

Министерство образования и науки Российской Федерации  
 Федеральное государственное бюджетное образовательное учреждение  
 высшего профессионального образования  
 «Самарский государственный технический университет»

УТВЕРЖДАЮ  
 Проректор по учебной работе СамГТУ

“ 29 ”  Я.М.Клебанов  
 2014 г.  
 М.П.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

**М1.В.ДВ.4.2 Компьютерные технологии мультимедиа**

*(указывается шифр и наименование дисциплины по учебному плану)*

Направление подготовки  
 (специальность)

230100.68 Информатика и вычислительная техника

*(код и наименование направления подготовки (специальности))*

Квалификация (степень) выпускника

Магистр

Магистерская программа

Информатика и вычислительная техника

Форма обучения

Очная

*(очная, очно-заочная, заочная)*

Выпускающая кафедра

Вычислительная техника

*(название)*

Кафедра-разработчик рабочей программы

Вычислительная техника

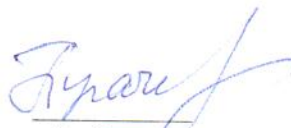
*(название)*

Семестр	Трудоем- кость час.	Лек- ций, час.	Практич. занятий, час.	Лабора- т. работ, час.	СРС, час	Зет	Форма промежуточного контроля (экз./зачет)
3	144	9		36	54	4	Экзамен,45
Итого	144	9		36	54	4	Экзамен,45

Самара  
 2014

Программа разработана в соответствии с требованиями Федерального закона от 27.12.2012 г. № 273-ФЗ «Об образовании в Российской Федерации», ФГОС ВО, Приказом Минобрнауки России от 19 декабря 2013 г. №1367 «Об утверждении порядка организации осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры» и учебного плана СамГТУ от 01.07.2014.

Составитель рабочей программы:  
к.т.н., доцент  
(должность, ученое звание, степень)

  
(подпись)  
26 августа 2014г.  
(дата)

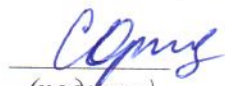
А.И.Пугачев  
(ФИО)

Рабочая программа утверждена на заседании кафедры:

«Вычислительная техника»  
(наименование кафедры-разработчика)

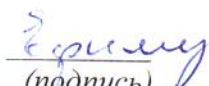
28 августа 2014г протокол № 1  
(дата и номер протокола)

зав. кафедрой-разработчиком

  
(подпись)  
28.08.2014г.  
(дата)


С.П. Орлов  
(ФИО)

Эксперт методической комиссии по  
УГНП

  
(подпись)  
28 августа 2014г.  
(дата)

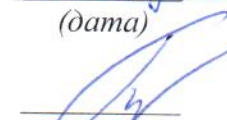
Н.В. Ефимушкина  
(ФИО)

Председатель методического совета  
факультета  
(на котором осуществляется обучение)

  
(подпись)  
28 августа 2014г.  
(дата)


В.В. Зайвый  
(ФИО)

Декан факультета  
(на котором осуществляется обучение)

  
(подпись)  
28 августа 2014г.  
(дата)

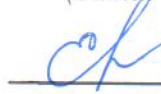
Н.Г. Губанов  
(ФИО)

СОГЛАСОВАНО:  
Зав. выпускающей кафедрой

  
(подпись)  
28 августа 2014г.  
(дата)

С.П. Орлов  
(ФИО)

Начальник УВО

  
(подпись)  
29.08.2014г.  
(дата)

О.Ю.Еремичева  
(ФИО)

## СОДЕРЖАНИЕ

1. Требования к результатам освоения дисциплины .....	4
2. Место дисциплины в структуре ОПОП .....	5
3. Структура и содержание дисциплины.....	6
3.1. Структура дисциплины.....	6
3.2. Содержание дисциплины .....	8
4. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине.....	11
5. Образовательные технологии.....	11
6. Формы контроля освоения дисциплины.....	11
6.1. Перечень оценочных средств для текущего контроля освоения дисциплины.....	11
6.2. Состав фонда оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине .....	12
7. Учебно-методическое и информационное обеспечение дисциплины .....	13
7.1. Перечень основной и дополнительной учебной литературы .....	13
7.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет».....	13
7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине .....	13
8. Материально-техническое обеспечение дисциплины .....	14
Дополнения и изменения к рабочей программе.....	15
Приложение 1. Аннотация рабочей программы .....	16



## 1. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Перечень планируемых результатов обучения по дисциплине определяется требованиями к результатам освоения ОПОП.

Таблица 1.

### Перечень планируемых результатов обучения по дисциплине

Планируемые результаты освоения ОПОП (компетенции), достижение которых обеспечивает дисциплина		Перечень планируемых результатов обучения по дисциплине
Коды компетенций	Содержание компетенций	
ОК-2	Способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности	<b>Знать:</b> новые методы исследования информационных технологий и вычислительной техники <b>Уметь:</b> применять перспективные методы исследований и решения профессиональных задач <b>Владеть:</b> навыками использования современных информационных технологий
ОК-4	Использование на практике умений и навыков в организации исследовательских и проектных работ, в управлении коллективом	<b>Знать:</b> этапы проведения исследовательских и проектных работ, методы контроля их выполнения <b>Уметь:</b> организовывать выполнение исследовательских и проектных работ, управлять коллективом разработчиков программного обеспечения <b>Владеть:</b> приемами ведения исследовательских и проектных работ
ПК-2	На основе знания педагогических приемов принятие непосредственного участия в учебной работе кафедр и других учебных подразделений по профилю направления «Информатика и вычислительная техника»"	<b>Знать:</b> педагогические приемы, формы организации и проведения занятий по направлению «Информатика и вычислительная техника» <b>Уметь:</b> доходчиво объяснять теоретический материал и вести практические занятия с обучающимися <b>Владеть:</b> существующими методами и алгоритмами решения задач распознавания и обработки данных
ПК-6	Применение современных технологий разработки программных комплексов с использованием CASE-средств, контроль качества разрабатываемых программных продуктов	<b>Знать:</b> способы применения CALS-технологий при разработке проекты программных комплексов, методы контроля качества разрабатываемых программных продуктов. <b>Уметь:</b> применять современные средства управления проектами при реализации планов информатизации предприятий. <b>Владеть:</b> навыками применения современных технологий разработки программных комплексов с



		использованием CASE-средств.
ПК-7	Организация работ и руководство коллективами разработчиков аппаратных и/или программных средств информационных и автоматизированных систем	<p><b>Знать:</b> методы настройки, наладки и эксплуатации программно-аппаратных комплексов вычислительных систем, методы оценки трудоемкости и ресурсоемкости проектов.</p> <p><b>Уметь:</b> проектировать вычислительные системы, применять перспективные методы решения профессиональных задач синтеза вычислительных систем на основе знания мировых тенденций развития вычислительной техники и информационных технологий; разрабатывать технические задания для проектов на разработку современных программных комплексов.</p> <p><b>Владеть:</b> навыками работы с технической документацией на современные программные комплексы, организации работы и руководства коллективом разработчиков программного обеспечения</p>

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Компьютерные технологии мультимедиа» относится к вариативной части блока 2 учебного плана.

Перечень предшествующих и последующих дисциплин, формирующих общекультурные и профессиональные компетенции.

Таблица 2.

№ п/п	Наименование компетенции	Предшествующие дисциплины	Последующие дисциплины (группы дисциплин)
<i>Общекультурные компетенции</i>			
1	ОК-2, способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности	Методология научных исследований, Вычислительные системы, Теория проектирования систем, менеджмент инноваций, Вычислительные системы	Математическая статистика, Теория проектирования систем, Средства коммуникации на транспорте
2	ОК-4, использование на практике умений и навыков в организации исследовательских и проектных работ, в управлении коллективом	Технология программирования	Геоинформационные системы, Медицинские информационные технологии, Интеллектуальные транспортные системы, Технологии телемедицины

№ п/п	Наименование компетенции	Предшествующие дисциплины	Последующие дисциплины (группы дисциплин)
<i>Профессиональные компетенции</i>			
3	ПК-2, на основе знания педагогических приемов принятие непосредственного участия в учебной работе кафедр и других учебных подразделений по профилю направления «Информатика и вычислительная техника»	Теоретическая информатика, Управление проектами	Математическая статистика, Теоретическая информатика, Геоинформационные системы, Управление проектами
4	ПК-6, применение современных технологий разработки программных комплексов с использованием CASE-средств, контроль качества разрабатываемых программных продуктов	Теория проектирования систем, Технология программирования	Теория проектирования систем, Интеллектуальные системы и базы знаний, Информационные технологии в медицине, Медицинские информационные технологии, Интеллектуальные транспортные системы Технологии медицины, Средства коммуникации на транспорте
5	ПК-7, организация работ и руководство коллективами разработчиков аппаратных и/или программных средств информационных и автоматизированных систем	Технология программирования, Вычислительные системы	Управление проектами, Информационные технологии в медицине, Информационные технологии транспортных систем



### 3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

#### 3.1 Структура дисциплины

Общая трудоемкость дисциплины составляет 4 зачетных единиц, 144 академических часов.

#### Трудоемкость дисциплины и виды учебной работы

Таблица 3.

##### Объем дисциплины по видам учебных занятий

Вид учебной работы	Всего часов	Контактная работа	Семестр
			2
<b>Аудиторные занятия (всего)</b>	<b>45</b>		<b>45</b>
В том числе:			
Лекции	9	9	9
Лабораторные работы (ЛР)	36	36	36
<b>Самостоятельная работа (всего)</b>	<b>99</b>		<b>99</b>
В том числе:			
Подготовка к лабораторным работам и оформление отчетов	54		54
Вид промежуточной аттестации (зачет, экзамен)	<b>Экзамен, 45</b>	8	<b>Экзамен, 45</b>
<b>ИТОГО:</b>	<b>144</b>		<b>144</b>
час. Зач. ед.	<b>4</b>		<b>4</b>
Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)		<b>53</b>	

#### Распределение учебной нагрузки по разделам дисциплины

Таблица 4.

№ модуля образовательной программы	№ раздела	Наименование раздела дисциплины	Виды учебной нагрузки и их трудоемкость, часы				
			Лекции	Практические занятия	Лабораторные работы	СРС	Всего часов
	1	Мультимедиа в дизайне интерактивных приложений	2		2		4
	2	Основы XAML MS SilverLight	4		6	12	22
	3	Оснащение приложений мультимедиа-средствами	3		28	42	73
	1-3	Подготовка к экзамену					45
<b>ИТОГО:</b>			<b>9</b>		<b>36</b>	<b>54</b>	<b>144</b>

### 3.2 Содержание дисциплины

#### Лекционный курс

Таблица 5.

№ лекции	Номер раздела	Тема лекции и перечень дидактических единиц	Трудоемкость, часов
	1	<b>МУЛЬТИМЕДИА В ИНТЕРАКТИВНЫХ ПРИЛОЖЕНИЯХ</b>	
1		<b>Тема 1.1 Технологии мультимедиа</b>	2
		1.1.1 <i>Введение</i> . Дисциплина «Компьютерные технологии мультимедиа», его задачи и место в подготовке магистров.	
		1.1.2 <i>Современные технологии мультимедиа</i> . HTML, JavaScript, Flash, MS Silverlight.	
		<b>Тема 1.2 Знакомство с MS Silverlight</b>	
		1.2.1 <i>Назначение и основы применения MS Silverlight</i> . Назначение MS Silverlight, состав, характеристики.	
		1.2.2 <i>Использование MS Silverlight</i> . Архитектура MS Silverlight. MS Silverlight и язык разметки XAML.	
	2	<b>ОСНОВЫ XAML MS SILVERLIGHT</b>	
2		<b>Тема 2.1 XAML – язык описания интерфейса</b>	2
		2.1.1 <i>Основные конструкции</i> . Связь XAML с XML. Теги. Синтаксис тегов. Иерархия. 2.1.2 <i>Графические ресурсы и объекты</i> . Элемент управления Canvas. Цвета, кисти. Фигуры в XAML.	
3		<b>Тема 2.2 Элементы управления в XAML</b>	2
		2.2.1 <i>Элементы компоновки</i> . Элемент управления StackPanel. Элемент управления Grid.	
		2.2.2 <i>Базовые элементы управления</i> . Класс Control. Кнопки. Текстовые элементы управления. Элементы на основе диапазона значений.	
	3	<b>ОСНАЩЕНИЕ ПРИЛОЖЕНИЙ МУЛЬТИМЕДИА-СРЕДСТВАМИ</b>	
4		<b>Тема 3.1 Геометрические объекты и графика</b>	2
		3.1.1 <i>Фигуры в XAML</i> . Line, Rectangle, Ellipse. Использование геометрических объектов.	
		3.1.2 <i>Трансформация</i> . Основные виды трансформации. Совмещение преобразований.	
		3.2.2 <i>Анимация</i> . Основные виды анимации. Запуск анимации. Параметры анимации. Анимация с помощью ключевых кадров.	
5		<b>Тема 3.3 Работа с аудио и видео</b>	1
		3.3.1 <i>Работа с аудио</i> . Автоматическое воспроизведение звука. Управление воспроизведением звука.	
		3.3.2 <i>Работа с видео</i> . Управление текущим состоянием видео. Управление положением курсора воспроизведения. Использование маркеров временной шкалы мультимедиа.	
Итого:			9



## Практические занятия

Практические занятия учебным планом не предусмотрены.

## Лабораторные работы

Таблица 7.

№ лаб. раб.	Номер раздела	Наименование лабораторной работы и перечень дидактических единиц	Трудоемкость, часов
1	1, 2	<b>Разработка SilverLight-приложения в среде MS Visual Studio</b> Шаблон Silverlight.	4
2	2	<b>Элементы управления Silverlight</b> Класс Control. Кнопки. Текстовые элементы управления. Элементы на основе диапазона значений.	4
3	3	<b>Геометрические объекты и графика</b> Кисти, системные цвета. Line, Rectangle, Ellipse. Использование геометрических объектов.	4
4	3	<b>Трансформация</b> Плоско-параллельное перемещение с помощью TranslateTransform/ Вращение с помощью RotateTransform. Масштабирование с помощью ScaleTransform. Совмещение преобразований.	4
5	3	<b>Анимация</b> Запуск анимации. Параметры анимации. Анимация с помощью ключевых кадров.	4
6	3	<b>Воспроизведение аудио</b> Автоматическое воспроизведение звука. Управление воспроизведением звука.	4
7	3	<b>Воспроизведение видео</b> Управление текущим состоянием видео.	4
8	3	<b>Создание индикатора загрузки и воспроизведения видео</b> Управление положением курсора воспроизведения. Использование маркеров временной шкалы мультимедиа.	4
9	3	<b>Создание стиля приложения</b> Создание стиля. Шаблоны стилей.	4
Итого:			36

### Самостоятельная работа студента

Таблица 8.

Раздел дисциплины	№ подраздела	Вид самостоятельной работы студента (СРС) и перечень дидактических единиц	Трудоемкость, часов
Разделы 1, 2	1.1, 2.2	Подготовка к лабораторной работе № 1 Подключение среды Silverlight к MS Visual Studio. Разработка программного кода приложения.	4
	2.2	Оформление отчета по лабораторной работе № 1	2
Раздел 3	2.2	Подготовка к лабораторной работе № 2 Элементы управления Silverlight Класс Control. Кнопки. Текстовые элементы управления. Элементы на основе диапазона значений.	4
	2.2	Оформление отчета по лабораторной работе № 2	2
	3.1	Подготовка к лабораторной работе № 3 Геометрические объекты и графика Кисти, системные цвета. Line, Rectangle, Ellipse. Использование геометрических объектов.	4
	3.2	Оформление отчета по лабораторной работе № 3	2
	3.2	Подготовка к лабораторной работе № 4 Трансформация Плоско-параллельное перемещение, масштабирование, вращение, совмещение преобразований	4
	3.2	Оформление отчета по лабораторной работе № 4	2
	3.2	Подготовка к лабораторной работе № 5 Анимация Запуск анимации. Параметры анимации. Анимация с помощью ключевых кадров.	4
	3.2	Оформление отчета по лабораторной работе № 5	2
	3.3	Подготовка к лабораторной работе № 6 Воспроизведение аудио Автоматическое воспроизведение звука. Управление воспроизведением звука.	4
	3.3	Оформление отчета по лабораторной работе № 6	2
	3.3	Подготовка к лабораторной работе № 7 Воспроизведение видео Управление текущим состоянием видео.	4
	3.3	Оформление отчета по лабораторной работе № 7	2
	3.3	Подготовка к лабораторной работе № 8 Создание индикатора загрузки и воспроизведения видео Управление положением курсора воспроизведения.	4
	3.3	Оформление отчета по лабораторной работе № 8	2
	3.4	Подготовка к лабораторной работе № 9 Создание стиля приложения Создание стиля. Шаблоны стилей.	4
3.4	Оформление отчета по лабораторной работе № 9	2	
		Итого:	54
Разделы 1-3		Подготовка к экзамену	45
		Всего:	<b>99</b>



#### 4. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

1. Пугачев, А. И. Основы компьютерной графики [Текст]: курс лекций / А. И. Пугачев; Самар. гос. техн. ун-т. - Самара : [б. и.], 2011. - 104 с. : ил.
2. Байдачный С.С. Silverlight 4: Создание насыщенных Web-приложений [Текст] – М.: СОЛОН-ПРЕСС, 2010. – 288 с.

#### 5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Интерактивные образовательные технологии, используемые в аудиторных занятиях

Таблица 9.

Семестр	Вид и тема занятия (лекция, практическое занятие, лабораторная работа)	Используемые интерактивные образовательные технологии	Количество часов
2	Лабораторная работа № 1	MS Visual Studio	2
	Лабораторная работа № 2	Компьютерная симуляция элементов управления Silverlight	2
	Лабораторная работа № 3	Компьютерная симуляция геометрических объектов	2
	Лабораторная работа № 4	Компьютерная симуляция операций трансформации	2
	Лабораторная работа № 5	Компьютерная симуляция операций анимации	2
	Лабораторная работа № 6	Компьютерное воспроизведение аудио	2
	Лабораторная работа № 7	Компьютерное воспроизведение видео	2
	Лабораторная работа № 8	Компьютерная симуляция индикации загрузки видео	2
	Лабораторная работа № 9	Компьютерная симуляция стиля оформления приложения	2
Итого:			18

#### 6. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

##### 6.1. Перечень оценочных средств для текущего контроля освоения дисциплины

*Текущая аттестация* студентов производится в дискретные временные интервалы лектором и преподавателями, ведущими лабораторные работы по дисциплине в следующих формах:

- выполнение лабораторных работ;
- защита лабораторных работ.

*Рубежная аттестация* студентов производится по окончании раздела в форме отчета по лабораторным работам.

*Промежуточный контроль* по результатам семестра по дисциплине проходит в форме письменного экзамена (включает в себя ответ на теоретические вопросы).

## **6.2. Состав фонда оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине**

Фонд оценочных средств, перечень заданий для проведения промежуточной аттестации, а также методические указания для проведения промежуточной аттестации приводятся в Приложении.

### **Примерный перечень вопросов к экзамену**

1. Технологии оснащения интерактивных приложений средствами мультимедиа
2. Формат HTML
3. Назначение MS Silverlight, состав, характеристики
4. Архитектура MS Silverlight
5. Использование MS Silverlight с MS Expression Studio
5. Подключение MS Silverlight к проекту Visual Studio
6. Объекты MS Silverlight
7. MS Silverlight и язык разметки XAML
8. Язык описания интерфейса XAML. Основные конструкции
9. Элемент управления Canvas
10. Элемент управления StackPanel
11. Элемент управления Grid
12. Базовые элементы управления. Класс Control
13. Кисти XAML. Системные цвета
14. Фигуры в XAML. Использование геометрических объектов
15. Трансформация. Основные виды трансформации
16. Анимация. Основные виды анимации
17. Запуск анимации. Параметры анимации.
18. Анимация с помощью ключевых кадров.
19. Воспроизведение аудиоинформации
20. Управление воспроизведением звука
21. Воспроизведение видеоинформации
22. Управление текущим состоянием видео
23. Использование маркеров временной шкалы мультимедиа
24. Ресурсы приложения
25. Ресурсы объектов
26. Стили. Понятие стиля. Создание стиля. Шаблоны стилей.



## 7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 7.1. Перечень основной и дополнительной учебной литературы

Таблица 10.

Обеспечение дисциплины основной и дополнительной литературой по дисциплине

N п/п	Автор	Название	Место издания	Наименование издательства	Год издания	Кол-во экз.
<b>Основная литература</b>						
1	Пугачев А.И.	Основы компьютерной графики	Самара	Самар. гос. техн. ун-т	2011	50
2	Крапивенко А.В.	Технологии мультимедиа и восприятие ощущений: учебное пособие	Москва	БИНОМ. Лаборатория знаний. – ЭБС «IPRbooks»	2012	электронный
3	Стативко Р.У.	Информационные технологии: учебное пособие	Белгород	Белгородский гос. технол.ун-т им. В.Г. Шухова– ЭБС «IPRbooks»	2012	электронный
4	Дьяконов В.П.	Новые информационные технологии: учебное пособие	Москва	СОЛОН-ПРЕСС – ЭБС «IPRbooks»	2008	электронный
<b>Дополнительная литература</b>						
1	Тузовский А.Ф.	Проектирование и разработка web- приложений: учебное пособие	Томск	Томский политехн. и-нт. – ЭБС «IPRbooks»	2014	электронный
2	Бердышев С.Н.	Искусство оформления сайта. 2-е изд.: учебное пособие	Москва	Дашков и К – ЭБС «IPRbooks»	2012	электронный
3	Аббасов И.Б.	Основы графического дизайна на компьютере в Photoshop CS6: учебное пособие	Москва	ДМК Пресс К – ЭБС «IPRbooks»	2013	электронный

• **Методические указания и материалы**

1. Компьютерная графика [Текст] : учебно-методическое пособие / А.И. Пугачев; Самар. гос. техн. ун-т – Самара: [б. и.], 2010. - 48 с. : ил.
2. Шешунова, Г. Г. Компьютерная графика [Текст] : лаборатор. практикум / Самар.гос.техн.ун-т. - Самара : [б. и.], 2007. - 145 с. : ил, табл.

### 7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет»

- сайт Microsoft SilverLight Toolkint. – Режим доступа: <http://silverlight.codeplex.com/>
- сайт компании Microsoft. – Режим доступа: <http://silverlight.net/showcase/>
- сетевой журнал «Компьютерная графика и мультимедиа». – Режим доступа: <http://cgm.computergraphics.ru/>
- уроки программирования на OpenGL. – Режим доступа: <http://pmg.org.ru/nehe/>

### **7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине**

1. Интегрированная среда разработки программного обеспечения Microsoft Visual Studio
2. Универсальная интегрированная среда разработки приложений Net Beans IDE and Net Beans Platform
3. Программная платформа Microsoft SilverLight
4. Библиотека Tao Framework

## **8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

1. Лекционные занятия:
  - комплект электронных презентаций/слайдов;
  - аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук).
2. Лабораторные работы:
  - лаборатория кафедры «Вычислительная техника», класс сетевых технологий, оснащенный компьютерами, объединенными в локальную сеть;
  - программное обеспечение, размещенное на сервере кафедры и на компьютерах в лабораториях.
3. Прочее:
  - рабочее место преподавателя, оснащенное компьютером с доступом в Интернет,
  - рабочие места студентов, оснащенные компьютерами с доступом в Интернет, предназначенные для работы в электронной образовательной среде.



**Дополнения и изменения в рабочей программе  
дисциплины «Компьютерные технологии мультимедиа»  
на 20\_\_/20\_\_ уч.г.**

Внесенные изменения на 20\_\_/20\_\_ учебный год

**УТВЕРЖДАЮ  
Проректор по учебной работе**

\_\_\_\_\_  
(подпись, расшифровка подписи)

" \_\_\_\_ " \_\_\_\_\_ 20... г

В рабочую программу вносятся следующие изменения:

- 1) .....
- 2) .....

или делается отметка о нецелесообразности внесения каких-либо изменений на данный учебный год

Рабочая программа пересмотрена на заседании кафедры

\_\_\_\_\_  
(дата, номер протокола заседания кафедры, подпись зав. кафедрой).

*УГС* ОДОБРЕНА на заседании методической комиссии по УГС " \_\_\_\_ " \_\_\_\_\_ 20\_\_ г."

*УГС* Эксперты методической комиссии по УГС (не менее двух)

\_\_\_\_\_  
*шифр наименование личная подпись расшифровка подписи дата*

**СОГЛАСОВАНО:**

Заведующий выпускающей кафедрой

\_\_\_\_\_  
*наименование кафедры личная подпись расшифровка подписи дата*

Декан

\_\_\_\_\_  
*наименование факультета, где производится обучение, личная подпись расшифровка подписи дата*

Начальник УВППО

\_\_\_\_\_  
*личная подпись расшифровка подписи дата*

### **Аннотация рабочей программы**

Дисциплина «Компьютерные технологии мультимедиа» относится к вариативной части блока М2. учебного плана профессионального блока дисциплин магистерской подготовки по направлению 230100.68 «Информатика и вычислительная техника». Дисциплина реализуется кафедрой «Вычислительная техника» на факультете автоматики и информационных технологий Самарского государственного технического университета.

#### **Требования к уровню освоения содержания дисциплины.**

Дисциплина «Компьютерные технологии мультимедиа» нацелена на формирование общих, общепрофессиональных и профессиональных компетенций, необходимых для реализации научно-исследовательской, проектно-конструкторской, проектно-технологической деятельности:

- ОК-2, способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности;
- ОК-4, использование на практике умений и навыков в организации исследовательских и проектных работ, в управлении коллективом;
- ПК-2, на основе знания педагогических приемов принятие непосредственного участия в учебной работе кафедр и других учебных подразделений по профилю направления «Информатика и вычислительная техника»;
- ПК-6, применение современных технологий разработки программных комплексов с использованием CASE-средств, контроль качества разрабатываемых программных продуктов;
- ПК-7, организация работ и руководство коллективами разработчиков аппаратных и/или программных средств информационных и автоматизированных систем.

Содержание дисциплины охватывает круг вопросов, связанных с синтезом и анимацией 3D-сцен, трансформацией объектов, теорией и приемами графического дизайна, принципами создания Web-сайтов, мультимедиа-дизайном информационных систем, Web-страниц и сайтов.

**Преподавание дисциплины предусматривает следующие формы организации учебного процесса:** лекции, лабораторные работы, самостоятельная работа студента, консультации.

**Программой дисциплины предусмотрены следующие виды контроля:** текущий и рубежный контроль успеваемости в форме отчетов по лабораторным работам и промежуточный контроль в форме экзамена.

**Общая трудоемкость освоения дисциплины** составляет 4 зачетных единицы, 144 часов. Программой дисциплины предусмотрены лекционные занятия (9 часов), лабораторные работы (36 часов), самостоятельная работа (54 часа), подготовка к экзамену (45 часов).



## Методические указания для обучающихся по освоению дисциплины

Вид учебных занятий	Организация деятельности студента
Лекция	<p><b>Написание конспекта лекций:</b> определить роль компьютерных технологий мультимедиа в дизайне интерактивных приложений.</p> <p>Сравнительный анализ HTML, JavaScript, Flash, MS Silverlight в оснащении приложений мультимедиа.</p> <p>Архитектура MS Silverlight.</p> <p>Основы XAML MS SilverLight, синтаксис основных конструкций.</p> <p>В процессе изучения следует обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удается разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии. Уделить внимание следующим понятиям: основные конструкции XAML; теги; синтаксис тегов; элементы компоновки, контейнер Grid; графические объекты и графика; трансформация и анимация графических объектов.</p>
Индивидуальные задания	<p>Знакомство с основной и дополнительной литературой. Составление аннотаций к прочитанным литературным источникам. Разработка кода разметки для типовых задач. Исследование алгоритма и подсистемы на модели.</p>
Практикум / лабораторная работа	<p>Методические указания к выполнению лабораторных работ по дисциплине «Компьютерные технологии мультимедиа» находятся на кафедре «Вычислительная техника»</p>
Подготовка к экзамену	<p>При подготовке к экзамену необходимо ориентироваться на конспекты лекций, рекомендуемую литературу и Методические указания к выполнению лабораторных работ.</p>



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

---

Кафедра "Вычислительная техника"

# КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ МУЛЬТИМЕДИА

*Методические указания к самостоятельной работе обучающихся*

Направление 230100.68 – Информатика и вычислительная техника

Самара  
Самарский государственный технический университет  
2015



## Введение

Цель самостоятельной работы обучающегося по дисциплине «Компьютерные технологии мультимедиа» - формирование профессиональных компетенций, необходимых для реализации проектно-конструкторской, проектно-технологической и сервисно-эксплуатационной деятельности магистров по направлению «Информатика и вычислительная техника».

Обучающийся должен уметь участвовать в исследовании, настройке и наладке программно-аппаратных комплексов, сопрягать аппаратные и программные средства в составе информационных и автоматизированных систем, обосновывать принимаемые проектные решения, выполнять эксперименты по проверке их корректности и эффективности.

В процессе самостоятельной работы студент должен овладеть и закрепить знания об архитектуре Silverlight, языке разметки XAML, методах оснащения приложений средствами мультимедиа.

### Задания для самостоятельной работы

#### Подготовка к лабораторной работе № 1

Общее время на самостоятельную работу – 1 час

Таблица 1

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
Задания для овладения знаниями	<i>Чтение текста книги:</i> Создание приложения в Visual Studio [1], С. 42 – 44. <i>Чтение текста дополнительной литературы:</i> Создание автономного проекта Silverlight [3] (С.31 - 34).	0,4
Задания для закрепления и систематизации знаний	<i>Работа с конспектом лекций:</i> изучить лекцию № 3, тема 1.3 Использование MS Silverlight с MS Visual Studio. Ответить на контрольные вопросы по лабораторной работе № 1 согласно методическим указаниям.	0,3
Задания для формирования умений	Разработать простейшее приложение в Visual Studio с использованием Silverlight согласно методическим указаниям.	0,3

#### Оформление отчета по лабораторной работе № 1 – 0,4 час.

Описать выполненные эксперименты по лабораторной работе в соответствии с формой, приведенной в методических указаниях. Подготовить ответы на контрольные вопросы. Подготовить ответы по сущности проведенных экспериментов, сделать выводы по результатам исследования. Образцы оформления титульного листа отчета и материалов исследований и экспериментов приведены в Приложении.

## Подготовка к лабораторной работе № 2

Общее время на самостоятельную работу – 1 час

Таблица 2

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
Задания для овладения знаниями	<i>Чтение текста книги:</i> Базовые элементы управления [1], с.105-112 <i>Чтение текста дополнительной литературы:</i> Элементы [3]. С.157-184.	0.4
Задания для закрепления и систематизации знаний	<i>Работа с конспектом лекций:</i> изучить лекцию № 3, тема 1.3.2 Использование MS Silverlight. Ответить на контрольные вопросы по лабораторной работе № 2 согласно методическим указаниям.	0.3
Задания для формирования умений	Разработать приложение в Visual Studio с использованием Silverlight согласно методическим указаниям.	0.3

Оформление отчета по лабораторной работе № 2 – 0,4 час.

Описать выполненные эксперименты по лабораторной работе в соответствии с формой, приведенной в методических указаниях. Подготовить ответы на контрольные вопросы. Подготовить ответы по сущности проведенных экспериментов, сделать выводы по результатам исследования. Образцы оформления титульного листа отчета и материалов исследований и экспериментов приведены в Приложении.

## Подготовка к лабораторной работе № 3

Общее время на самостоятельную работу – 0,7 час.

Таблица 3

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
Задания для овладения знаниями	<i>Чтение текста книги:</i> Графические примитивы [1], С.145-149. <i>Чтение текста дополнительной литературы:</i> Двумерная компьютерная графика [2], С.15-35.	0,3
Задания для закрепления и систематизации знаний	<i>Работа с конспектом лекций:</i> изучить лекцию № 6 , тема 3.1 Геометрические объекты и графика. Ответить на контрольные вопросы по лабораторной работе № 3 согласно методическим указаниям.	0,2
Задания для формирования умений	Разработать приложение в Visual Studio с использованием Silverlight согласно методическим указаниям.	0,2

Оформление отчета по лабораторной работе № 3 – 0,4 час.



Описать выполненные эксперименты по лабораторной работе в соответствии с формой, приведенной в методических указаниях. Подготовить ответы на контрольные вопросы. Подготовить ответы по сущности проведенных экспериментов, сделать выводы по результатам исследования. Образцы оформления титульного листа отчета и материалов исследований и экспериментов приведены в Приложении.

### Подготовка к лабораторной работе № 4

Общее время на самостоятельную работу – 0,7 час.

Таблица 4

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
Задания для овладения знаниями	<i>Чтение текста книги:</i> Трансформация [1] С. 161-167. <i>Чтение текста дополнительной литературы:</i> Классы преобразований [3] С.287-292	0,4
Задания для закрепления и систематизации знаний	<i>Работа с конспектом лекций:</i> изучить лекцию № 7 , тема 3.2.1. Трансформация. Ответить на контрольные вопросы по лабораторной работе № 4 согласно методическим указаниям.	0,3
Задания для формирования умений	Разработать приложение в Visual Studio с использованием Silverlight согласно методическим указаниям.	0,3

### Оформление отчета по лабораторной работе № 4 – 0,4 час.

Описать выполненные эксперименты по лабораторной работе в соответствии с формой, приведенной в методических указаниях. Подготовить ответы на контрольные вопросы. Подготовить ответы по сущности проведенных экспериментов, сделать выводы по результатам исследования. Образцы оформления титульного листа отчета и материалов исследований и экспериментов приведены в Приложении.

### Подготовка к лабораторной работе № 5

Общее время на самостоятельную работу – 0,7 час.

Таблица 5

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
Задания для овладения знаниями	<i>Чтение текста книги:</i> Введение в анимацию. [1] С.167-170 <i>Чтение текста дополнительной литературы:</i> Основы анимации [3] С.331-357.	0,4
Задания для закрепления и	<i>Работа с конспектом лекций:</i> изучить лекцию № 7 , тема 3.3.2 Анимация.	0,3

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
систематизации знаний	Ответить на контрольные вопросы по лабораторной работе № 5 согласно методическим указаниям.	
Задания для формирования умений	Разработать приложение в Visual Studio с использованием Silverlight согласно методическим указаниям.	0,3

Оформление отчета по лабораторным работам № 5 – 0,4 час.

Описать выполненные эксперименты по лабораторной работе в соответствии с формой, приведенной в методических указаниях. Подготовить ответы на контрольные вопросы. Подготовить ответы по сущности проведенных экспериментов, сделать выводы по результатам исследования. Образцы оформления титульного листа отчета и материалов исследований и экспериментов приведены в Приложении.

### Подготовка к лабораторной работе № 6

Общее время на самостоятельную работу – 0,7 час.

Таблица 6

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
Задания для овладения знаниями	<i>Чтение текста книги:</i> Работа с аудио и видео [1] С. 172-184. <i>Чтение текста дополнительной литературы:</i> Управление воспроизведением [3] С. 393-398.	0,4
Задания для закрепления и систематизации знаний	<i>Работа с конспектом лекций:</i> изучить лекцию № 8, тема 3.1.1 Работа с аудио. Ответить на контрольные вопросы по лабораторной работе № 6 согласно методическим указаниям.	0,3
Задания для формирования умений	Разработать приложение в Visual Studio с использованием Silverlight согласно методическим указаниям.	0,3

Оформление отчета по лабораторной работе № 6 - 0,4 час.

Описать выполненные эксперименты по лабораторной работе в соответствии с формой, приведенной в методических указаниях. Подготовить ответы на контрольные вопросы. Подготовить ответы по сущности проведенных экспериментов, сделать выводы по результатам исследования. Образцы оформления титульного листа отчета и материалов исследований и экспериментов приведены в Приложении.



## Подготовка к лабораторной работе № 7

Общее время на самостоятельную работу – 0,7 час.

Таблица 7

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
Задания для овладения знаниями	<i>Чтение текста книги:</i> Работа с аудио и видео [1] С. 183-191. <i>Чтение текста дополнительной литературы:</i> Воспроизведение видеофайлов [3] С. 398-405.	0,4
Задания для закрепления и систематизации знаний	<i>Работа с конспектом лекций:</i> изучить лекцию № 8, тема 3.3.2 Работа с видео. Ответить на контрольные вопросы по лабораторной работе № 7 согласно методическим указаниям.	0,3
Задания для формирования умений	Разработать приложение в Visual Studio с использованием Silverlight согласно методическим указаниям.	0,3

Оформление отчета по лабораторной работе № 7 - 0,4 час.

Описать выполненные эксперименты по лабораторной работе в соответствии с формой, приведенной в методических указаниях. Подготовить ответы на контрольные вопросы. Подготовить ответы по сущности проведенных экспериментов, сделать выводы по результатам исследования. Образцы оформления титульного листа отчета и материалов исследований и экспериментов приведены в Приложении.

## Подготовка к лабораторной работе № 8

Общее время на самостоятельную работу – 0,7 час.

Таблица 8

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
Задания для овладения знаниями	<i>Чтение текста дополнительной литературы:</i> Программные средства, применяемые при разработке драйверов [3] С. 406-413.	0,4
Задания для закрепления и систематизации знаний	<i>Работа с конспектом лекций:</i> изучить лекцию № 8, тема 3.3.2 Работа с видео. Ответить на контрольные вопросы по лабораторной работе № 8 согласно методическим указаниям.	0,3
Задания для формирования умений	Разработать приложение в Visual Studio с использованием Silverlight согласно методическим указаниям.	0,3

Оформление отчета по лабораторной работе № 8 - 0,4 час.

Описать выполненные эксперименты по лабораторной работе в соответствии с формой, приведенной в методических указаниях. Подготовить ответы на контрольные вопросы. Подготовить ответы по сущности проведенных экспериментов, сделать выводы по результатам исследования. Образцы оформления титульного листа отчета и материалов исследований и экспериментов приведены в Приложении.

### Подготовка к лабораторной работе № 9

Общее время на самостоятельную работу – 0,7 час.

Таблица 9

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
Задания для овладения знаниями	<i>Чтение текста книги:</i> Ресурсы и стили [1] С. 193-201. <i>Чтение текста дополнительной литературы:</i> Стили и поведения [3] С. 481-506.	0,4
Задания для закрепления и систематизации знаний	<i>Работа с конспектом лекций:</i> изучить лекцию № 9, тема 3.4 Ресурсы и стили приложений Ответить на контрольные вопросы по лабораторной работе № 9 согласно методическим указаниям.	0,3
Задания для формирования умений	Разработать приложение в Visual Studio с использованием Silverlight согласно методическим указаниям.	0,3

### Оформление отчета по лабораторной работе № 9 - 0,5 час.

Описать выполненные эксперименты по лабораторной работе в соответствии с формой, приведенной в методических указаниях. Подготовить ответы на контрольные вопросы. Подготовить ответы по сущности проведенных экспериментов, сделать выводы по результатам исследования. Образцы оформления титульного листа отчета и материалов исследований и экспериментов приведены в Приложении.



### Список рекомендуемой литературы

1. Байдачный, С.С. Silverlight 4: Создание насыщенных Web-приложений [Текст] : / С.С. Байдачный. - М.: СОЛОН-ПРЕСС, 2010. - 288 с. : ил. - ISBN 978- 5-91359-079-4
2. Пугачев, А.И. Основы компьютерной графики [Текст] : / А.И. Пугачев. - Самара : Самар. гос. техн. ун-т, 2011. - 104 с.
3. Мак-Дональд, Silverlight 5 с примерами на С# для профессионалов, 4-е изд. : Пер. с англ. - М. : ООО «И.Д. Вильямс», 2013. - 848 с. : ил. - ISBN 978-5-8459-1784-3.

В отчете должно быть описано:

- задание на выполнение лабораторной работы
- листинги кодов на XAML
- скрин-шоты
- выводы по результатам проведенных экспериментов.



Образец титульного листа отчета по лабораторным работам



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

---

Кафедра "Вычислительная техника"

КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ  
МУЛЬТИМЕДИА

Отчет по лабораторной работе № \_\_\_\_

**«Название лабораторной работы»**

Выполнил:

Студент

Проверили:

Самара 2015

## СОДЕРЖАНИЕ

Введение .....	2
Задания для самостоятельной работы.....	2
Подготовка к лабораторной работе № 1 .....	2
Подготовка к лабораторной работе № 2 .....	3
Подготовка к лабораторной работе № 3 .....	3
Подготовка к лабораторной работе № 4 .....	4
Подготовка к лабораторной работе № 5 .....	4
Подготовка к лабораторной работе № 6 .....	5
Подготовка к лабораторной работе № 7 .....	6
Подготовка к лабораторной работе № 8 .....	6
Подготовка к лабораторной работе № 9 .....	7
Список рекомендуемой литературы .....	8
Приложение .....	10



Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

«Самарский государственный технический университет»

**Факультет автоматизации и информационных технологий**

**Кафедра «Вычислительная техника»**

**УТВЕРЖДАЮ**

Заведующий кафедрой «Вычислительная техника»,

д.т.н., профессор

С.П. Орлов



## **ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**

**текущего контроля и промежуточной аттестации**

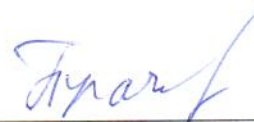
**дисциплины: М2.В.ДВ.4.2 Компьютерные технологии мультимедиа**

в составе основной образовательной программы по направлению подготовки  
(специальности): **230100.68 , «Информатика и вычислительная техника»**

по уровню высшего образования: **магистр**

направленность (профиль) программы: **«Информатика и вычислительная техника»**

Разработчик



Пугачев А.И.

(подпись)

Самара 2014

# Паспорт фонда оценочных средств

по дисциплине

## М2.В.ДВ.4.2 «Компьютерные технологии мультимедиа»

№ п/п	Контролируемые разделы дисциплины	Код контролируемой компетенции	Наименование оценочного средства
1.	Мультимедиа в дизайне интерактивных приложений	ОК-2, ОК4	реферат, экзаменационные билеты
2.	Основы XAML MS SilverLight	ПК-2, ПК-6	реферат, экзаменационные билеты
3.	Оснащение приложений мультимедиа-средствами	ПК-6, ПК-7	реферат, экзаменационные билеты

ОК-2, способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности;

ОК-4, использование на практике умений и навыков в организации исследовательских и проектных работ, в управлении коллективом;

ПК-2, на основе знания педагогических приемов принятие непосредственного участия в учебной работе кафедр и других учебных подразделений по профилю направления «Информатика и вычислительная техника»;

ПК-6, применение современных технологий разработки программных комплексов с использованием CASE-средств, контроль качества разрабатываемых программных продуктов;

ПК-7, организация работ и руководство коллективами разработчиков аппаратных и/или программных средств информационных и автоматизированных систем.

### Критерии выставления оценки:

- оценка **«отлично»** выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал монографической литературы, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач;

- оценка **«хорошо»** выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения;

- оценка **«удовлетворительно»** выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ;

- оценка **«неудовлетворительно»** выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы.

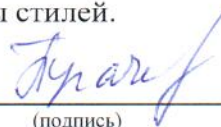


**Перечень вопросов к ЭКЗАМЕНУ (для промежуточной аттестации)  
по дисциплине**

**«Компьютерные технологии мультимедиа»**

1. Технологии оснащения интерактивных приложений средствами мультимедиа
2. Формат HTML
3. Назначение MS Silverlight, состав, характеристики
4. Архитектура MS Silverlight
5. Использование MS Silverlight с MS Expression Studio
5. Подключение MS Silverlight к проекту Visual Studio
6. Объекты MS Silverlight
7. MS Silverlight и язык разметки XAML
8. Язык описания интерфейса XAML. Основные конструкции
9. Элемент управления Canvas
10. Элемент управления StackPanel
11. Элемент управления Grid
12. Базовые элементы управления. Класс Control
13. Кисти XAML. Системные цвета
14. Фигуры в XAML. Использование геометрических объектов
15. Трансформация. Основные виды трансформации
16. Анимация. Основные виды анимации
17. Запуск анимации. Параметры анимации.
18. Анимация с помощью ключевых кадров.
19. Воспроизведение аудиоинформации
20. Управление воспроизведением звука
21. Воспроизведение видеоинформации
22. Управление текущим состоянием видео
23. Использование маркеров временной шкалы мультимедиа
24. Ресурсы приложения
25. Ресурсы объектов
26. Стили. Понятие стиля. Создание стиля. Шаблоны стилей.

Разработчик \_\_\_\_\_



(подпись)

Пугачев А. И.

**Темы рефератов**  
**по дисциплине**  
**«Компьютерные технологии мультимедиа»**

1. Технологии оснащения интерактивных приложений средствами мультимедиа
2. Adobe Flash и SilverLight в мультимедиа
3. Архитектура SilverLight
4. Язык разметки XAML
5. Графика в SilverLight
6. Анимация в SilverLight
7. Звук и видео в SilverLight
8. Трехмерные средства Silverlight
9. Сферы применения мультимедийных технологий
10. Мультимедиа в Web-дизайне
11. Компьютерные технологии мультимедиа в мобильных устройствах
12. Мультимедиа технологии в образовании
13. Виртуальные тренажеры в военной сфере
14. Дополненная реальность – новый вид мультимедиа технологий
15. Технологии создания мультимедиа продуктов

**Контролируемые компетенции:** ОК-3; ОПК-5; ПК-4.

Разработчик \_\_\_\_\_

  
(подпись)

Пугачев А. И.



Образец реферата.

## **Реферат на тему «Язык разметки XAML» по дисциплине «Компьютерные технологии мультимедиа»**

XAML (eXtensible Application Markup Language) – это декларативный язык разметки. С точки зрения модели программирования .NET Framework язык XAML упрощает создание пользовательского интерфейса для приложения .NET Framework. Можно создать видимые элементы пользовательского интерфейса в декларативной разметке XAML, а затем отделить определение пользовательского интерфейса от логики времени выполнения, используя файлы кода программной части, присоединенные к разметке с помощью определений разделяемых классов [3]. Язык XAML напрямую представляет создание экземпляров объектов в конкретном наборе резервных типов, определенных в сборках. В этом заключается его отличие от большинства других языков разметки, которые, как правило, представляют собой интерпретируемые языки без прямой связи с системой резервных типов. Язык XAML обеспечивает рабочий процесс, позволяющий нескольким участникам разрабатывать пользовательский интерфейс и логику приложения, используя потенциально различные средства.

При представлении в виде текста файлы XAML являются XML-файлами, которые обычно имеют расширение .xaml. Файлы можно сохранять в любой кодировке, поддерживаемой XML, но обычно используется кодировка UTF-8.

### **Объектные элементы XAML**

Объектный элемент обычно объявляет экземпляр типа. Этот тип определен в сборках, предоставляющих резервные типы для технологии, в которой XAML используется как язык.

Синтаксис объектного элемента всегда начинается с открывающейся угловой скобки (<). За ней следует имя типа, для которого требуется создать экземпляр. (Имя может содержать префикс. Описание этого понятия будет приведено далее.) После этого при необходимости можно объявить атрибуты объектного элемента. Для завершения тега объектного элемента введите закрывающую угловую скобку (>). Вместо этого можно использовать самозакрывающуюся форму, в которой отсутствует содержимое, установив в конце тега косую черту и закрывающую угловую скобку (/>).

Ниже приведен пример фрагмента разметки.

```
<StackPanel>  
  <Button Content="ОК"/>  
</StackPanel>
```

Здесь указано два объектных элемента: <StackPanel> (с содержимым и последующим закрывающим тегом) и <Button .../> (самозакрывающаяся форма с несколькими атрибутами). Объектные элементы StackPanel и Button сопоставлены с именем класса, определенным WPF и являющимся частью сборок WPF. При задании тега объектного элемента создается инструкция обработки XAML для создания нового экземпляра. Каждый экземпляр создается путем вызова

конструктора базового типа или структуры по умолчанию при синтаксическом анализе и загрузке XAML-кода.

## Элементы и их атрибуты

XAML предлагает очень простую и ясную схему определения различных элементов и их свойств. Например, добавим кнопку в создаваемую по умолчанию разметку окна:

```
<Window x:Class="WpfApplication1.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 HorizontalAlignment="Right" Width="517">
        <Ellipse Fill="AliceBlue" HorizontalAlignment="Left" Height="167"
Margin="129,78,50,0" Stroke="Black" VerticalAlignment="Top" Width="220"/>
        <Button x:Name="button1" Width="100" Height="30" Content="Кнопка"
Margin="336,269,81,21" />
    </Grid>
</Window>
```

Сначала идет элемент самого высшего уровня - Window, Здесь объявлено два базовых пространства имен:

`xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"` – это базовое пространство имен WPF, которое охватывает все классы WPF, включая элементы управления, которые применяются при построении пользовательского интерфейса. Так как данное пространство имен объявлено без префикса, то оно распространяется на весь XAML-документ;

`xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"` – пространство имен XAML. Оно включает различные свойства утилит XAML, которые позволяют влиять на то, как XAML-документ следует интерпретировать. Данное пространство имен отображается на префикс `x`. Этот префикс можно помещать перед именем элемента (`x:ИмяЭлемента`).

Затем идет вложенный элемент Grid - контейнер для других элементов, и в нем уже определены элементы Ellipse, задающий закрашенный эллипс и Button, представляющий кнопку.

На рис. 1 приведен результат применения данной разметки.

Для кнопки можно определить свойства в виде атрибутов. Здесь определены свойства `x:Name` (имя кнопки), `Width`, `Height` и `Content`. Подобным образом можно определить и другие свойства, которые нам нужны. Если не определять свойства, то они будут использовать значения по умолчанию.

При создании нового проекта WPF в дополнение к создаваемому файлу `MainWindow.xaml` создается также файл отделенного кода `MainWindow.xaml.cs`, где, как предполагается, должна находиться логика приложения связанная с разметкой из `MainWindow.xaml`. Файлы XAML позволяют определить интерфейс окна, но для создания логики приложения, например, для определения обработчиков событий элементов управления, нам все равно придется воспользоваться кодом C# [4].



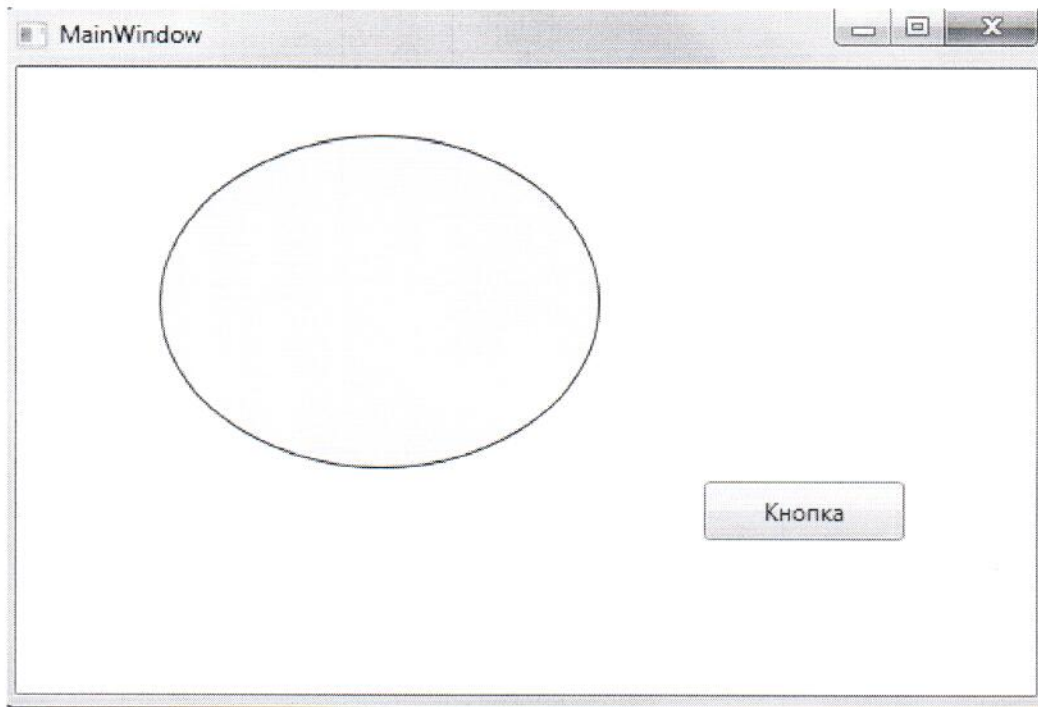


Рис. 1

По умолчанию в разметке окна используется атрибут `x:Class`:

```
Window x:Class="XAML.MainWindow"  
.....
```

Атрибут `x:Class` указывает на класс, который будет представлять данное окно. В данном случае это будет класс `MainWindow`. Этот класс как раз и находится в файле отдельного кода `MainWindow.xaml.cs`, который Visual Studio создает автоматически. По умолчанию он имеет некоторый код:

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
using System.Windows;  
using System.Windows.Controls;  
using System.Windows.Data;  
using System.Windows.Documents;  
using System.Windows.Input;  
using System.Windows.Media;  
using System.Windows.Media.Imaging;  
using System.Windows.Navigation;  
using System.Windows.Shapes;  
  
namespace WpfApplication1  
{  
    /// <summary>  
    /// Логика взаимодействия для MainWindow.xaml  
    /// </summary>  
    public partial class MainWindow : Window
```

```

{
    public MainWindow()
    {
        InitializeComponent();
    }
}

```

При компиляции весь пользовательский интерфейс, определенный в xaml-файле, например, в файле `MainWindow.xaml`, компилируется вместе с классом из отделенного кода в одну сборку. А через метод `InitializeComponent()` класс `MainWindow` вызывает скомпилированный ранее код XAML, разбирает его и по нему строит графический интерфейс окна.

## Компоновка

Чтобы перейти уже непосредственно к созданию красивых интерфейсов и их компонентов, сначала необходимо познакомиться с компоновкой. Компоновка (layout) представляет собой процесс размещения элементов внутри контейнера. Возможно, вы обращали внимание, что одни программы и веб-сайты на разных экранах с разным разрешением выглядят по-разному: где-то лучше, где-то хуже. В большинстве своем такие программы используют жестко закодированные в коде размеры элементов управления. WPF уходит от такого подхода в пользу так называемого "резинового дизайна", где весь процесс позиционирования элементов осуществляется с помощью компоновки.

Благодаря компоновке мы можем удобным нам образом настроить элементы интерфейса, позиционировать их определенным образом. Например, элементы компоновки в WPF позволяют при сжатии или растяжении масштабировать элементы, что очень удобно, а визуально не создает всяких шероховатостей типа незаполненных пустот на форме.

В WPF компоновка осуществляется при помощи специальных контейнеров. Фреймворк предоставляет нам следующие контейнеры: `Grid`, `UniformGrid`, `StackPanel`, `WrapPanel`, `DockPanel` и `Canvas`.

Различные контейнеры могут содержать внутри себя другие контейнеры. Кроме данных контейнеров существует еще ряд элементов, такие как `TabPanel`, которые могут включать другие элементы и даже контейнеры компоновки, однако на саму компоновку не столь влияют в отличие от выше перечисленных. Кроме того, если не хватает стандартных контейнеров, можно определить свои с нужной нам функциональностью.

Контейнеры компоновки позволяют эффективно распределить доступное пространство между элементами, найти для него наиболее предпочтительные размеры.

Все выше перечисленные контейнеры компоновки наследуются от абстрактного класса `Panel`, а само дерево наследования можно представить как показано на рис.2

Процесс компоновки проходит два этапа: измерение (`measure`) и расстановка (`arrange`). На этапе измерения контейнер производит измерение предпочтительного для дочерних элементов места. Однако не всегда контейнер имеет достаточно места, чтобы расставить все элементы по их предпочтительным размерам, поэтому их размеры приходится усекать. Затем происходит этап непосредственной расстановки дочерних элементов внутри контейнера.



Наиболее мощным и часто используемым видом контейнеров служит Grid, напоминающий обычную таблицу. Он содержит столбцы и строки, количество которых задает разработчик. Для определения строк используется свойство RowDefinitions, а для определения столбцов - свойство ColumnDefinitions:

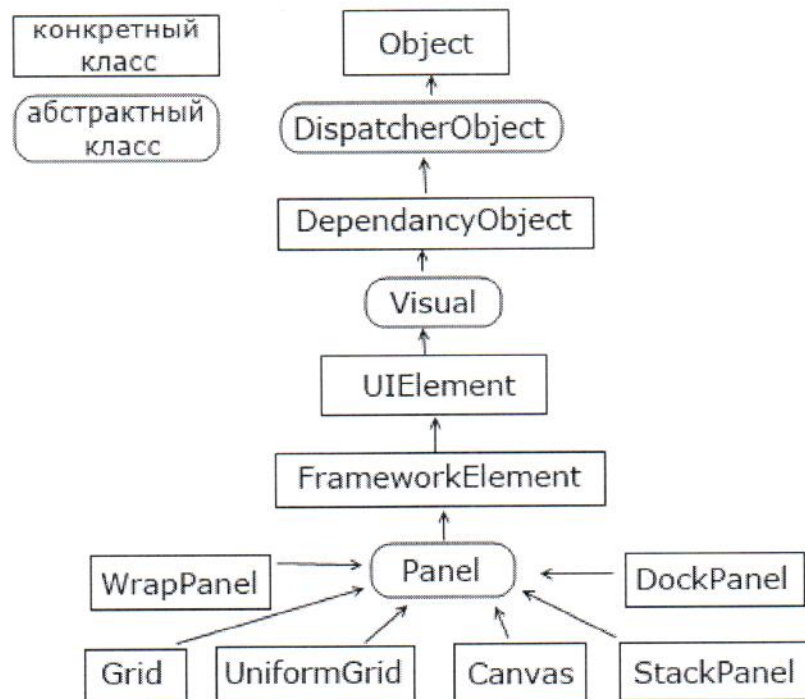


Рис. 2

Пример использования данного вида контейнеров.

```

<Grid.RowDefinitions>
  <RowDefinition></RowDefinition>
  <RowDefinition></RowDefinition>
  <RowDefinition></RowDefinition>
</Grid.RowDefinitions>
<Grid.ColumnDefinitions>
  <ColumnDefinition></ColumnDefinition>
  <ColumnDefinition></ColumnDefinition>
  <ColumnDefinition></ColumnDefinition>
</Grid.ColumnDefinitions>
  
```

Каждая строка задается с помощью вложенного элемента RowDefinition, который имеет открывающий и закрывающий тег. При этом задавать дополнительную информацию необязательно. То есть в данном случае в контейнере определено 3 строки.

Чтобы задать позицию элемента управления с привязкой к определенной ячейке Grid, в разметке элемента нужно прописать значения свойств Grid.Column и Grid.Row, тем самым указывая, в каком столбце и строке будет находиться элемент. Кроме того, если нужно растянуть элемент управления на несколько строк или столбцов, то можно указать свойства Grid.ColumnSpan и Grid.RowSpan, как в следующем примере:

```
Window x:Class="Layout.MainWindow"
```

```

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
Title="MainWindow" Height="300" Width="450">
<Grid ShowGridLines="True">
  <Grid.RowDefinitions>
    <RowDefinition></RowDefinition>
    <RowDefinition></RowDefinition>
    <RowDefinition></RowDefinition>
  </Grid.RowDefinitions>
  <Grid.ColumnDefinitions>
    <ColumnDefinition></ColumnDefinition>
    <ColumnDefinition></ColumnDefinition>
    <ColumnDefinition></ColumnDefinition>
  </Grid.ColumnDefinitions>
  <Button Grid.Column="0" Grid.Row="0" Content="Строка 0 Столбец 0" />
  <Button Grid.Column="0" Grid.Row="1" Content="Объединение трех столбцов"
Grid.ColumnSpan="3" />
  <Button Grid.Column="2" Grid.Row="2" Content="Строка 2 Столбец 2" />
</Grid>
</Window>

```

Результат показан на рис.3.

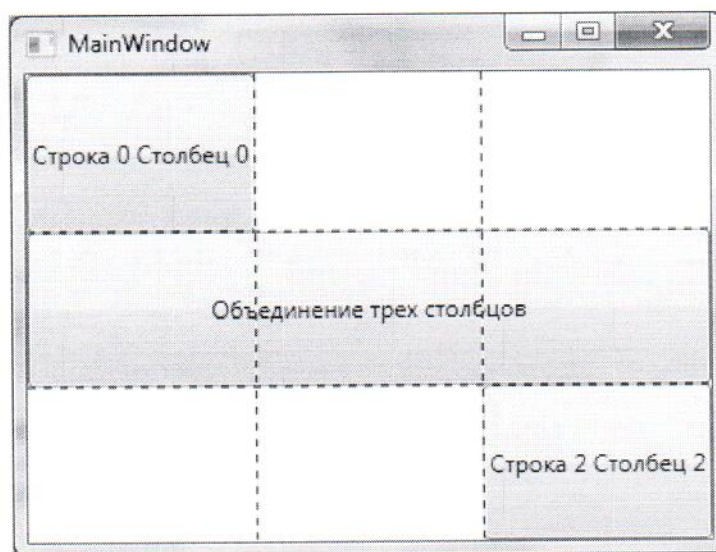


Рис. 3

Имеются и большое число контейнеров других типов.

Аналогичен контейнеру Grid контейнер UniformGrid, только в этом случае все столбцы и строки одинакового размера и используется упрощенный синтаксис для их определения:

```

<UniformGrid Rows="2" Columns="2" Grid.Row="2">
  <Button Content="Left Top" />
  <Button Content="Right Top" />
  <Button Content="Left Bottom" />
  <Button Content="Right Bottom" />
</UniformGrid>

```



Элемент `GridSplitter` представляет собой некоторый разделитель между столбцами или строками, путем сдвига которого можно регулировать ширину столбцов и высоту строк.

```
<Grid>
  <Grid.ColumnDefinitions>
    <ColumnDefinition Width="*" />
    <ColumnDefinition Width="Auto" />
    <ColumnDefinition Width="*" />
  </Grid.ColumnDefinitions>
  <Button Grid.Column="0" Content="Левая кнопка" />
  <GridSplitter Grid.Column="1" ShowsPreview="False" Width="3"
    HorizontalAlignment="Center" VerticalAlignment="Stretch" />
  <Button Grid.Column="2" Content="Правая кнопка" />
</Grid>
```

На рис. 4 показан результат использования приведенной разметки.

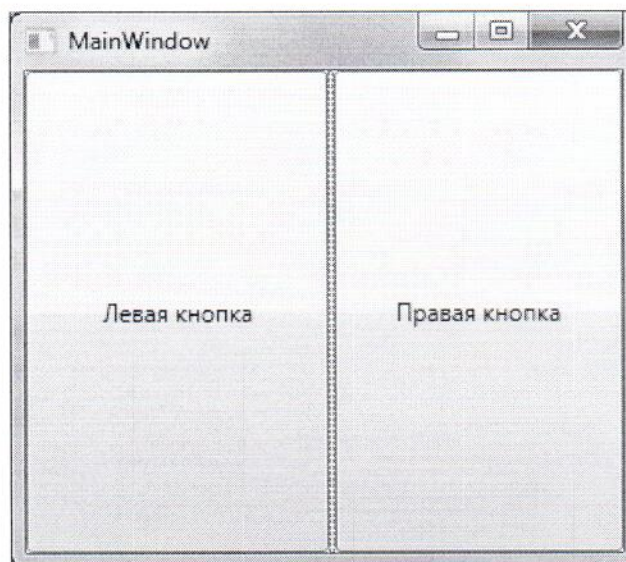


Рис. 4

Двигая центральную линию, разделяющую правую и левую части, можно устанавливать их ширину.

Контейнер `DockPanel` прижимает свое содержимое к определенной стороне внешнего контейнера. Для этого у вложенных элементов надо установить сторону, к которой они будут прижиматься с помощью свойства `DockPanel.Dock`. Например,

```
<Window x:Class="Layout.MainWindow"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  Title="MainWindow" Height="300" Width="300">
  <Grid>
    <DockPanel LastChildFill="True">
      <Button DockPanel.Dock="Top" Background="AliceBlue" Content="Верхняя
кнопка" />
```

```

        <Button DockPanel.Dock="Bottom" Background="Blue" Content="Нижняя
кнопка" />
        <Button DockPanel.Dock="Left" Background="Aquamarine" Content="Левая
кнопка" />
        <Button DockPanel.Dock="Right" Background="DarkGreen" Content="Правая
кнопка" />
        <Button Background="LightGreen" Content="Центр" />
    </DockPanel>
</Grid>
</Window>

```

Размещение объявленных здесь кнопок показано на рис. 5.

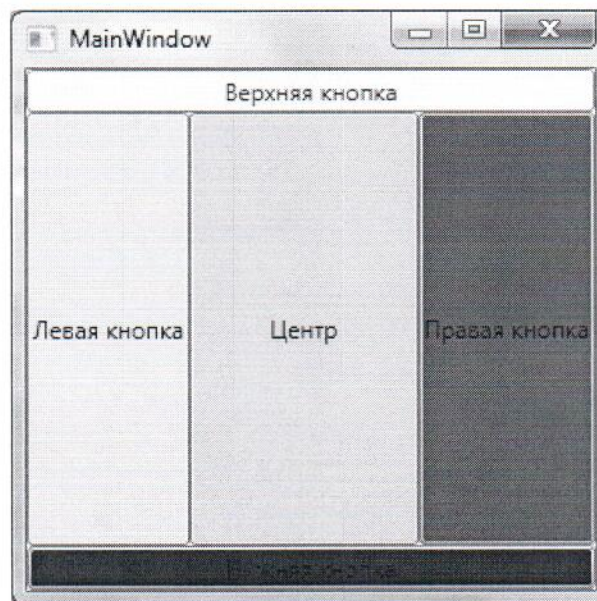


Рис. 5

WrapPanel, подобно StackPanel, располагает все элементы в одной строке или колонке в зависимости от того, какое значение имеет свойство Orientation - Horizontal или Vertical.

Контейнер Canvas является наиболее простым контейнером. Для размещения на нем необходимо указать для элементов точные координаты. Для установки координат элементов используются свойства Canvas.Left, Canvas.Right, Canvas.Bottom, Canvas.Top. Например, свойство Canvas.Left указывает, на сколько единиц от левой стороны контейнера будет находиться элемент, а свойство Canvas.Top - насколько единиц ниже верхней границы контейнера находится элемент.

## Элементы управления

Чтобы как-то взаимодействовать с пользователем, получать от пользователя ввод с клавиатуры или мыши и использовать введенные данные в программе, нам нужны элементы управления. WPF предлагает нам богатый стандартный набор элементов управления

Все элементы управления могут быть условно разделены на несколько подгрупп:

- элементы управления содержимым, например кнопки (Button), метки (Label);



- специальные контейнеры, которые содержат другие элементы, но в отличие от элементов Grid или Canvas не являются контейнерами компоновки - ScrollViewer, GroupBox;
- декораторы, чье предназначение создание определенного фона вокруг вложенных элементов, например, Border или Viewbox;
- элементы управления списками, например, ListBox, ComboBox;
- текстовые элементы управления, например, TextBox, RichTextBox;
- элементы, основанные на диапазонах значений, например, ProgressBar, Slider;
- элементы для работ с датами, например, DatePicker и Calendar;
- меню;
- остальные элементы управления, которые не вошли в предыдущие подгруппы, например, Image.

Все элементы управления наследуются от общего класса System.Window.Control и имеют ряд общих свойств. Рассмотрим некоторые из них.

## События и команды

Чтобы взаимодействовать с пользователем в WPF предусмотрены события. Подключить обработчики событий можно декларативно в файле xaml-кода, а можно стандартным способом в файле отделенного кода.

Декларативное подключение:

```
<Button x:Name="Button1" Height="30" Width="60" Content="Click"
Click="Button_Click" />
```

Обработчик события размещается в файле логики MainWindow.xaml.cs.

```
public partial class MainWindow : Window
{
    public MainWindow()
    {
        InitializeComponent();
        Button1.Click += Button1_Click;
    }
    // обработчик, подключаемый в XAML
    private void Button_Click(object sender, RoutedEventArgs e)
    {
        MessageBox.Show("Hi from Button_Click");
    }
    // обработчик, подключаемый в конструкторе
    private void Button1_Click(object sender, RoutedEventArgs e)
    {
        MessageBox.Show("Hi from Button1_Click");
    }
}
```

Кроме обработки событий интерфейс приложений WPF взаимодействует с пользователем с помощью команд.

В WPF имеются следующие встроенные команды:

- Команды приложения (ApplicationCommands) - класс общих команд приложения (CancelPrint (Отменить печать), Close (Заккрыть), ContextMenu (Конекстное меню), Copy (Копировать), CorrectionList (Список исправлений), Cut (Вырезать), Delete (Удалить), Find (Найти), Help (Справка), New (Создать), Open (Открыть), Paste (Вставить), Prit (Печать), PrintPreview (Предварительный просмотр), Properties (Свойства), Redo (Повторить), Replace (Заменить), Save (Сохранить), SaveAs (Сохранить как), SelectAll (Выделить все), Stop (Остановить), Undo (Отменить) и др.
- Команды навигации (NavigationCommands) - класс команд для навигации по содержимому, например, в браузерных приложениях (BrowseBack (Назад), BrowseForward (Вперед), BrowseHome (Домой), BrowseStop (Остановить), Favorites (Избранное), FirstPage (Первая страница), GoToPage (Переход), LastPage (Последняя страница), NextPage (Следующая страница), PreviousPage (Предыдущая страница), Refresh (Обновить) и др.)
- Команды компонентов интерфейса (ComponentCommands) - используются для перемещения и выделения содержимого элементов управления (MoveDown, MoveLeft, MoveRight, MoveUp, ScrollPageDown, ScrollPageUp, SelectToEnd, SelectToHome и др.
- Команды редактирования (EditingCommands) (AllignKeyCenter (Выравнивание по центру), AllignKeyRight, DecreaseFontSize (Уменьшение высоты шрифта), MoveDownByLine (Переход на строку вниз) и др.)
- Команды для управления мультимедиа (MediaCommands) (DecreaseVolume (Уменьшить громкость), Play (Воспроизвести), Rewind (Перемотка), Record (Запись) и др.

Для создания насыщенных приложений с большими объемами текста со сложным форматированием, перемежающегося с изображениями используются документы.

Все документы в WPF делятся на две группы:

- фиксированные документы (fixed documents). Формат и расположение содержимого таких документов фиксировано и не может быть изменено. На различных устройствах с различным разрешением экрана из содержимое быть выглядеть одинаково и не будет оптимизировано. Такие документы преимущественно предназначены для печати.
- потоковые документы (flow documents). Эти документы предназначены для просмотра на экране монитора. А WPF выполняет оптимизацию документа под конкретные параметры среды.

Потоковые документы в WPF представлены классом FlowDocument, который наследуются от класса FrameworkContentElement и поэтому поддерживают такие механизмы, как привязка, анимация и другие, правда, не используют компоновку. Для использования объекта FlowDocument необходимо поместить его в один из контейнеров - FlowDocumentReader, FlowDocumentPageViewer или FlowDocumentScrollViewer. Пример создания простейшего потокового документа выглядит следующим образом:

```
<FlowDocumentScrollViewer>  
    <FlowDocument>
```



```

    <Paragraph>Hello World!</Paragraph>
    <Paragraph>22.05.1984</Paragraph>
  </FlowDocument>
</FlowDocumentScrollView>

```

## Двумерная графика

Одним из способов построения двумерной графики в окне - это использование фигур. Фигуры фактически являются обычными элементами как например кнопка или текстовое поле. К фигурам относят такие элементы как Polygon, Ellipse, Rectangle (прямоугольник), Line (обычная линия), Polyline - включает несколько линий. Все они наследуются от класса Shape, который определяет ряд общих свойств:

- Fill заполняет фон фигуры с помощью кисти - аналогичен свойству Background у прочих элементов;
- Stroke задает кисть, которая отрисовывает границу фигуры - аналогичен свойству BorderBrush у прочих элементов;
- StrokeThickness задает толщину границы фигуры - аналогичен свойству BorderThickness у прочих элементов;
- StrokeStartLineCap и StrokeEndLineCap задают для незамкнутых фигур (Line) контур в начале и в конце линии соответственно;
- StrokeDashArray задает границу фигуры в виде штриховки, создавая эффект пунктира;
- StrokeDashOffset задает расстояние до начала штриха;
- StrokeDashCap задает форму штрихов;
- Stretch устанавливает способ заполнения внешнего контейнера;
- DefiningGeometry устанавливает объект Geometry для фигуры;
- GeometryTransform применяет преобразования к объекту Geometry.

Как обычные элементы управления они поддерживают традиционные свойства типа Width и Height, а также события, например, Click.

Линия (Line) задается с помощью двух координат по оси X и двух координат по оси Y:

```
<Line Stroke="Red" X1="10" X2="210" Y1="50" Y2="50" />
```

Элемент Polyline задается с помощью нескольких точек:

```
<Polyline Points="10,10 50,50 120,10 10,10" Stroke="Red" />
```

Элемент Polygon задается также как и Polyline с помощью нескольких точек, однако при этом первая и последняя точка соединяются между собой, образуя замкнутую линию:

```
<Polygon Points="10,80 50,50 120,80" Stroke="Red" StrokeThickness="3"
StrokeDashArray="1 1 5" />
```

Элементы Ellipse и Rectangle задаются с помощью свойств Width и Height:

```
<Rectangle Width="60" Height="40" Fill="Beige" />
```

```
<Ellipse Width="30" Height="30" Fill="Red" />
```

Классы Path и Geometry позволяют конструировать более сложные фигуры из простых с помощью теоретико-множественных операций.

## Трехмерная графика

Трехмерная графика в приложении WPF имеет некоторые ключевые моменты:

- окно просмотра (Viewport3D), которое и содержит трехмерную сцену.
- сам объект или геометрия.
- камера, которая устанавливает, как сцена или объект будет отображаться.
- освещение, которое и содержит трехмерную сцену.
- материал, который вместе с освещением определяет внешний вид трехмерного объекта.

Итак, контейнером верхнего уровня для трехмерной сцены является объект Viewport3D. Его можно представлять как окно, ведущее в трехмерное пространство. Важно понимать, что Viewport3D - это такой же объект со всеми свойствами, как и скажем кнопка или текстовое поле, которое мы можем позиционировать на форме как угодно.

Viewport3D имеет свойство Camera, которое принимает объект камеры. Также Viewport3D содержит свойство Children, в котором определяются один или несколько объектов ModelVisual3D. Каждый из этих объектов ModelVisual3D имеет свойство Content, которое принимает в качестве значения один объект GeometryModel3D, либо объект Model3DGroup, который содержит несколько объектов GeometryModel3D.

Далее в объекте GeometryModel3D определяется число треугольников, из которых состоит объект, а также свойство Material, которое управляет материалом объекта, и свойство Geometry, определяющее саму геометрию объекта.

Далее объект Geometry содержит свойство MeshGeometry3D, которое уже через свойства Positions, Normals, TriangleIndices и TextureCoordinates, которые и определяют позиции треугольников, нормали, наложение текстуры.

XAML код будет выглядеть следующим образом:

```
<Window x:Class="Wpf3DApp.MainWindow"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  Title="MainWindow" Height="300" Width="300">
  <Grid>
    <Viewport3D>
      <Viewport3D.Camera>
        <PerspectiveCamera Position="0.5,0.5,3.5" LookDirection="0,0,-3.5" />
      </Viewport3D.Camera>
      <Viewport3D.Children>
        <ModelVisual3D>
          <ModelVisual3D.Content>
            <DirectionalLight Color="White" Direction="-1,-1,-2" />
          </ModelVisual3D.Content>
        </ModelVisual3D>
        <ModelVisual3D>
          <ModelVisual3D.Content>
```



```

    <GeometryModel3D>
      <GeometryModel3D.Geometry>
        <MeshGeometry3D Positions="0,0,0 1,0,0 0,1,0 1,1,0
0,0,1 1,0,1 0,1,1 1,1,1"
          TriangleIndices="0,2,1 1,2,3 0,4,2 2,4,6
          0,1,4 1,5,4 1,7,5 1,3,7
          4,5,6 7,6,5 2,6,3 3,6,7"/>
      </GeometryModel3D.Geometry>
      <GeometryModel3D.Material>
        <DiffuseMaterial Brush="Blue" />
      </GeometryModel3D.Material>
    </GeometryModel3D>
  </ModelVisual3D.Content>
  <ModelVisual3D.Transform>
    <RotateTransform3D>
      <RotateTransform3D.Rotation>
        <AxisAngleRotation3D x:Name="rotate" Axis="0 1 0" />
      </RotateTransform3D.Rotation>
    </RotateTransform3D>
  </ModelVisual3D.Transform>
</ModelVisual3D>
</Viewport3D.Children>
</Viewport3D>
  <Slider Height="25" VerticalAlignment="Top" Minimum="0" Maximum="360"
Value="{Binding ElementName=rotate, Path= Angle}" />
</Grid>
</Window>

```

На рис. 6 показан результат применения данного XAML-кода.

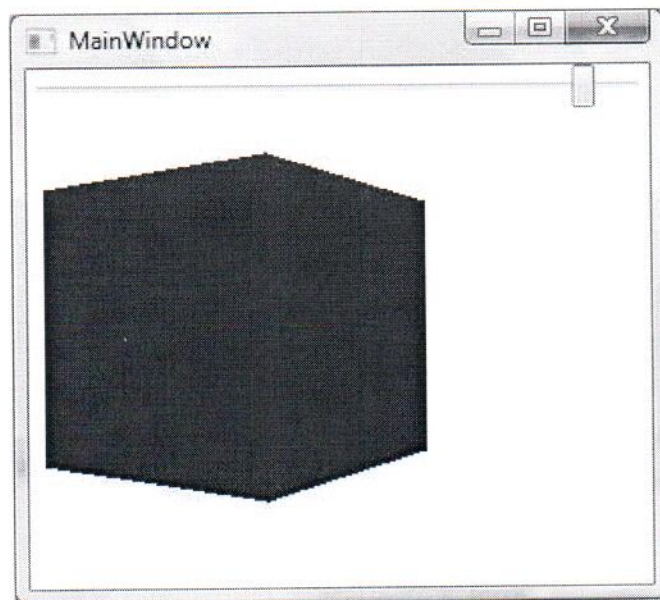


Рис. 6

## Заключение

В данной работе кратко изложена суть языка XAML, его назначение, синтаксис и основные объекты. Освещены вопросы использования XAML для проектирования интерфейса приложений, включающих в том числе и мультимедиа-компоненты, в рамках WPF-проектов.

В рабочей среде Visual Studio код XAML можно создавать, перетаскивая элементы управления в разметку. Поэтому на первый взгляд может показаться, что изучать синтаксис XAML тоже нет необходимости. Конечно, это не так. Понимание синтаксиса XAML критически важно для создания приложений как WPF- так и Silverlight-проектов. Только зная XAML, можно понять ключевые концепции и создавать правильную разметку. Кроме того, редактировать многие компоненты разметки XAML и решать многие задачи (в том числе подключение обработчиков к событиям, определение ресурсов, создание шаблонов элементов управления, связывание данных, создание анимации) можно только вручную.

## Библиографический список

1. *Байдачный С.С.* Silverlight 4: Создание насыщенных Web-приложений – М.: СОЛОН-ПРЕСС, 2010. – 288 с. – ISBN 978-5-91359-079-4
2. *Мак-Дональд, Мэтью* Silverlight 5 с примерами на C# для профессионалов, 4-е изд. : Пер. с англ. – М. : ООО “И.Д. Вильямс”, 2013. – 848 с. : ил.
3. Общие сведения о языке XAML. – Режим доступа: [https://msdn.microsoft.com/ru-ru/library/ms752059\(v=vs.110\).asp](https://msdn.microsoft.com/ru-ru/library/ms752059(v=vs.110).asp)
4. XAML. – Режим доступа: <http://metanit.com/sharp/wpf/2.php>



Протокол экспертизы соответствия уровня достижения студентов \_\_\_\_\_ запланированных результатов обучения  
(фамилия, И.О.)

по дисциплине «Компьютерные технологии мультимедиа»

Перечень результатов обучения	Структурные элементы заданий по дисциплине						
	Реферат	Подготовка к лабораторным занятиям	Подготовка к практическим занятиям	Курсовой проект	Тестирование №1	Тестирование №2	Вопросы к экзамену
ОК-2, способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности	+	+			Вопросы к тестированию		Вопросы к экзамену
ОК-4, использование на практике умений и навыков в организации исследовательских и проектных работ, в управлении коллективом	+	+					+
ПК-2, на основе знания педагогических приемов принятие непосредственного участия в учебной работе кафедр и других учебных подразделений по профилю направления «Информатика и вычислительная техника»	+	+					+
ПК-6, применение современных технологий разработки программных комплексов с использованием CASE-средств, контроль качества разрабатываемых программных продуктов	+	+					+
ПК-7, применение перспективных методов исследования и решения профессиональных задач на основе знания мировых тенденций развития вычислительной техники и информационных технологий	+	+					+

Оценки по пятибалльной шкале выставляются в ячейках, соответствующих компетенциям (по строке), подлежащим оцениванию по результатам конкретного элемента задания по дисциплине (по столбцам) в соответствии с запланированными в рабочей программе видами СРС и ответами на вопросы во время зачета.

Преподаватель \_\_\_\_\_ Пугачев А. И. « \_\_\_\_\_ » 20 \_\_\_\_ г.  
(подпись)