** — процесс обучения студентов направления 09.03.02 Информационные системы и технологии профиль подготовки «Информационные технологии в медиаиндустрии»

**Предмет исследования** — учебные материалы по теме «Разработка WPF (Windows Presentation Foundation) приложений в C#» дисциплины «Алгоритмические языки и системы программирования».

**Цель работы** — разработать электронное учебное пособие по дисциплине «Алгоритмические языки и системы программирования» на тему «Разработка WPF приложений в C#».

Задачи работы:

1. Изучить литературу и интернет-источники с целью формирования знаний по теме «Разработка WPF приложений в C#».
2. Проанализировать литературу и интернет-источники по верстке страниц с помощью средств PHP (Personal Home Page), HTML (Hyper Text Markup Language) и CSS (Cascading Style Sheets).
3. Проанализировать литературу и интернет-источники с целью выделения требований, предъявляемых к электронному учебному пособию на современном этапе развития образования.
4. Спроектировать структуру и интерфейс электронного учебного пособия.
5. Реализовать электронное учебное пособие на тему «Разработка WPF приложений в C#».

# **1 ТЕОРЕТИЧЕСКИЙ ОБЗОР УЧЕБНО-МЕТОДИЧЕСКИХ МАТЕРИАЛОВ ПО ТЕМЕ «СОЗДАНИЕ WINDOWS PRESENTATION FOUNDATION ПРИЛОЖЕНИЙ В C#»**

## **1.1 Характеристика рассматриваемой предметной области**

### **1.1.1 Общая характеристика предметной области**

Платформа .NET принесла с собой небольшую лавину новых технологий. Появился совершенно новый способ написания веб-приложений (ASP.NET), совершенно новый способ подключения к базам данных (ADO.NET), новые безопасные к типам языки (C# и VB.NET) и управляемая исполняющая среда (CLR). Не последнее место занимала и технология Windows Forms — библиотека классов для построения Windows-приложений.

С годами в Microsoft разработали многочисленные инструменты для создания пользовательского интерфейса (C/C++/Windows API, VB6, MFC и т.д.), предназначенные для построения настольных приложений. Каждый из этих программных инструментов предлагает кодовую базу для представления основных аспектов приложения с графическим интерфейсом, включая главные окна, диалоговые окна, элементы управления, системы меню и т.п. В начальном выпуске платформы .NET API-интерфейс Windows Forms быстро стал предпочтительной моделью разработки пользовательских интерфейсов, благодаря его простой, но очень мощной объектной модели.

Хотя с помощью Windows Forms было успешно разработано множество полноценных настольных приложений, следует признать, что его программная модель довольно ассиметрична. Просто говоря, сборки System.Windows.Forms.dll и System.Drawing.dll не обеспечивают прямой под-



держки многих дополнительных технологий для построения полноценного настольного приложения.

Например, чтобы построить стилизованную блестящую кнопку, придется создать специальный элемент управления и нарисовать каждый аспект этой новой кнопки (во всех разных состояниях), используя низкоуровневую модель рисования. Хуже того, обычные окна разрезаются на отдельные области, и каждому элементу управления отводится собственная такая область. В результате не существует такого хорошего способа рисования в одном элементе управления (например, эффекта свечения под кнопки), чтобы он распространялся на область, принадлежащую другому элементу. И не стоит даже думать о создании анимационных эффектов вроде вращающегося текста, мерцающих кнопок, сворачивающихся окон или активных предварительных просмотров, потому что каждая деталь должна быть нарисована вручную.

Технология Windows Presentation Foundation (WPF) является частью платформы .NET и представляет собой подсистему для проектирования графических интерфейсов.

Если при создании приложений на основе Windows Forms за обрисовку элементов управления и графики отвечали такие части операционной системы Windows, как User32 и GDI+, то WPF приложения основываются на DirectX. Используя WPF, значительная часть работы по отрисовке графики (3D-моделей, кнопок) ложится на графический процессор видеокарты.

В своей основе технология WPF содержит язык разметки на базе XAML. В отдельных случаях на XAML можно написать целую программу, однако чаще приложения строятся из программного кода и из кода разметки. Обычно XAML используется для изготовления визуального оформления приложения. Можно создавать красивый графический интерфейс, используя код на управляемых языках C# и VB.NET, или декларативное объявление интерфейса, либо совмещать тот и другой способ. Код обработки событий пишется отдельно.

К основным достоинствам языка можно отнести:

- использование языков платформы .NET — VB.NET и C# для создания логики приложения;
- возможность декларативного определения графического интерфейса с помощью XAML;
- качество внешнего вида не зависит от разрешения экрана;
- по сравнению с Windows Forms упрощено: привязка данных, создание трехмерных моделей, использование таких элементов, как стили, шаблоны, темы и др;
- возможность взаимодействовать с Windows Forms, благодаря чему, в приложениях WPF можно использовать традиционные элементы управления из Windows Forms;
- аппаратное ускорение графики за счет графического процессора видеокарты — вне зависимости от того, работаете ли вы с 2D или 3D;
- создание приложений под ОС семейства Windows XP/7/8/10.

В тоже время WPF имеет определенные ограничения. Несмотря на поддержку трехмерной графики, для создания игр, лучше использовать другие средства — DirectX или специальные фреймворки, такие как Monogame или Unity.

По сравнению с приложениями на Windows Forms объем программы на WPF более ресурсоемкий к аппаратной части компьютера. Потребление оперативной памяти в процессе работы в среднем несколько выше.

В электронном учебном пособии рассматриваются теоретические и практические разделы по данной теме. Так же имеется глоссарий, содержащий список терминов и определений. Для контроля знаний разработаны тесты.

Теоретический раздел разделен на главы, начиная: с описания программы и ее внешнего вида, базовые кисти, концепции содержимого программы, кнопки и другие элементы управления; заканчивая, анимацией.

В разделе практика, студенты закрепят полученные знания из теории, разрабатывая приложения, имеющие в итоге готовый и рабочий вид.

В процессе изучения материала могут возникнуть сложности с пониманием тех или иных терминов, для такого случая и разработан глоссарий.

### **1.1.2 Основные понятия и определения**

**Фреймворк** — программная платформа, определяющая структуру программной системы; программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта.

**.NET Framework** — программная платформа, выпущенная компанией Microsoft в 2002 году. Основой платформы является общезыковая среда исполнения Common Language Runtime, которая подходит для разных языков программирования.

**DirectX** — это набор API, разработанных для решения задач, связанных с программированием под Microsoft Windows. Наиболее широко используется при написании компьютерных игр.

**Windows Presentation Foundation (WPF)** — система для построения клиентских приложений Windows с визуально привлекательными возможностями взаимодействия с пользователем, графическая (презентационная) подсистема в составе .NET Framework (начиная с версии 3.0), использующая язык XAML.

**Электронное учебное пособие** — современный программно-методический обучающий комплекс, соответствующий времени, потребностям студента и запросам практики.

**Декларативное описание интерфейса** — декларативность означает, что описывается желаемый результат, а не последовательность действий для его достижения. Примером такого подхода является HTML, который служит для описания контента, а не функций, при помощи которых он должен генерироваться.

**XAML** — представляет собой язык разметки, используемый для создания экземпляров объектов .NET. Хотя язык **XAML** — это технология, которая может быть применима ко многим различным предметным областям, его главное назначение — конструирование пользовательских интерфейсов WPF.

### **1.1.3 Анализ существующих учебно-методических разработок по windows presentation foundation**

Рабочая программа по дисциплине «Визуальное программирование» [17].

Целью изучения дисциплины является формирование у студентов практических навыков по основам визуального, объектно-ориентированного и декларативного программирования, необходимых для создания сложных программных комплексов. Ознакомление студентов с Windows Presentation Foundation (WPF). В данной дисциплине рассмотрены методы и способы создания компьютерной программы путём манипулирования графическими объектами. Также приводятся сведения о декларативном языке расширенной разметки для приложений **XAML**, основы технологии и базовые компоненты WPF, рассматриваются вопросы разработки приложений на базе WPF. Данная дисциплина позволяет научиться создавать «богатые» интерактивные программные интерфейсы с высокой функциональностью Windows приложений. WPF объединяет мультимедиа, графику, анимацию и интерактивность в одной программной платформе. Спектр использования технологии, эргономичность дизайна и многофункциональность создаваемых приложений WPF, простота проектирования позволяют широко применять данную технологию начиная с создания динамичных сайтов заканчивая созданием корпоративных бизнес приложений.

Рабочая программа по дисциплине «Технология разработки пользовательских интерфейсов» [16]. Дисциплина «Технология разработки пользовательских интерфейсов» является одной из дисциплин, на базе которых стро-

ится подготовка специалистов в области разработки и применения программного обеспечения средств вычислительной техники и автоматизированных систем. Целью данной дисциплины является знакомство с основными технологиями разработки пользовательских графических интерфейсов современных приложений и формирование знаний и навыков в области разработки интерфейсов на языке XAML с использованием системы WPF. Дисциплина «Технология разработки пользовательских интерфейсов» (М2. В.6.2) входит в состав вариативной части профессионального цикла дисциплин учебного плана (М2. В).

Для её успешного освоения необходимы базовые и специальные знания, полученные при изучении основной образовательной программы (ООП) бакалаврской подготовки по направлению «Информатика и вычислительная техника» и близких направлений.

#### **1.1.4 Анализ литературы**

Для разработки электронного учебного пособия на тему «Разработка WPF приложений в C#» при помощи PHP, JavaScript, HTML и CSS, проанализирован определенный перечень интернет-источников и литературы.

**Петсольд Ч.** «**Microsoft Windows Presentation Foundation**» [13]. В данной книге изложены два параллельных подхода программирования, на которых основывается Windows Presentation Foundation(WPF). При изучении данной книги циклично повторяются особенности применения языка XAML к тонкостям программирования на C#, что позволяет понять их взаимодействие при проектировании интерфейсов. Демонстрируется большое количество примеров по взаимодействию C# и XAML. Так же «фирменное» пояснение Чарльза Пецольда концепции пользовательских интерфейсов UI (user interface) легко учит разработчиков тому, как создавать интерфейсы нового поколения для своих приложений.

**Макдональд М. «WPF 4: Windows Presentation Foundation в .NET 4.0 с примерами на С# 2010 для профессионалов» [8].** Книга представляет углубленное руководство по WPF для профессионалов, знакомых с языком С#, платформой .NET и средой разработки Visual Studio. Опыт работы с предыдущими версиями WPF не обязателен.

Описываются все основные средства WPF — от XAML (языка разметки интерфейсов WPF) до разработки трехмерной графики и анимации. По ходу изучения происходит ознакомление с кодом, который включает работу с классами ADO.NET, служащими для запросов к базе данных.

**Натан А. «WPF 4. Подробное руководство» [11].** Эта книга рассчитана для разработчиков, заинтересованных в проектировании и создании пользовательских интерфейсов для Windows. Неважно, что вы разрабатываете: программы для массового потребителя или бизнеса, повторно используемые элементы управления — здесь можно найти сведения, позволяющие извлечь максимум пользы из платформы. Книга написана справочным и простым языком.

**Стиллмен Э., Грин Дж. «Изучаем С#» [18].** В отличие от большинства книг, построенных на основе скучного изложения спецификаций и примеров, в этой книге имеется возможность сразу приступить к практике, написанию собственного кода на языке С#. Описывается минимальный набор инструментов. Особенностью данного издания является уникальный способ подачи материала, выделяющий серию «Head First» издательства O'Reilly в ряду множества скучных книг, посвященных программированию.

**Хеник Б. «HTML и CSS: путь к совершенству» [20].** Данная книга рассказывает о том, как эффективно использовать в своей работе технологии HTML и CSS. Книга знакомит основным синтаксисом языка и предлагает способы решения тех или иных задач при верстке сайта.

**Бейли Л., Моррисон М. «Изучаем PHP и MySQL» [1].** Книга является визуальным руководством. Она демонстрирует теоретические материалы, практические работы. В книге описаны концепции программирования на

PHP и под MySQL: верификация форм, работа с сессиями, запросы к базе данных, операции с файлами и многое другое.

**Котеров Д., Симдянов И. «PHP 7» [6].** Рассмотрены основы языка PHP и его рабочего окружения в Windows, Mac OS X и Linux. Отражены радикальные изменения в языке PHP, произошедшие с момента выхода предыдущего издания: трейты, пространство имен, анонимные функции, замыкания, элементы строгой типизации, генераторы, встроенный веб-сервер и многие другие возможности. Приведено описание синтаксиса PHP 7, а также функций для работы с массивами, файлами, системой управления базами данных (СУБД) MySQL, memcached, регулярными выражениями, графическими примитивами, почтой, сессиями и т. д.

Особое внимание уделено рабочему окружению: сборке PHP-FPM и веб-сервера nginx, СУБД MySQL, протоколу SSH, виртуальным машинам VirtualBox и менеджеру виртуальных машин Vagrant. Рассмотрены современные подходы к веб-разработке, система контроля версий Git, GitHub и другие бесплатные Git-хостинги, новая система распространения программных библиотек и их разработки, сборка веб-приложений менеджером Composer, стандарты PSR и другие инструменты, и приемы работы современного PHP-сообщества.

**Гутманс Э., Баккен С., Ретанс Д. «PHP 5. Профессиональное программирование» [3].** Авторы делятся своими уникальными знаниями и предлагают множество реалистичных примеров. Описываются решения для разработки собственных веб-приложений, расширений, пакетов и сценариев командной строки. Рассматриваются: синтаксис и расширенные объектно-ориентированные возможности PHP — от свойств и методов до полиморфизма, интерфейсов и рефлексии; обработка ошибок; написание мощных веб-приложений (обработка ввода пользователя, работа с cookies, сессии); библиотека PEAR; интеграция с MySQL, SQLite и другими СУБД.

**Мазуркевич А., Еловой Д. «МВ PHP: настольная книга программиста» [9].** Книга описывает историю создания языка PHP и рассматривают-

ся основные аспекты. Также описаны классы и объекты, представляющие наибольшую сложность в изучении языка. Для начинающих программистов здесь есть параграфы о базовых типах данных и написании выражений.

### **1.1.5 Анализ интернет-источников**

**НОУ Интуит — курс «Microsoft: Разработка приложений на базе WPF»** [12]. В курсе лекций рассматриваются теоретические и практические вопросы создания корпоративных информационных систем на основе технологий Microsoft: WPF, SQL Server 2008, Silverlight, службы WCF Data Services и модели «сущность-связь» — Entity Data Model (EDM). Приведены примеры программ, написанные на языке программирования C#.

Приводятся сведения о языке расширенной разметки XAML, базовые компоненты WPF, основы технологии Silverlight, назначение службы WCF Data Services и вопросы её создания, разработка клиентских приложений для Silverlight. Обсуждаются вопросы создания корпоративных приложений.

**Интенсивный онлайн-курс «Базовый HTML и CSS»** [5]. На сайте описываются теоретические материалы и практические задания по верстке страниц сайта.

**PHP.SU** [30]. Портал содержит множество статей по PHP, MySQL, Apache, PEAR, регулярным выражениям, XML, другим веб-технологиям.

**www.site-do.ru** [31]. Сайт содержит материалы по разным языкам веб-программирования. В нем мало теории, но достаточно материалов для закрепления практических знаний языков.

**php-start** [29]. Бесплатный курс программирования на PHP. Курс состоит из двух частей: теоретической и практической. Целью теоретического курса является изучение общих концепций функционирования веб-приложений, а также самых важных инструментов в арсенале разработчика. В ходе практического курса демонстрируется процесс создания реального проекта на примере простого интернет-магазина.



## **1.2 Педагогический анализ рассматриваемой предметной области**

### **1.2.1 Характеристика учебного заведения**

ФГАОУ ВО «Российский государственный профессионально-педагогический университет» (РГППУ) — это федеральное государственное автономное образовательное учреждение высшего образования (вуз Екатеринбург), в состав которого входят институты, колледж, филиал и представительства в различных городах Российской Федерации.

Университет является первым инженерно-педагогическим институтом в стране. Его уникальность заключается в инженерно-педагогической подготовке специалистов. Выпускники РГППУ могут работать преподавателями и инженерами.

### **1.2.2 Педагогический адрес**

Электронное учебное пособие «Разработка WPF приложений в C#» предназначено для студентов направления 09.03.02 Информационные системы и технологии профиля подготовки «Информационные технологии в медиаиндустрии» по дисциплине «Алгоритмические языки и системы программирования».

### **1.2.3 Расширенный педагогический адрес**

Продукт можно использовать для обучения студентов, обучающихся по специальностям, связанным с изучением языков программирования. ЭУП можно применять для обучения на тренингах или семинарах по программированию, а также использовать для самостоятельного обучения.

## 1.2.4 Анализ рабочей программы

Дисциплина «Алгоритмические языки и системы программирования» закладывают основу профессиональных знаний бакалавра по направлению подготовки 09.03.02 Информационные системы и технологии профиля подготовки «Информационные технологии в медиаиндустрии».

Целью освоения дисциплины «Алгоритмические языки и системы программирования» является формирование у будущего бакалавра способности использовать технологии разработки объектов профессиональной деятельности в областях, связанных с системами массовой информации, дизайном, медиаиндустрией, а также деятельностью предприятий различного профиля в условиях экономики информационного общества. Дисциплина закладывает основы профессионального подхода к программному решению практических задач.

### **Задачи** дисциплины:

- овладение приемами работы с современными пакетами прикладных программ, обеспечивающих широкие возможности обработки информации;
- формирование у студентов представления о возможностях использования средств вычислительной техники, технологий программирования при решении различного вида экономических, производственных и учебных задач;
- развитие навыков алгоритмического мышления, получение знаний и практических навыков по алгоритмизации, разработке, отладке и тестированию программ;
- формирование навыков реализации типовых алгоритмов и структур данных и их модификаций на выбранном рабочем языке программирования;
- развитие навыков самостоятельного использования технической литературы по вопросам разработки программного обеспечения;
- систематизация и расширение знаний приемов и методов программирования, подготовка к их осознанному использованию при решении раз-

личного вида прикладных задач;

- накопление практического опыта разработки программного обеспечения с использованием эффективных алгоритмов обработки информации.

Дисциплина относится к части Дисциплины по выбору вариативной части учебного плана и изучается в течении 3-го и 4-го семестров 2-го курса. Дисциплина базируется на компетенциях, сформированных при изучении курсов «Математика» и «Информатика».

В соответствии с Федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 09.03.02 Информационные системы и технологии (уровень бакалавриата), дисциплина «Алгоритмические языки и системы программирования» формирует у студентов следующие компетенции:

- способность использовать технологии разработки объектов профессиональной деятельности в областях: машиностроение, приборостроение, техника, образование, медицина, административное управление, юриспруденция, бизнес, предпринимательство, коммерция, менеджмент, банковские системы, безопасность информационных систем, управление технологическими процессами, механика, техническая физика, энергетика, ядерная энергетика, силовая электроника, металлургия, строительство, транспорт, железнодорожный транспорт, связь, телекоммуникации, управление инфокоммуникациями, почтовая связь, химическая промышленность, сельское хозяйство, текстильная и легкая промышленность, пищевая промышленность, медицинские и биотехнологии, горное дело, обеспечение безопасности подземных предприятий и производств, геология, нефтегазовая отрасль, геодезия и картография, геоинформационные системы, лесной комплекс, химико-лесной комплекс, экология, сфера сервиса, системы массовой информации, дизайн, медиа индустрия, а также предприятия различного профиля и все виды деятельности в условиях экономики информационного общества (ПК-17);

- способностью адаптировать приложения к изменяющимся условиям функционирования (ПК-32).

Общая трудоемкость дисциплины составляет 5 зачетных единиц, всего 180 часов, из них 68 аудиторных часов (16 часов лекций и 52 часа лабораторных занятий) и 112 часов самостоятельной работы.

По окончании курса студенты должны знать:

- технологию решения задач с помощью компьютера;
- основные характеристики типов данных (целочисленный, вещественный, булевский, символьный, строковый) и структур данных (массивы, динамические списки, очереди, стеки), используемых в выбранном языке программирования;
- возможности среды разработки и платформы Framework.Net.

По окончании курса студенты должны уметь:

- реализовать готовые алгоритмы на языке высокого уровня;
- реализовывать приложения с графическим интерфейсом, используя стандартные визуальные компоненты среды программирования;
- разрабатывать алгоритмы, используя изложенные в курсе общие схемы, методы и приемы построения алгоритмов, выбирая подходящие структуры данных для представления информационных объектов;
- настраивать среду разработки, подключать дополнительные компоненты, работать с файлами;
- описывать пользовательские структуры данных, задавая поля и методы, создавая динамические списки структур;
- применять методы классов для организации поиска и сортировки в списках и массивах;
- конструировать обработчики исключительных ситуаций с целью построения кода, устойчивого к различным видам ошибок.

По окончании курса студенты должны владеть:

- терминологией, методологией и инструментальными средствами предметной области;
- технологией отладки программного кода;

- технологией использования справочной системы для поиска необходимых методов встроенных классов при решении задач.

### **1.2.5 Характеристика и специфические особенности текущего состояния педагогического процесса**

Педагогический процесс — это организованная и целенаправленная деятельность людей (обучающихся и обучающихся, воспитателей и воспитанников) с целью формирования необходимых знаний, практических навыков и умений, морально-политических, психологических и физических качеств личности и группы.

Педагогический процесс неразрывно связан со всеми другими общественными процессами (экономическим, политическим, нравственным, культурным и др.). Его сущность, содержание и направленность зависят от состояния общества, реального взаимодействия производительных сил и производственных отношений.

Как субъект педагогического процесса педагог получает специальное педагогическое образование, осознает себя ответственным перед обществом за подготовку подрастающих поколений.

Педагог как объект педагогического процесса в результате непрерывного образования, самовоспитания, общения с учениками подвергается воспитательным воздействиям и стремится к самосовершенствованию, формирует свою педагогическую культуру.

Одно из направлений современного обучения — это использование компьютерных средств и телекоммуникационных технологий для получения новых знаний на всех ступенях обучения.

Сегодня в нашей стране происходит становление новой системы образования, ориентированной на интеграцию в мировое информационно-образовательное пространство. Этот процесс сопровождается заметными из-

менениями в организации процесса обучения, который должен соответствовать современным техническим возможностям.

Применение информационных средств обучения предоставляет хорошие технические возможности для реализации различных дидактических идей и принципов организации учебного процесса, наполняет деятельность учителя принципиально новым содержанием. Однако это не является показателем того, что компьютер, берущий на себя часть функции учителя, способен вытеснить преподавателя из процесса обучения вовсе. Напротив, применение информационных технологий призвано активизировать процесс преподавания, повысить интерес учащихся к изучаемой дисциплине и эффективность учебного процесса. Они позволяют достичь большей глубины понимания изучаемого материала.

Одним из элементов и средств информационного обучения являются электронные учебные пособия.

К преимуществам электронного учебного пособия относится то, что оно позволяет охватить 100% количественный состав учащихся, повышает качество обучения, уменьшает время на поиск и оформление литературы.

В частности, разработанное электронное учебное пособие содержит понятный интерфейс, материалы для проведения лекций и контроля, лабораторные работы для закрепления изучаемых материалов на практике.

У преподавателя появляется больше времени на индивидуальную работу с каждым студентом. Снизится потребность в издании методического материала на бумажных носителях.

В общем электронное учебное пособие, помимо экономии материальных средств и времени будет способствовать росту качества обучения студентов университета.

## **1.3 Требования к разрабатываемому педагогическому продукту и выбор средств для его реализации**

### **1.3.1 Понятие электронного учебного пособия**

Использование электронного учебного пособия в обучении позволяет преподавателю на этапе первичного взаимодействия активно включить учащихся в учебный процесс и, создавая внешние предпосылки для формирования мотивов учения при работе с ЭУП, поддержать интерес к изучаемой дисциплине. Внешняя мотивация учебно-познавательной деятельности обучаемого не характеризуется продолжительной устойчивостью. Задача преподавателя заключается в преобразовании внешних мотивов обучения, провоцируемых специфическими свойствами компьютера, во внутренние потребности учащегося, опирающиеся на содержательную составляющую ЭУП.

Основная задача электронного учебного пособия на этапе получения новых знаний заключается в привлечении в процесс обучения иных, нежели традиционный (учебник или книга), возможностей человеческого мозга, в частности, слуховой и эмоциональной памяти, с целью максимального облегчения понимания и запоминания наиболее существенных понятий, утверждений и примеров. Основные фрагменты пособия или темы наряду с текстом и иллюстрациями содержат аудио или видеозапись лекторского изложения материала. Педагог дает свое понимание изучаемого предмета, расставляет необходимые смысловые акценты, которые трудно бывает передать в обычном учебнике. Текстовая часть сопровождается многочисленными перекрестными ссылками, позволяющими сократить время поиска необходимой информации, а также мощным поисковым центром и индексом.

Учебно-познавательные действия обучающихся при работе с ЭУП связаны с переформулированием учебного материала, его критическим осмыслением, поиском рационального способа принятия решения, сравнения и со-

поставления вариантов, реального проектирования, разбора нестандартных производственных ситуаций и других видов учебной деятельности.

Электронные учебные пособия создают базу для образования по-новому. Создав централизованный фонд, доступ к которому свободен по сети Интернет, и накапливая базу электронных учебных пособий, можно будет перейти к систематическому компьютерному образованию по любой форме дистанционного образования. Тогда каждый преподаватель сможет произвести выбор одного или несколько мультимедийных учебных пособий, которые он желал бы использовать в своей работе.

Технические возможности персонального компьютера, если компьютер используется как обучающее средство, позволяют: активизировать учебный процесс, индивидуализировать обучение, повысить наглядность учебного материала, сочетать теоретические знания с закреплением практических навыков, повысить и поддерживать интерес учащихся к обучению.

Электронные учебные пособия применимы и для стандартной формы обучения, как прекрасный иллюстративный материал. В связи с этим, все меньше остается преподавателей, отрицающих пользу новых информационных технологий. Образование меняется и качественно. Степень восприятия теперь определяется не только качеством электронного учебника, но и способностью обучающегося учиться.

Исключительно высокая степень наглядности представленного материала в электронных учебных пособиях, взаимосвязь различных компонентов, комплексность и интерактивность делают программы незаменимыми помощниками, как для обучающихся, так и для обучающихся.

При создании электронных учебных пособий используется современная компьютерная информационная технология — мультимедиа, позволяющая объединить в компьютерной системе текст, звук, видеоизображение, графическое изображение и анимацию. Комплексные занятия с привлечением аудиовизуальных материалов, представленных на компьютере, создают условия для расширения диапазонов видов образовательной деятельности



обучающих, стимулируют их способности к образованию и самообразованию.

### **1.3.2 Требования к электронным учебным пособиям**

Электронное учебное пособие — программно-методический обучающий комплекс, соответствующий типовой учебной программе и обеспечивающий возможность школьнику или студенту самостоятельно, или с помощью преподавателя освоить учебный курс или его раздел [25].

ЭУП создаются со встроенной структурой, словарями, справочными материалами, возможностью поиска. И при грамотном использовании могут стать мощным инструментом в изучении большинства дисциплин.

Электронные учебные пособия должны удовлетворять следующим качествам:

- возможность применять простой и удобный механизм навигации в пределах ЭУП;
- возможность использовать развитую поисковую систему в пределах ЭУП, в частности, при использовании гипертекстового формата издания;
- возможность создать автоматическую проверку уровня знаний студента;
- возможность специального варианта структуризации материала;
- возможность включить в состав ЭУП интерактивные фрагменты для обеспечения оперативного диалога с обучающимся.

Внедрение в ЭУП элементов мультимедиа позволяет осуществлять одновременную передачу различных видов информации. Сочетание звука, графики, анимации и видео позволяет улучшить восприятие нового материала.

Таким образом, электронные учебные пособия имеют большую практическую ценность и с их помощью можно не только сообщать фактическую информацию, снабженную иллюстративным материалом, но и наглядно демонстрировать те или иные процессы, которые невозможно показать при ис-

пользовании стандартных методов обучения. Кроме того, обучающийся может воспользоваться ЭУП самостоятельно, без помощи преподавателя или руководителя, находя ответы на интересующие его вопросы. Также важное значение ЭУП состоит в том, что преподаватель может быстро дополнять и изменять текстовый или иллюстративный материал при возникновении такой необходимости.

При разработке интерфейса следует принимать во внимание две группы требований:

- определяемые существующими стандартами в области создания интерактивных приложений;
- определяемые, психофизиологическими особенностями человека.

Основным назначением ЭУП для обучающихся является систематизация знаний, полученных обучающимися при изучении дисциплины.

ЭУП должно содержать в себе полную систему упражнений и текстовых заданий по основным разделам, а также обширный перечень итоговых тематических контрольных работ и тестов по всем разделам.

### **1.3.3 Анализ и выбор средств для реализации продукта**

Электронное учебное пособие «Разработка WPF приложений в C#» создавалось по средствам следующих технологий:

- HTML;
- CSS;
- PHP;
- iSpring Suite;
- JavaScript.

HTML (от англ. HyperText Markup Language — «язык гипертекстовой разметки») — стандартизированный язык разметки документов в сети Интернет. Большинство веб-страниц содержат описание разметки на языке HTML (или XHTML). Язык HTML интерпретируется браузерами; получен-

ный в результате интерпретации форматированный текст отображается на экране монитора компьютера или мобильного устройства [23].

Текстовые документы, содержащие разметку на языке HTML (такие документы традиционно имеют расширение .html или .htm), обрабатываются специальными приложениями, которые отображают документ в его форматированном виде. Такие приложения, называемые «браузерами» или «интернет-обозревателями», обычно предоставляют пользователю удобный интерфейс для запроса веб-страниц, их просмотра (и вывода на иные внешние устройства) и, при необходимости, отправки введённых пользователем данных на сервер. Наиболее популярными на сегодняшний день браузерами являются Google Chrome, Mozilla Firefox, Opera, Internet Explorer и Safari.

К достоинствам HTML можно отнести следующие аспекты:

- HTML страницы имеют значительно меньший вес;
- страницы, сверстанные на HTML работают и загружаются намного быстрее, расходуя меньше ресурсов сервера;
- для просмотра контента достаточно только браузера;
- универсальность и гибкость настройки под любое устройство (смартфон, компьютер, планшет и любые другие).

Недостатки HTML:

- HTML имеет фиксированный набор тэгов. Нельзя создавать свои тэги, понятные другим пользователям;
- это исключительно технология представления данных. HTML не несет информации о значении содержания, заключенного в тэгах;
- значимость тэгов в нем не определена, поэтому с его помощью нельзя описать иерархию данных.

В дополнение к этому языку, для стилизации элементов интерфейса (шрифты, цвета, отступы и др.) использовался язык CSS.

CSS (англ. Cascading Style Sheets — каскадные таблицы стилей) — формальный язык описания внешнего вида документа, написанного с использованием языка разметки [22].

Основной целью разработки языка CSS являлось разделение описания логической структуры веб-страницы (которое производится с помощью HTML или других языков разметки) от описания внешнего вида этой веб-страницы (которое теперь производится с помощью формального языка CSS). Такое разделение может увеличить доступность документа, предоставить большую гибкость и возможность управления его представлением, а также уменьшить сложность и повторяемость в структурном содержимом. Кроме того, CSS позволяет представлять один и тот же документ в различных стилях или методах вывода, таких как экранное представление, печатное представление, чтение голосом (специальным голосовым браузером или программой чтения с экрана), или при выводе устройствами, использующими шрифт Брайля.

К положительным качествам языка можно отнести следующие:

- простота самого языка CSS вместе с принципом отделения оформления от содержания дает возможность сократить время на разработку и поддержку сайта;
- уменьшается время загрузки страниц веб-сайта за счет переноса правил представления данных в отдельный CSS-файл. Благодаря этому браузер загружает только структуру документа, а также данные, хранимые на странице, а представление этих данных загружается браузером только один раз и может быть закеширован, благодаря этому уменьшается трафик, время загрузки, а также нагрузка на сервер;
- простота изменения дизайна. Один CSS управляет отображением множества HTML-страниц. Когда возникает необходимость изменить дизайн сайта, то незачем править каждую страницу. Для последующего изменения дизайна всего лишь нужно изменить CSS-файл, и как результат, изменение дизайна делается быстрее;
- повышение совместимости с разными платформами за счет использования веб-стандартов.

Единственным недостатком CSS является сложность настройки стилей для разных браузеров. Большинство из них воспринимают синтаксис по-разному.

PHP (англ. PHP: Hypertext Preprocessor — «PHP: препроцессор гипертекста»; первоначально Personal Home Page Tools — «Инструменты для создания персональных веб-страниц») — скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических веб-сайтов [28].

В области веб-программирования, в частности серверной части, PHP — один из популярных сценарных языков (наряду с JSP (Java Server Pages), Perl и языками, используемыми в ASP.NET).

Популярность в области построения веб-сайтов определяется наличием большого набора встроенных средств для разработки веб-приложений. Основные из них:

- автоматическое извлечение POST (Power-On Self-Test) и GET-параметров, а также переменных окружения веб-сервера в предопределённые массивы;
- взаимодействие с большим количеством различных систем управления базами данных (MySQL, MySQLi, SQLite, PostgreSQL, Oracle, Microsoft SQL Server и множество других);
- автоматизированная отправка HTTP-заголовков;
- работа с HTTP-авторизацией;
- работа с cookies и сессиями;
- работа с локальными и удалёнными файлами, сокетами;
- обработка файлов, загружаемых на сервер;
- работа с XForms.

Выбором для разработки данного языка служили такие положительные стороны как:

- простота в изучении;
- эффективность;
- гибкость.

Недостатки:

- глобальные параметры конфигурации влияют на базовый синтаксис языка, что затрудняет настройку сервера и разворачивание приложений;
- объекты передаются по значению, что смущает многих программистов, привыкших к передаче объектов по ссылке, как это делается в большинстве других языков;
- веб-приложения, написанные на PHP, зачастую имеют проблемы с безопасностью.

Отсутствие строгой типизации, такой как в Java или C++, делает этот язык простым в изучении. Например, при создании переменной совершенно необязательно указывать её тип или беспокоиться об её усечении или переполнении.

Благодаря своему «движку», сценарии в PHP выполняются с большой скоростью, что позволяет создавать на PHP серьезные веб-приложения.

Так как PHP является встраиваемым языком, это дает чрезвычайную гибкость в процессе разработки. Чаще всего сценарии PHP интегрируются в HTML страницы, но при необходимости могут встраиваться и в JavaScript, WML, XML и другие языки.

iSpring Suite — это инструмент для быстрого обучения, созданный iSpring для разработки профессиональных курсов электронного обучения в PowerPoint со встроенными (или автономными) викторинами, опросами и взаимодействиями. Программа совместима с основными стандартами LMS (SCORM / AICC), а также есть возможность публиковать курсы специально для BlackBoard LMS. Поддерживается также новый формат xAPI (Tin Can) для LRS. Экспорт может быть сплошным SWF-файлом, EXE-файлом, HTML-

страницей или ZIP-архив. Кроме того, формат межплатформенного вывода (Flash + HTML5) позволяет просматривать публикуемый контент на любом современном устройстве.

История iSpring Suite восходит к 2005 году, когда команда выпустила Flashspring Pro, прототип всех продуктов iSpring. Это простой конвертер PowerPoint-to-Flash с бесплатными и коммерческими версиями. Позже iSpring выпустила совместимость с Flashspring Ultra и SCORM. Сопутствующие продукты FlashSpring Lite и FlashSpring Server были выпущены в 2006 и 2007 годах соответственно. В 2008 году FlashSpring был ребрендирован и получил новое имя - iSpring. 2009 год стал годом, когда компания представила полностью обновленный инструмент электронного обучения под названием iSpring Presenter. Этот инструмент нашел признание рынка в связи с высоким качеством подготовленных курсов по разумной цене. В 2010 году iSpring Presenter заняла второе место в конкурсе Brandon Hall Excellence Technology Awards в категории Rapid Authoring. В 2011 году продукты iSpring впервые были представлены на европейских выставках. В том же году iSpring Suite был выпущен в качестве преемника iSpring Presenter.

В 2012 году Джо Ганчи из eLearningJoe опубликовал обзор iSpring Suite в журнале Learning Solutions Magazine. В мае 2015 года Ganci провела вебинар для сети Training Magazine Network, сравнивая iSpring Suite наряду с пятью другими ведущими инструментами разработки электронного обучения на базе PowerPoint. Джо Ганчи опубликовал окончательные результаты своих исследований в статье для журнала Learning Solutions Magazine.

Текущая версия iSpring Suite, iSpring Suite 8.0, вышла в октябре 2015 года.

JavaScript — мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Является реализацией языка ECMAScript (стандарт ECMA-262) [27].

Если HTML определяет структуру и контент веб-страницы, а CSS задает формат и внешний вид, то JavaScript добавляет интерактивность и создает богатые веб-приложения.

Однако, под общим термином «JavaScript», в контексте веб-браузера, понимаются несколько очень разных вещей. Одна из них — это базовый язык программирования (ECMAScript), а другая — коллекция Web APIs, включая DOM (объектно-ориентированная модель документа).

Преимуществами JavaScript являются:

- JavaScript предоставляет большое количество возможностей для решения самых разнообразных задач. Гибкость языка позволяет использовать множество шаблонов программирования применительно к конкретным условиям;
- популярность JavaScript открывает перед программистом немалое количество готовых библиотек, которые позволяют значительно упростить написание кода и нивелировать несовершенства синтаксиса;
- применение во многих областях. Широкие возможности JavaScript дают программистам шанс попробовать себя в качестве разработчика самых разнообразных приложений, а это, безусловно, подогревает интерес к профессиональной деятельности.

Недостатки:

- необходимость обеспечивать кроссбраузерность. Код должен корректно выполняться во всех, или хотя бы самых популярных, браузерах;
- система наследования в языке вызывает трудности в понимании происходящего. В JavaScript реализовано наследование, основанное на прототипах. В большинстве языков «класс потомок наследует родительский класс». Но в JavaScript такими вещами занимаются непосредственно объекты.



## 2 ОПИСАНИЕ ЭЛЕКТРОННОГО УЧЕБНОГО ПОСОБИЯ

### 2.1 Описание электронного учебного пособия

#### 2.1.1 Структура электронного учебного пособия

Цель: Обучение студентов навыкам построения графических интерфейсов при помощи технологии WPF.

Структурная схема продукта представлена ниже (рисунок 1).

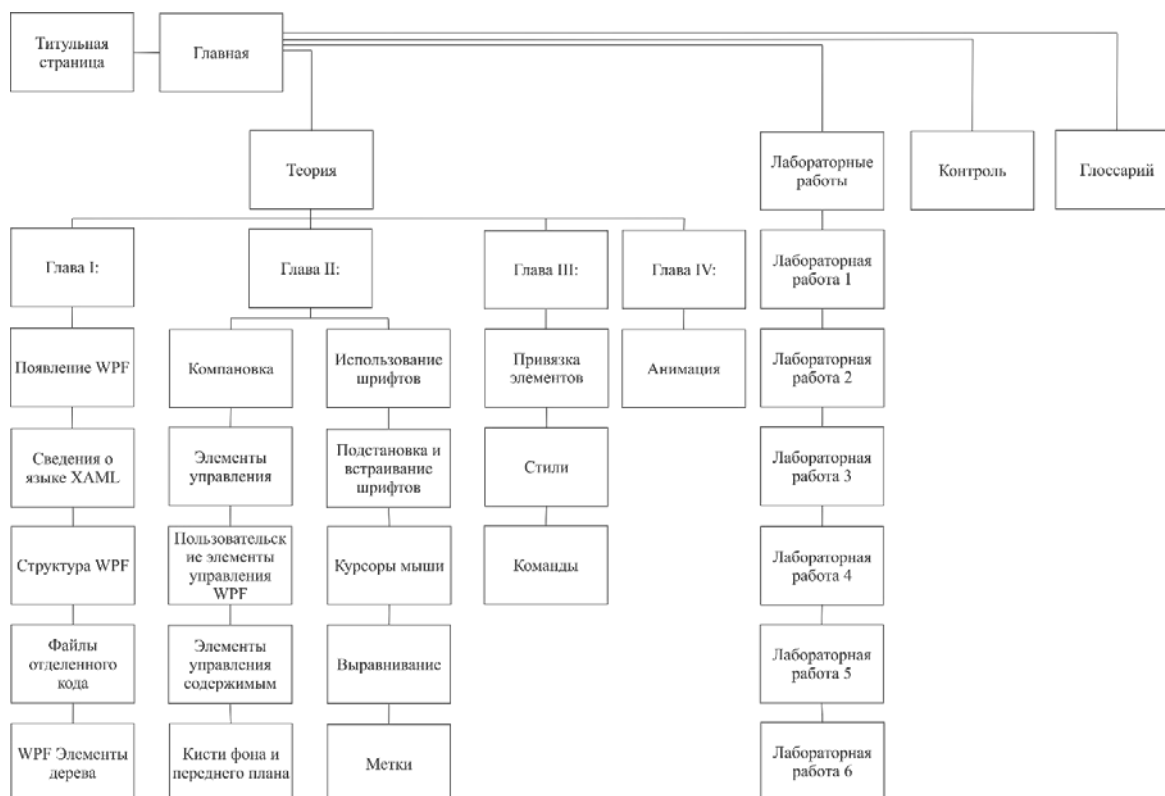


Рисунок 1 — Структурная схема электронного учебного пособия

#### 2.1.2 Описание интерфейса электронного учебного пособия

Для запуска электронного учебного пособия нужно открыть браузер (Mozilla Firefox, Google Chrome, Opera или Microsoft edge), установленный в системе, и ввести в адресной строке адрес сервера, на котором расположено

ЭУП, и нажать клавишу «Enter». Перед вами откроется стартовая страница (рисунок 2), окрашенная в голубой цвет. На ней представляется краткая информация об университете, его логотип, название электронного учебного пособия, информация об исполнителе и о руководителе проекта.



Рисунок 2 — Стартовая страница электронного учебного пособия

Нажав кнопку «Начать обучение» открывается главная страница пособия.

Макет электронного учебного пособия разработан при помощи табличной верстки. Структура верстки (рисунок 3) содержит следующие блоки:

- шапка (логотип и заголовок);
- навигационная цепочка;

- главное меню;
- поле контента;
- подвал.

Макет статичен, что положительно сказывается на отображении материала при разных размерах экрана.

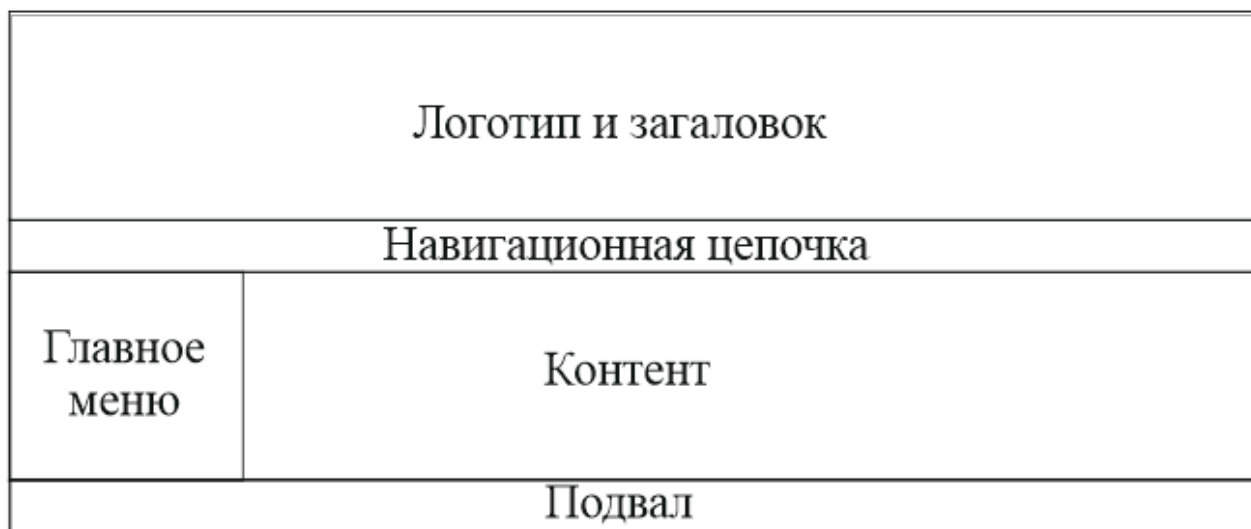


Рисунок 3 — Блочная композиция электронного учебного пособия

Для того чтобы студенты могли лучше сосредоточиться на изучаемом материале, интерфейс содержит более строгий стиль. Нет широкого цветового контраста, что в свою очередь отвлекало бы внимание и утомляло пользователя.

Достоинствами электронного учебного пособия является простая и функциональная навигация по разделам. Слова в тексте, требующие разъяснения, содержат комментарии. В конце каждой лабораторной работы находится программа, демонстрирующая результат выполненной работы. Для контроля знаний разработан тест из 31 вопроса. Для избегания повторения вопросов, в случайном порядке выходит только 25. По окончании тестирования демонстрируются на экране и отправляются на электронную почту преподавателю. Реализовано поле «поиск» в глоссарии. Напечатав в нем термин, результатом отображается только он, а остальные термины скрываются.

Главное меню располагается слева (рисунок 4). Блоки без желтых галочек являются гиперссылками на последующие страницы. Остальные блоки с

галочками обозначают, что этот элемент меню раздвигается и имеет подразделы.

При наведении курсором мыши тема выделяется темно-синим цветом.

Меню содержит следующие основные разделы:

1. «Главная».
2. «Теория»:
  - глава I;
  - глава II;
  - глава III;
  - глава IV.
3. «Лабораторные работы».
4. «Контроль».
5. «Глоссарий».

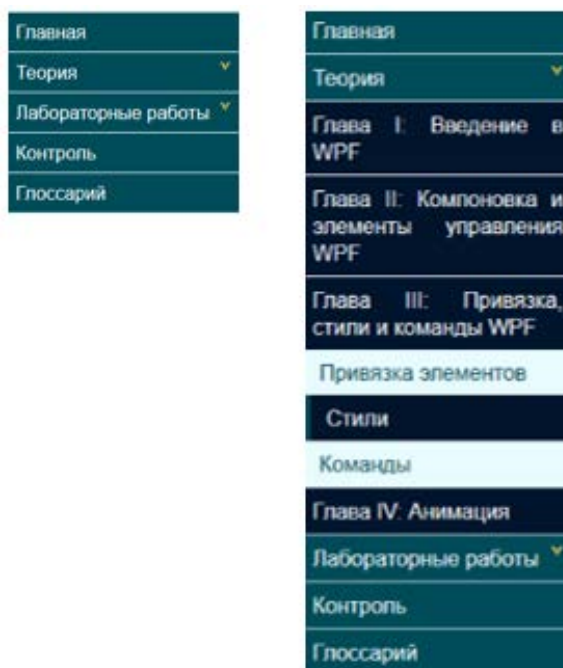


Рисунок 4 — Главное меню в свернутом и раскрытом виде

При переходе в какой-то раздел электронного учебного пособия, в навигационной цепочке появляется путь к нему (рисунок 5), тем обеспечивая дополнительное удобство навигации по электронному учебному пособию.



Рисунок 5 — Навигационная цепочка

Каждое «звено» цепи является гиперссылкой, направленной в содержание главы или раздела (рисунок 6).

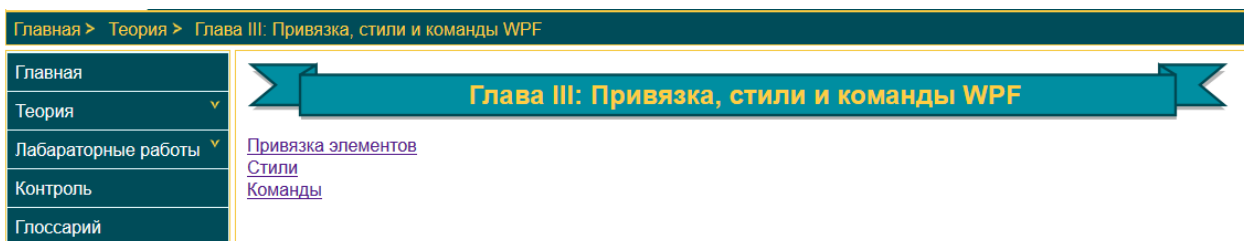


Рисунок 6 — Содержание третьей главы

На каждой странице пособия располагается кнопка анимированная «Вверх» (рисунок 7). Она поднимает скролл страницы на самый верх, что в свою очередь создает дополнительный комфорт работы с электронным учебным пособием.



Рисунок 7 — Кнопка «Вверх»

### 2.1.3 Главная страница

В разделе «Главная» представлена краткая информация о пособии, рекомендации для студентов и преподавателей. Также описана актуальность изучаемой технологии.

На странице (рисунок 8) содержится слайдер, 3 выдвигающихся меню и введение.

Слайдер выполняет эстетическую роль. Демонстрирует, какие возможно создавать проекты по средствам, изучаемой технологии, тем самым мотивируя студента. Так же подчеркивает положительные качества Windows Presentation Foundation (WPF).

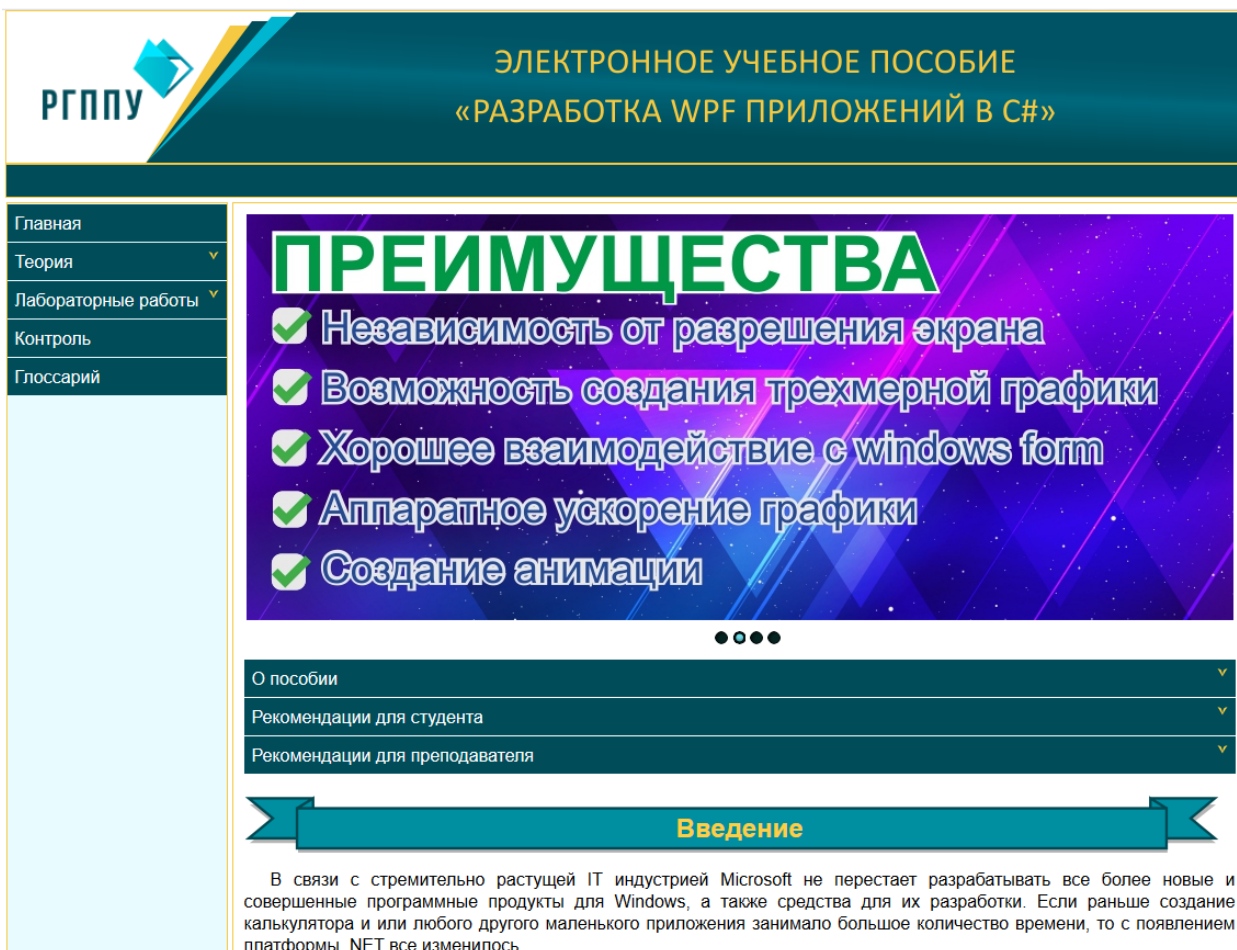


Рисунок 8 — Главная страница

Выдвигающиеся меню содержат основную информацию об электронном учебном пособии, рекомендации по использованию его студентом и преподавателем. Для просмотра информации в одном из 3 меню, нажмите по нему.

Введение описывает актуальность Windows Presentation Foundation (WPF). Описывается общее представление языка и его возможность взаимодействия с другими языками программирования.

#### 2.1.4 Теоретический блок

Теоретический раздел предназначен для формирования знаний о языке XAML и работе с технологией WPF. Блок включает в себя четыре главы:

1. Основа WPF.
2. Компоновка и элементы управления WPF.

3. Привязка, стили и команды WPF.

4. Анимация.

Первая глава описывает историю появления технологии, основные сведения, о применяемом языке XAML. Так же описана структура WPF и принципы ее работы.

В разделе «Компоновка и элементы управления WPF» описываются всевозможные элементы пользовательского интерфейса, такие как кисти фона, курсоры мыши, кнопки, блоки для вписывания текста и д. р.

В 3 главе описываются способы создания связей между объектами и их взаимодействие друг с другом. Рассказаны способы стилизации объектов. Также представлено взаимодействие с командами C#.

Последняя глава посвящается анимации, ее видам и работе с ней.

На страницах с теоретическим материалом содержатся таблицы (рисунок 9), фрагменты кода (рисунок 10), рисунки, поясняющие комментарии (рисунок 11), основные заголовки темы (рисунок 12) и маркированные списки (рисунок 13).

Ключевое слово	Назначение
x:Array	Представляет тип массива .NET на XAML
x:ClassModifier	Позволяет определять видимость типа класс (internal или public), обозначенного ключевым словом Class
x:FieldModifier	Позволяет определять видимость члена типа (internal, public, private или protected) для любого именованного элемента корня. Именованный элемент определяется с использованием ключевого слова Name
x:Key	Позволяет установить значение ключа для элемента XAML, которое должно быть помещено в элемент словаря
x:Name	Позволяет указывать сгенерированное C# имя заданного элемента XAML
x:Null	Представляет null-ссылку
x:Static	Позволяет ссылаться на статический член типа
x:Type	XAML-эквивалент операции C# typeof (вызывает System.Type на основе указанного имени)
x:TypeArgument	Позволяет устанавливать элемент как обобщенный тип с определенными параметрами

Таблица 1. Ключевые слова XAML

Рисунок 9 — Таблица ключевых слов XAML

```
<Window x:Class="MyFirstWpfProject.MainWindow"
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
Title="MainWindow" Height="350" Width="525">
  <Grid>
  </Grid>
</Window>
```

Рисунок 10 — Фрагмент кода XAML

## ИХ **дескрипторов** со специальным

е пре  
эмен  
бъек  
ущий

Лексическая единица (слово, словосочетание) информационно-поискового языка, служащая для описания основного смыслового содержания документа или формулировки запроса при поиске документа (информации) в информационно-поисковой системе. Дескриптор однозначно ставится в соответствие группе ключевых слов естественного языка, отобранных из текста, относящегося к определённой области знаний.

Рисунок 11 — Поясняющий комментарий к слову «дескриптор»

## Структура WPF

Рисунок 12 — Заголовок темы «Структура WPF»

При наведении курсора изображение увеличивается. Названия рисунков выделены жирным шрифтом.

Маркированные списки отмечаются цветным ромбиком.

- ◆ **Логическая структура дерева;**
- ◆ **Визуальная структура дерева.**

Рисунок 13 — Маркированный список

Каждая тема в содержит в конце перечень контрольных вопросов для самоконтроля (рисунок 14).

### **Вопросы для самоконтроля.**

1. Назовите интерфейс, представляющий собой набор подпрограмм определений, протоколов и инструментов для создания прикладного программного обеспечения.
2. Какие преимущества и отличия имеет Windows Presentation Foundation перед Windows Forms?
3. За счет чего графические приложения WPF меньше влияют на производительность системы?

Рисунок 14 — Вопросы для самоконтроля

## **2.1.5 Практический блок**

Лабораторные работы служат для закрепления теоретических знаний на практике. Данный блок содержит 6 лабораторных работ.



В первой лабораторной работе студенты знакомятся с основной структурой и созданием проекта. Так же их задача в рамках этой работы разработать калькулятор.

Во второй лабораторной работе на основе разработанного калькулятора, изменяется его внешний вид (шрифт, цвета, форма кнопок) по средствам языка XAML.

Третья лабораторная работа содержит материал о создании стилей для объектов. Строятся несколько фигур и при помощи стилей задаются им цвета, тени, заливка, градиентная заливка и ориентация на рабочей области.

В четвертой работе были рассматриваются основные элементы управления, такие как: Menu, ToolBar, TabControl, ToolTip. Помимо создания самих объектов, настраивается их взаимодействие друг с другом.

В пятой лабораторной работе рассматриваются панели динамического размещения интерфейсных элементов.

В последней лабораторной работе рассмотрены два вида анимации. Линейная и анимация по шаблону. Работа выполняется на основе 3 лабораторной работы. Созданным окружностям задается анимация при нажатии на кнопку, так же добавляется еще несколько объектов, демонстрирующих работу анимации вдоль пути.

Лабораторные работы имеют аналогичные параметры оформления текста, что и теория. Работы сопровождаются рисунками. Элементы интерфейса или пункты, которым нужно следовать при выполнении задания, отмечены красным цветом. Заголовки, такие как цели, задания и др. выделяются жирным шрифтом с зеленым цветом. Разработаны программы (рисунок 15), демонстрирующие результат проделанной работы.

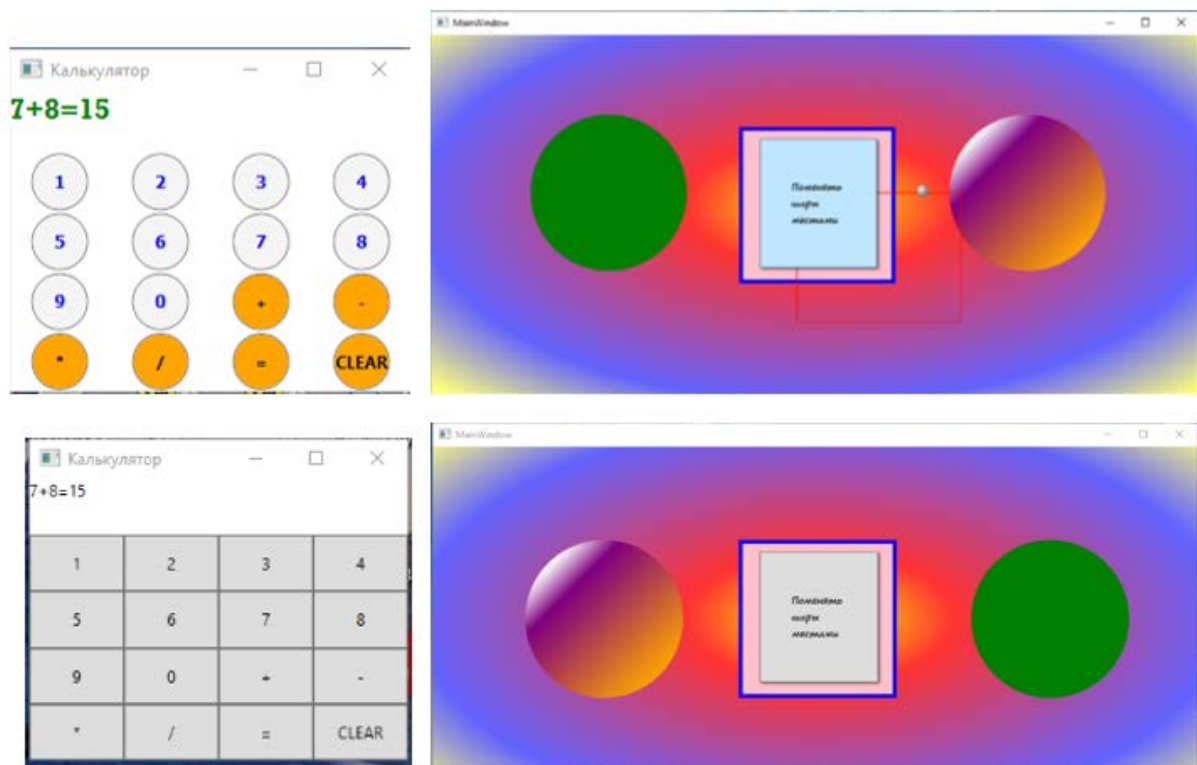


Рисунок 15 — Программы для демонстрации

Для примера рассмотрим процесс создания калькулятора на основе первой лабораторной работы.

Перед выполнением лабораторной работы студенты знакомятся с целью и заданием проекта (рисунок 16).

**Цель:**

Ознакомиться с основной структурой WPF проекта.

---

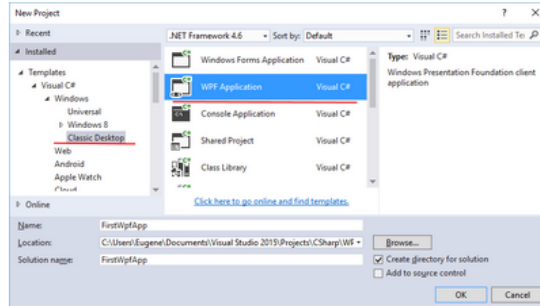
**Задание: создайте приложение "калькулятор".**

---

Рисунок 16 — Цель и задание проекта

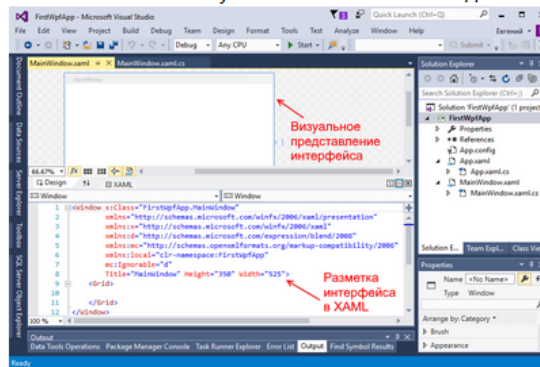
Начинается все с описания процесса создания проекта (рисунок 17). Демонстрируются скриншоты окна программы и пошагово расписывается процесс создания. Красным выделяются кнопки и разделы интерфейса программы.

Открыть Visual Studio и в меню **File (Файл)** выберите пункт **New (Создать) -> Project... (Проект...)**. Перед вами откроется диалоговое окно создания проекта, в котором мы выберем шаблон WPF Application:



**Рисунок 1. Окно создания проекта**

Укажите проекту какое-нибудь имя и нажмем кнопку ОК. И Visual Studio создаст новый проект



**Рисунок 2. Окно нового проекта**

Рисунок 17 — Описание процесса создания проекта

Следующим этапом студенты знакомятся с содержимым панели элементов управления (рисунок 18). Изучив элементы, нужно выбрать и переместить поля ввода и кнопки в окно проекта.

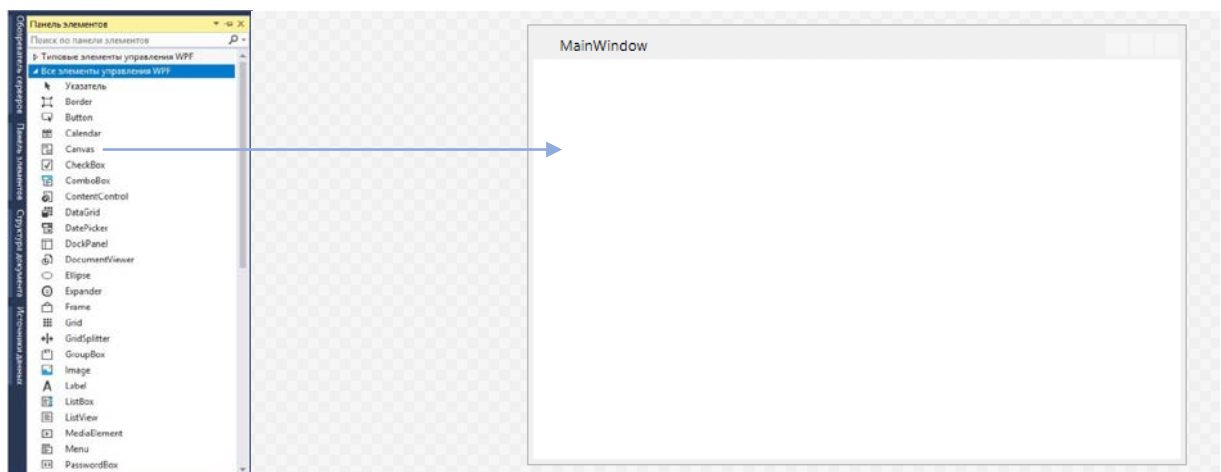


Рисунок 18 — Панель элементов управления

Добавив все необходимые элементы управления на форму, студенты проводят ряд настроек внешнего вида при помощи расширенного языка разметки XAML (рисунок 19).

```

<Window x:Class="FirstWpfApp.MainWindow"
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
Title="Калькулятор" Height="250" Width="300">
  <Grid x:Name="LayoutRoot" Background="White">
    <Grid.RowDefinitions>
      <RowDefinition />
      <RowDefinition />
      <RowDefinition />
    </Grid.RowDefinitions>
  </Grid>

```

Рисунок 19 — Фрагмент кода программы XAML

Полностью настроив параметры внешнего вида (отступы, размеры, распределение элементов по столбцам и строкам всех элементов управления), остается задать им необходимые действия. Для этого открывается файл App.xaml.cs, где студенты вписывают код С# (рисунок 20)

Здесь вы задали массив кнопок как в обычном калькуляторе. После изменения разметки, визуальное представление интерфейса также изменится, и там вы сможете разглядеть подобие калькулятора. Теперь откройте файл логики, привязанной к данной разметке, MainWindow.xaml.cs. Сейчас он имеет следующий код:

```

{
public partial class MainWindow : Window
{
public MainWindow()
{
InitializeComponent();
}
}
}

```

И измените его на следующий:

```

public partial class MainWindow : Window
{
string leftop = ""; // Левый операнд
string operation = ""; // Знак операции
string rightop = ""; // Правый операнд
public MainWindow()
{
InitializeComponent();
// Добавляем обработчик для всех кнопок на гриде
foreach (UIElement c in LavoutRoot.Children)

```

Рисунок 20 — Фрагменты кода С#

Работа выполнена. Студент запускает процесс компиляции программы и проверяет результат своей проделанной работы (рисунок 21).

Что бы удостовериться в правильности результата, необходимо в папке «Рабочая» открыть заранее подготовленную программу, соответствующую требованиям лабораторной работы, и сравнить со своей.

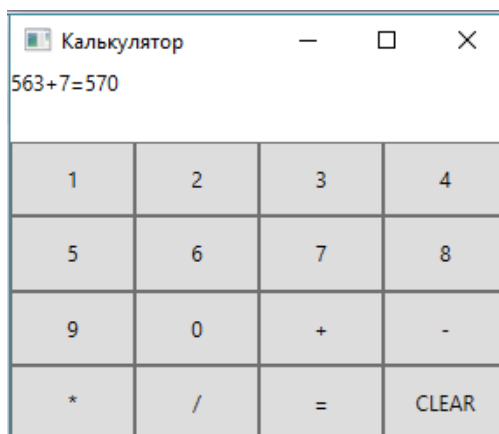


Рисунок 21 — Приложение «Калькулятор»

Если программа работает верно студент отвечает на вопросы для самоконтроля и демонстрирует работу преподавателю.

В такой последовательности выполняются остальные лабораторные работы.

### 2.1.6 Блок контроля

После того, как студент пройдет курс по разработке WPF приложений, ознакомится с теоретическими материалами и выполнит все 6 лабораторных работ можно перейти к итоговому контролю знаний. В нем содержится 31 тестовый вопрос. Из них в случайном порядке выходит 25. Для начала тестирования студент вводит свои данные, ФИО и номер группы. Вопросы составлены по типам: выбор одного верного ответа, несколько правильных ответов, установление соответствия и ввод ответа на вопрос самостоятельно. По окончании тестирования результаты каждого студента отправляются на электронную почту преподавателя.

Контроль не содержит элементов основного шаблона в целях концентрации внимания на вопросах. Тест начинается с приветствия (рисунок 22). Нажав на «начать тест» открывается окно регистрации студента (рисунок 23), где он указывает ФИО и номер группы.

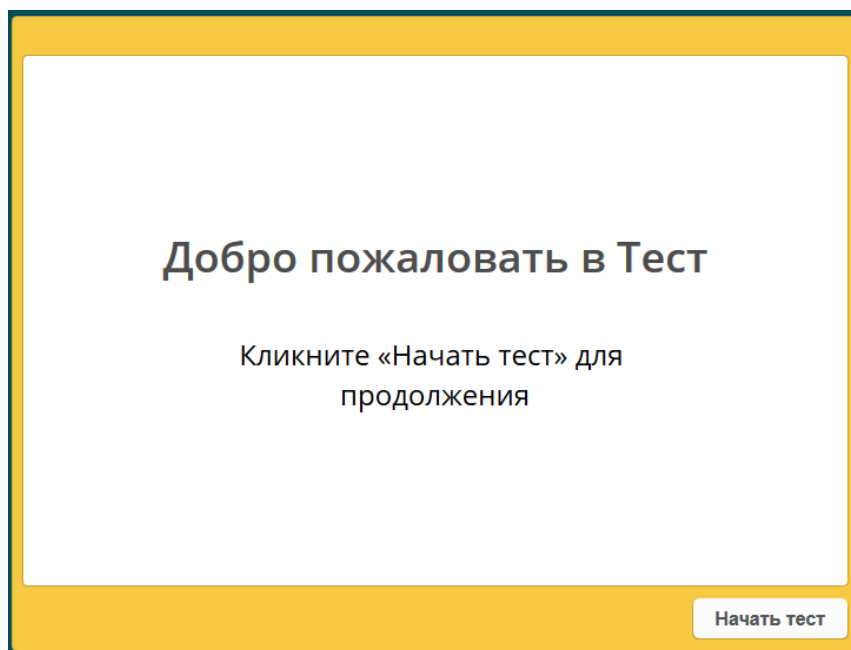


Рисунок 22 — Начало тестирования «Приветствие»

The image shows a rectangular window with a yellow border. Inside, the text is centered. At the top, it says "Введите ваши данные" in a bold, dark font. Below that, there are two input fields. The first is labeled "ФИО:\*" and contains the text "Меньшиков Анатолий Иванович". The second is labeled "Группа:\*" and contains the text "ЗКТ-401С". In the bottom right corner, there is a white button with the text "Отправить" in black.

Рисунок 23 — Ввод данных о студенте

Вопросы в тесте разделяются на следующие типы: заполнение пропуска в тексте, выбор одного правильного ответа, выбор нескольких правильных ответов, самостоятельная запись правильного ответа, установление соответствия. Все перечисленные варианты продемонстрированы далее (рисунок 24).

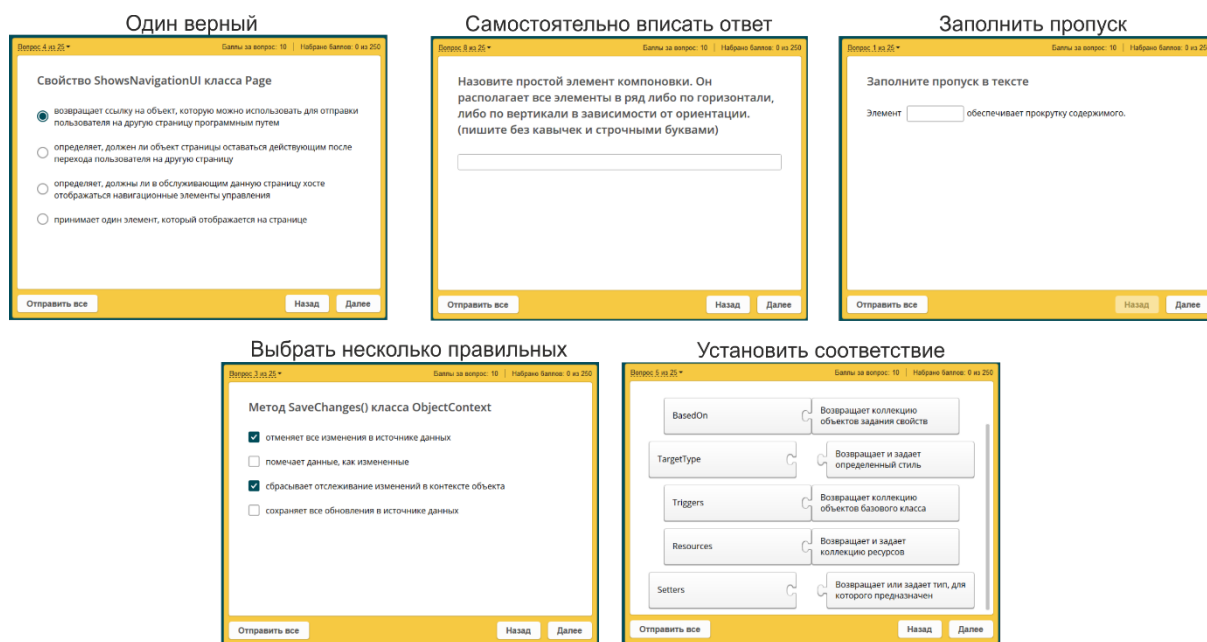


Рисунок 24 — Формы теста

После проведения теста результаты отображаются в процентах и количестве баллов (рисунок 25). Студент видит количество набранных баллов и количество необходимых баллов для получения положительной оценки.

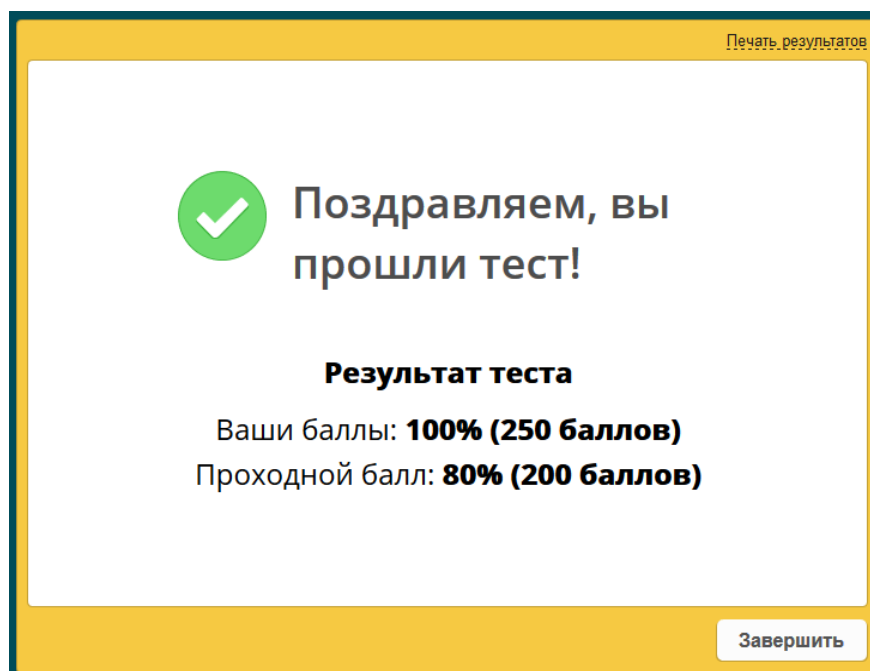


Рисунок 25 — Результаты теста

Для удобства преподавателя подробные результаты отправляются к нему на электронную почту (рисунок 26). Он видит ФИО и группу студента, который проходил тест. Отображается информация о количестве баллов,

набранных студентом, и результат, пройден тест или нет. Вопросы с верными и ошибочными ответами также присутствуют в отчете.

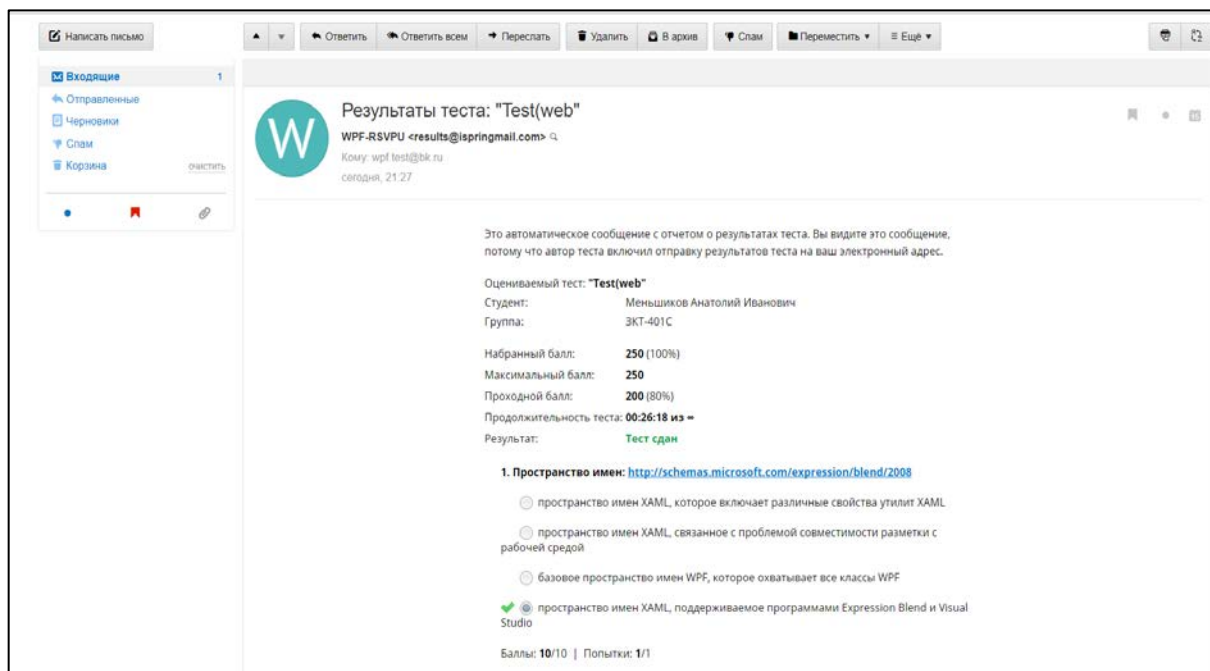


Рисунок 26 — Электронный ящик преподавателя с результатами теста

В итоге можно сказать, что контроль является неотъемлемой частью ЭУП и позволяет проводить объективную оценку знаний обучающихся.

### 2.1.7 Глоссарий

«Глоссарий» (рисунок 27) содержит список терминов и определений по теме «разработка WPF приложений в C#». Все термины расположены в алфавитном порядке. Шапка таблицы выделяется синим цветом.

Студент вписывает в строку поиска интересующий его термин. Становясь активным, поле ввода окрашивается в желтый цвет. JavaScript начинает обрабатывать, каждую введенную букву и сравнивать ее с буквами содержащимися в каждой строке таблицы, в результате поиска все остальные термины исчезают, оставляя только результат запроса (рисунок 28).





Рисунок 27 — Глоссарий

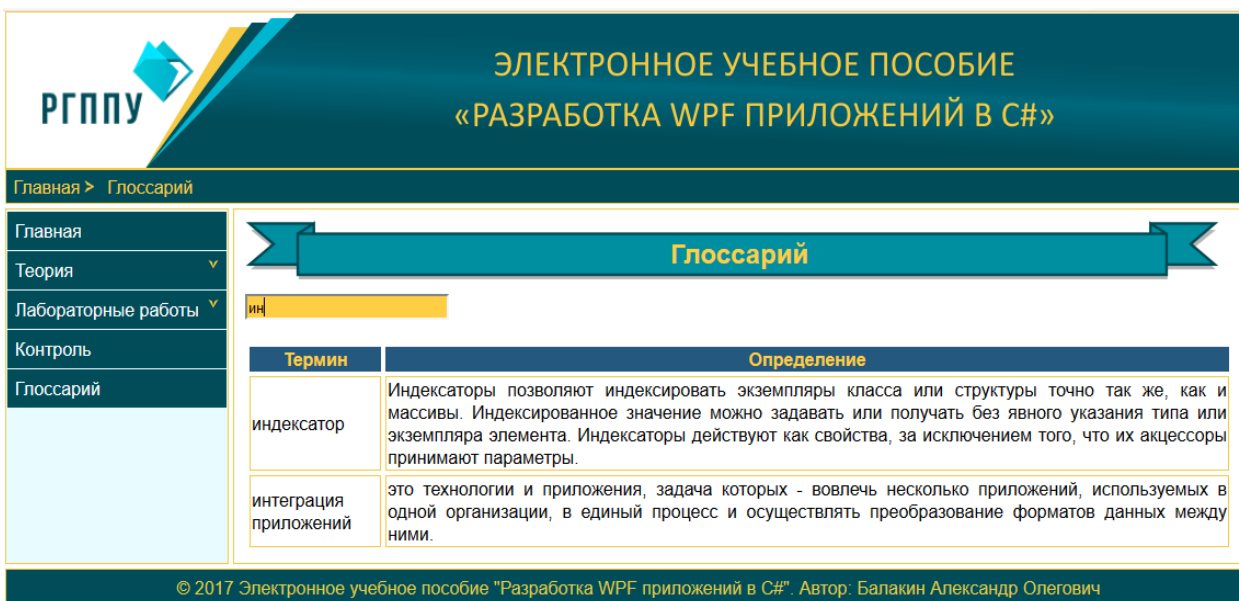


Рисунок 28 — Глоссарий с заполняемым полем поиска

## 2.2 Методические указания по использованию электронного учебного пособия в учебном процессе

Электронное учебное пособие рекомендуется использовать для аудиторных занятий, поскольку защита лабораторных работ подразумевает присутствие преподавателя. Помимо этого, деятельность преподавателя носит

консультативный характер (обучающиеся советуются с преподавателем по вопросам, вызвавшим затруднения).

Данное пособие может применяться для самостоятельного изучения технологии WPF.

Теоретические материалы, представленные в пособии, содержат необходимый спектр «научных знаний» по изучаемой технологии, их можно использовать при проведении лекционных занятий.

Лабораторные работы применяются при проведении практических занятий и организации самостоятельной работы обучающихся, поскольку эти задания способствуют закреплению полученных знаний на практике.

«Контроль» используется для проведения контрольных работ. Позволяет проводить объективную оценку знаний обучающихся.

«Глоссарий» содержит список основных терминов и определений. Тем самым помогая студенту в изучении теоретического материала.

Электронное учебное пособие имеет простой и понятный интерфейс. Даже самый «начинающий» пользователь, умеющий пользоваться интернет-браузером, с легкостью сможет разобраться с использованием данного продукта.

### **2.2.1 Инструкции по работе с электронным учебным пособием**

При работе с ЭУП, обучающемуся следует выполнять следующие инструкции:

1. Включить компьютер.
2. Открыть браузер (Google Chrome, Mozilla Firefox, Opera или любой другой).
3. Введите в адресной строке адрес ЭУП.
4. В открывшемся окне нажмите кнопку «Начать обучение».
5. Попав на главную страницу, навигацию по пособию осуществляйте по меню, расположенному слева.

6. Перейдя в любой из разделов, в шапке отобразится навигационная цепочка «хлебные крошки». Ей вы так же можете пользоваться, как средством навигации.

### **2.2.2 Инструкция по контролю знаний**

Для прохождения контроля студенту следует выполнить следующие инструкции:

1. На главной странице электронного учебного пособия в меню слева перейти в раздел «Контроль».
2. На странице приветствия нажать «Начать обучение».
3. Ввести свои данные (ФИО и название группы).
4. Пойти тест, ответив на все вопросы. Тест не будет считаться завершённым, если будут пропущены ответы.
5. Показать результаты преподавателю.

Для проверки итогов прохождения тестов преподавателю стоит выполнить следующие инструкции:

1. Открыть браузер (Google Chrome, Mozilla Firefox, Opera или любой другой).
2. Перейти на сайт электронной почты (Mail.ru).
3. Ввести данные для авторизации (логин — wpf.test@bk.ru, пароль — rgppu.1979.wpf).

В каждом письме содержится результат прохождения теста и информация о тестируемом.

## ЗАКЛЮЧЕНИЕ

Актуальность, выбранной темы изучения, связана с быстро развивающимся техническим прогрессом. Встает острая необходимость в подготовке компетентных кадров в области программирования. Вакансии в отрасли программной разработки, на рынке труда, хорошо оплачиваются. В связи с большим спросом на программную продукцию, необходимо в краткие сроки создавать качественные программные продукты.

Мотивом выбора предметной области WPF служило то, что данная система достаточно современная и более совершенна по сравнению со своим предшественником Windows Form. Поэтому имеет смысл изучать данную платформу и обучать других.

В первой главе рассмотрены основные характеристики предметной области, проведен анализ наиболее подходящей литературы и интернет источников для изучения, проведен педагогический анализ предметной области, проанализирована рабочая программа по дисциплине «Алгоритмические языки и системы программирования». Так же посмотрел пару учебно-методических разработок.

Проанализировав обе учебно-методические разработки, можно сделать вывод о том, в данных дисциплинах рассмотрены методы и способы создания компьютерной программы путём манипулирования графическими объектами. Также приводятся сведения о декларативном языке расширенной разметки для приложений XAML, основы технологии и базовые компоненты WPF, рассматриваются вопросы разработки приложений на базе WPF.

По средствам собранного материала был описан внешний вид электронного учебного пособия. Для более конструктивного понимания расположения контента разработана блок схема расположения страниц ЭУП.

Выбранная литература поможет более грамотно подойти к разработке электронного учебного пособия по выбранной теме. Несмотря на то, что данные источники не 2017 года, предлагаемые в них методы и примеры, подходят для решения поставленных задач. Интернет источники подходят в качестве справочного материала.

Результатом решения поставленных задач были следующие этапы. Подобрал тему для дипломного проекта. Актуализировал, выбранную тему. Сформировал цели и задачи проекта, связанные с обучением студентов технологии «WPF». Подобрана и проанализирована литература, освещающая технические аспекты разработки электронного учебного пособия по средствам PHP, JavaScript, HTML и CSS. Помимо технической составляющей подобраны источники, связанные непосредственно с теорией по технологии «WPF». Проанализировано несколько интернет источников.

Использование ЭУП дает возможность студенту сделать процесс обучения более результативным и обеспечить формирование и закрепление новых знаний, умений и навыков в определенной предметной области и в определенном объеме в индивидуальном режиме, либо при ограниченной по объему методической помощи преподавателя. Еще одним преимуществом является грамотное и удобное структурирование информации и быстрого передвижения по разделам, что убыстряет процесс изучения материала.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Бейли Л. Изучаем PHP и MySQL [Текст]: учебное пособие / Л. Бейли, М. Моррион — пер. с англ. Тунин К. — Москва: Эксмо, 2010. — 800 с.
2. Бибо Б. jQuery в действии [Текст] / Б. Бибо, И. Кац, А. де Роза. — пер. с англ. А. Тумаркин — Санкт-Петербург: Питер, 2017. — 528 с.
3. Гудманс Э. PHP 5. Профессиональное программирование [Текст] / Э. Гудманс, С. Баккен, Д. Ретанс. — пер. с англ. Киселев А. — Санкт-Петербург: Символ-Плюс, 2006. — 704 с.
4. Декетт Д. HTML и CSS. Разработка и дизайн веб-сайтов [Текст]: / Д. Декетт. — пер. с англ. Райтмана М. А. — Москва: Эксмо, 2013. — 480 с.
5. Интенсивный онлайн-курс «Базовый HTML и CSS» [Электронный ресурс]. — Режим доступа: <https://htmlacademy.ru/> (дата обращения: 04.10.2017).
6. Котеров Д. В. PHP 7 [Текст]: учебное пособие / Д. В Котеров, И. В. Симдянов. — под. ред. Кондукова Е. — Санкт-Петербург: БХВ-Петербург, 2016. — 1088 с.
7. Кулиненко И. М. Методика создания электронного учебника [Текст] / И. М. Кулиненко. — Челябинск: ЧМК, 2012 — 11 с.
8. Мак-дональд М. WPF 4: Windows Presentation Foundation в .NET 4.0 с примерами на C# 2010 для профессионалов [Текст] / М. Мак-дональд. — под. ред. Артеменко Ю. Н. — пер. с англ. Волковой Я. П., Моргунова А. А., Мухина Н. А. — Москва: ООО «И. Д. Вильямс», 2011. — 1024 с.
9. Мазуркевич А. MV PHP: настольная книга программиста [Текст] / А. Марзукевич, Д. Еловой. — Минск: Новое знание, 2003. — 480 с.
10. Макфарланд Д. С. Большая книга CSS3 [Текст] / Д. С. Макфарланд. — под. ред. Веницкий Д. — пер. с англ. Вильченский Н. — 3-е изд.

— Санкт-Петербург: Питер, 2014. — 608 с.

11. Натан А. WPF 4. Подробное руководство [Текст] / А. Натан. — пер. с англ. А. Слинкин. — Санкт-Петербург: Символ-Плюс, 2011. — 880 с.

12. НОУ Интуит — курс «Microsoft: Разработка приложений на базе WPF» [Электронный ресурс]. — Режим доступа: <http://www.intuit.ru/studies/courses/690/546/info> (дата обращения: 13.05.2017).

13. Петсольд Ч. Microsoft Windows Presentation Foundation [Текст] / Ч. Петсольд. — пер. с англ. Матвеев Е. А. — Москва: Русская Редакция; Санкт-Петербург: Питер, 2008. — 944 с.

14. Петюшкин А. В. HTML в Web-дизайне [Текст] / А. В. Петюшкин. — под. ред. Кондукова Е. — Санкт-Петербург: БХВ-Петербург, 2004. — 400 с.

15. Прохоренок Н. А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера [Текст] / Н. А. Прохоренок. — под. ред. Кондукова Е. — 3-е изд. — Санкт-Петербург: БХВ-Петербург, 2010. — 912 с.

16. Рабочая программа по дисциплине «Технология разработки пользовательских интерфейсов» [Электронный ресурс]. — Режим доступа: <http://docplayer.ru/65278006-Bazovaya-rabochaya-programma-discipliny-tehnologiya-razrabotki-polzovatel'skih-interfeysov-informatika-i-vychislitel'naya-tehnika.html> (дата обращения: 21.09.2017).

17. Садыков Р. А. Визуальное программирование [Текст]: рабочая программа / Р. А. Садыков. — Южноуральск: Южно-Уральский государственный университет, 2013 — 43 с.

18. Стиллмен Э. Изучаем C# [текст]: учебное пособие / Э. Стиллмен, Д. Грин. — 3-е изд. — Санкт-Петербург: Питер, 2014. — 816 с.

19. Флэнаган Д. JavaScript: карманный справочник [Текст]: справочник / Д. Флэнаган. — под. ред. Тригуб С. Н. — пер. с англ. Сысонок А. Г. — 3-е изд. — Москва: ООО «И. Д. Вильямс», 2013. — 320 с.

20. Хеник Б. HTML и CSS: путь к совершенству [Текст] / Б. Хеник.

— пер. с англ. ООО «Питер». — Санкт-Петербург: Питер, 2011. — 336 с.

21. Шмитт К. CSS. Рецепты программирования [Текст] / К. Шмитт. — пер. с англ. Коротяева Т. — Москва: Русская редакция; Санкт-Петербург: БХВ-Петербург, 2011. — 672 с.

22. CSS [Электронный ресурс]. — Режим доступа: <https://ru.wikipedia.org/wiki/CSS> (дата обращения: 15.11.2017).

23. HTML [Электронный ресурс]. — Режим доступа: <https://ru.wikipedia.org/wiki/HTML> (дата обращения: 15.11.2017).

24. HTML.NET [Электронный ресурс]. — Режим доступа: <http://ru.html.net> (дата обращения: 11.21.2017).

25. Infed.ru [Электронный ресурс]. — Режим доступа: <http://infed.ru/articles/87/> (дата обращения: 10.11.2017).

26. ISpring Suite [Электронный ресурс]. — Режим доступа: [https://en.wikipedia.org/wiki/ISpring\\_Suite](https://en.wikipedia.org/wiki/ISpring_Suite) (дата обращения: 15.11.2017).

27. JavaScript [Электронный ресурс]. — Режим доступа: <https://ru.wikipedia.org/wiki/JavaScript> (дата обращения: 15.11.2017).

28. PHP [Электронный ресурс]. — Режим доступа: <https://ru.wikipedia.org/wiki/PHP> (дата обращения: 11.12.2017).

29. PHP Start [Электронный ресурс]. — Режим доступа: <https://php-start.com/> (дата обращения: 12.09.2017).

30. PHP.SU [Электронный ресурс]. — Режим доступа: <http://www.php.su/> (дата обращения: 13.05.2017).

31. [www.site-do.ru](http://www.site-do.ru) [Электронный ресурс]. — Режим доступа: [https://www.site-do.ru/php/uroki\\_php.php](https://www.site-do.ru/php/uroki_php.php) (дата обращения: 20.08.2017).



# ПРИЛОЖЕНИЕ

**Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования**

**«Российский государственный профессионально-педагогический университет»**

Институт инженерно-педагогического образования  
Кафедра информационных систем и технологий  
направление 44.03.04 Профессиональное обучение (по отраслям)  
профиль «Информатика и вычислительная техника»  
профилизация «Компьютерные технологии»

УТВЕРЖДАЮ

Заведующий кафедрой

\_\_\_\_\_ Н. С. Толстова

« \_\_\_\_\_ » \_\_\_\_\_ 2018 г.

## ЗАДАНИЕ

### на выполнение выпускной квалификационной работы бакалавра

студента 4 курса, группы ЗКТ-401С Балакина Александра Олеговича

1. Тема: Электронное учебное пособие «Разработка WPF приложений в C#»  
утверждена распоряжением по институту от \_\_\_\_\_ г. №

.

2. Руководитель Ширева Светлана Николаевна, ст. преподаватель каф. ИС ФГАОУ ВО РГППУ

3. Место преддипломной практики ФГАОУ ВО РГППУ

4. Исходные данные к ВКР рабочая программа учебной дисциплины «Алгоритмические языки и системы программирования», Петсольд Ч. «Microsoft Windows Presentation Foundation», Шмитт К. «CSS. Рецепты программирования», НОУ Интуит — курс «Microsoft: Разработка приложений на базе WPF», Петюшкин А. В. «HTML в Web-дизайне».

5. Содержание текстовой части ВКР (перечень подлежащих разработке вопросов)

- провести анализ литературных и интернет-источников по теме «Разработка WPF приложений в C#»;
- произвести анализ учебной программы дисциплины «Алгоритмические языки и системы программирования» с целью определения структуры электронного учебного пособия;
- систематизировать и структурировать учебный материал по рассматриваемой теме;
- разработать дизайн, удобный интерфейс, структурировать материал электронного учебного пособия;
- реализовать электронное учебное пособие «Разработка WPF приложений в C#».

6. Перечень демонстрационных материалов

Презентация, выполненная средствами Microsoft PowerPoint.  
учебное пособие «Разработка WPF приложений в С#».

7. Календарный план выполнения выпускной квалификационной работы

№ п/п	Наименование этапа ВКР	Срок выполнения этапа	Процент выполнения ВКР	Отметка руководителя о выполнении
1	Сбор информации по выпускной работе и сдача зачета по преддипломной практике	14.05.2017	10	
2	Выполнение работ по разрабатываемым вопросам их изложение в выпускной работе:			
	Анализ литературы и интернет-источников по проблеме исследования	22.05.2017	5	
	Выделение требований к электронным учебным пособиям	28.05.2017	5	
	Подбор, систематизация и структурирование материала по теме исследования	04.07.2017	5	
	Разработка интерфейса электронного учебного пособия	12.09.2017	20	
	Реализация электронного учебного пособия	18.09.2017	20	
3	Оформление текстовой части ВКР	08.10.2017	20	
4	Выполнение демонстрационных материалов к ВКР	03.01.2018	5	
5	Нормоконтроль	10.01.2018	5	
6	Подготовка доклада к защите в ГЭК	13.01.2018	5	

8. Консультанты по разделам выпускной квалификационной работы

Наименование раздела	Консультант	Задание выдал		Задание принял	
		подпись	дата	подпись	дата

Руководитель \_\_\_\_\_  
подпись дата

Задание получил \_\_\_\_\_  
подпись студента дата

9. Выпускная квалификационная работа и все материалы проанализированы.  
Считаю возможным допустить Балакина А. О. к защите выпускной квалификационной работы в государственной экзаменационной комиссии.

Руководитель \_\_\_\_\_  
подпись дата

10. Допустить Балакина А. О. к защите выпускной квалификационной работы в государственной экзаменационной комиссии (протокол заседания кафедры от \_\_.\_\_.2018 №\_\_\_\_)

Заведующий кафедрой \_\_\_\_\_  
подпись дата