

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ

«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ»

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(СамГТУ)

УТВЕРЖДАЮ

Проректор по УР
СамГТУ

Деморецкий Д.А.
2015 м.п.



УТВЕРЖДАЮ

Проректор по УМР
Университета ИТМО

Шехонин А.А.
2015 м.п.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.В ДВ.5.2 Компьютерные технологии мультимедиа

(указывается шифр и наименование дисциплины по учебному плану)

Направление подготовки 09.04.01. «Информатика и вычислительная техника»

Квалификация (степень) выпускника Магистр

Магистерская программа «Программное обеспечение интеллектуальных систем и технологий»

Форма обучения очная

Выпускающая кафедра Прикладного программирования и технологических инноваций Университета ИТМО, Вычислительная техника СамГТУ

Кафедра-разработчик рабочей программы Вычислительная техника СамГТУ
(название)

Семестр	Трудоем- кость час.	Лек- ций, час.	Практич. занятий, час.	Лаборат. работ, час.	СРС, час.	Форма промежуточного контроля (экз./зачет)
2	144	17		34	93	экз.
Итого	144	17		34	93	экз.

Санкт-Петербург
Самара
2015 г.

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 26.12.2014г.

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


(подпись)

А.И.Пугачев
(ФИО)

28.01.2015г.
(дата)

Рабочая программа утверждена на заседании кафедры:
«Вычислительная техника» 30.01.2015 г. протокол № 11
(наименование кафедры- (дата и номер протокола)
разработчика)

Зав. кафедрой-разработчиком


(подпись)

С.П.Орлов
(ФИО)

30.01.2015г.
(дата)

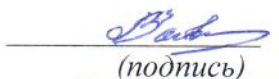
Эксперт методической комиссии
по УГНП


(подпись)

Н.В.Ефимушкина
(ФИО)

30.01.2015г.
(дата)

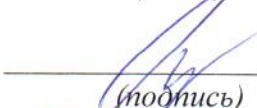
Председатель методического
совета факультета АИТ
(на котором осуществляется обучение)


(подпись)

В.В.Зайвый
(ФИО)

30.01.2015г.
(дата)

Декан факультета АИТ
(на котором осуществляется обучение)


(подпись)

Н.Г.Губанов
(ФИО)

02.02.2015г.
(дата)

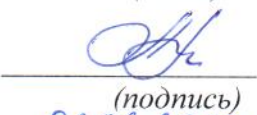
СОГЛАСОВАНО:
Зав. выпускающей кафедрой


(подпись)

С.П.Орлов
(ФИО)

30.01.2015г.
(дата)

Начальник УВО


(подпись)

А.Н.Лукьянова
(ФИО)

02.02.2015г.
(дата)

СОДЕРЖАНИЕ

1. Планируемые результаты обучения.....	4
2. Место дисциплины в структуре ОПОП	5
3. Структура и содержание дисциплины.....	7
3.1. Структура дисциплины.....	7
3.2. Содержание дисциплины	8
4. Учебно-методическое обеспечение для самостоятельной работы.....	11
5. Образовательные технологии.....	11
6. Формы контроля освоения дисциплины	11
7. Основная, дополнительная и учебно-методическая литература.....	13
8. Ресурсы информационно-телекоммуникационной сети Интернет.....	14
9. Информационные технологии.....	14
10. Материально-техническое обеспечение дисциплины	14
Дополнения и изменения к рабочей программе.....	16
Приложение 1. Аннотация рабочей программы	17
Приложение 2. Методические указания для самостоятельной работы обучающихся	18
Приложение 3. Фонд оценочных средств дисциплины	25
Приложение 4. Методические указания для обучающихся по освоению дисциплины	45

1. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Планируемые результаты освоения основной профессиональной образовательной программы (ОПОП) – компетенции обучающихся определяются требованиями стандарта по направлению подготовки 09.04.01 «Информатика и вычислительная техника» и формируются в соответствии с матрицей компетенций ОПОП.

Планируемые результаты обучения по дисциплине – знания, умения, навыки и (или) опыт деятельности, характеризующие этапы формирования компетенций и обеспечивающие достижение планируемых результатов освоения образовательной программы, формируются в соответствии с картами компетенций ОПОП.

Таблица 1

Планируемые результаты обучения по дисциплине

Шифр компетенции	Наименование компетенции	Планируемые результаты обучения по дисциплине
Общие компетенции		
ОК-6	Способность проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности	<p>Знать: методы организации работ по обеспечению надежности вычислительной техники и разработке интеллектуальных систем З (ПК-6) III</p> <p>Уметь: применять навыки организации работ по обеспечению надежности вычислительной техники и разработке интеллектуальных систем У (ПК-6) III</p> <p>Владеть: навыками организации работ по обеспечению надежности вычислительной техники и разработке интеллектуальных систем В (ПК-6) III</p>
Общепрофессиональные компетенции		
ОПК-5	Владение методами и средствами получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях	<p>Знать: методы средства получения, хранения и переработки информации с помощью современных компьютерных технологий З1 (ОПК-5) III</p> <p>Уметь: получать, хранить и перерабатывать информацию с помощью современных компьютерных технологий У1 (ОПК-5) III</p> <p>Владеть: методами и средствами получения, хранения и переработки информации с помощью современных компьютерных технологий В1 (ОПК-5) III</p>
ОПК-6	Способность анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и	<p>Знать: методы расчета и анализа характеристик надежности ВС, обеспечения высокой надежности аппаратных и программных компонентов, а также методы поиска неисправностей, планирования и выполнения профилактических работ в ВС З1 (ПК-7) I</p>

Шифр компетенции	Наименование компетенции	Планируемые результаты обучения по дисциплине
	рекомендациями	Уметь: проектировать вычислительные системы с требуемой надежностью, формировать требования к их надежности в технических заданиях на проектирование У1 (ПК-7) I Владеть: навыками работы с технической документацией на современные распределенные ВС, методами организации работы и руководства коллективами их разработчиков и обслуживающего персонала В1 (ПК-7) I
Профессиональные компетенции		
ПК-7	Применение перспективных методов исследования и решения профессиональных задач на основе знания мировых тенденций развития вычислительной техники и информационных технологий	Знать: мировые тенденции повышения надёжности средств вычислительной техники З1 (ПК-7) I Уметь: реализовывать эти тенденции в каждом конкретном случае У1 (ПК-7) I, Владеть: необходимыми навыками построения распределённых вычислительных систем требуемой надёжности В1 (ПК-7) I

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Компьютерные технологии мультимедиа» относится к вариативной части блока 1 учебного плана.

В таблице 2 приведены предшествующие и последующие дисциплины, направленные на формирование компетенций дисциплины в соответствии с матрицей компетенций ОПОП.

Таблица 2.

Предшествующие и последующие дисциплины, направленные на формирование компетенций

№ п/п	Наименование компетенции	Предшествующие дисциплины	Последующие дисциплины
Общие компетенции			
1	ОК-6, способность проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности	Интеллектуальные системы и базы знаний, Надежность распределенных вычислительных систем	Интеллектуальные системы и базы знаний, Надежность распределенных вычислительных систем, Математическая статистика
Общепрофессиональные компетенции			
2	ОПК-5, владение	Технология	Теоретическая

№ п/п	Наименование компетенции	Предшествующие дисциплины	Последующие дисциплины
	методами и средствами получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях	программирования, Учебная практика, Теоретическая информатика, Теория проектирования систем, Надежность распределенных вычислительных систем, Учебная практика,	информатика, Теория проектирования систем, Надежность распределенных вычислительных систем, Медицинские информационные системы, Интеллектуальные транспортные системы, Производственная практика, Преддипломная практика
3	ОПК-6, способность анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями	Вычислительные системы, Технология программирования, Иностранный язык для научных публикаций, Надежность распределенных вычислительных систем	Надежность распределенных вычислительных систем, Математическая статистика, Медицинские информационные системы, Интеллектуальные транспортные системы, Научно-исследовательская работа, Государственная итоговая аттестация
Профессиональные компетенции			
4	ПК-7, применение перспективных методов исследования и решения профессиональных задач на основе знания мировых тенденций развития вычислительной техники и информационных технологий	Вычислительные системы Теория проектирования систем, Управление проектами, Интеллектуальные системы и базы знаний, Надежность распределенных вычислительных систем	Теория проектирования систем, Управление проектами, Интеллектуальные системы и базы знаний, Информационные технологии в медицине, Информационные технологии транспортных систем, Надежность распределенных вычислительных систем, , Государственная итоговая аттестация

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

3.1 СТРУКТУРА ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 4 зачетных единиц, 144 часа.

Таблица 3

Трудоемкость дисциплины и виды учебной работы

Вид учебной работы	Всего часов	Семестр
		2
Аудиторные занятия (всего)	51	51
В том числе: лекции	17	17
лабораторные работы (ЛР)	34	34
Самостоятельная работа (всего)	93	93
В том числе: контактная внеаудиторная работа	4	4
самостоятельное изучение теоретического материала (подготовка к ЛР)	27	27
оформление и подготовка к отчету по ЛР	17	17
подготовка к экзамену	45	45
ИТОГО:	144	144
час.	4	4
з.е.	4	4

Распределение учебной нагрузки по разделам дисциплины

Таблица 4

№ 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	6		2	4	12
2.	Основы XAML MS SilverLight	4		4	5	13
3.	Оснащение приложений мультимедиа-средствами	7		28	35	70
	Контактная внеаудиторная работа	-	-	-	4	4
	Подготовка к экзамену	-	-	-	45	45
	ИТОГО:	17		34	93	144

3.2 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Таблица 5

Лекции

№ лекции	Номер раздела	Тема лекции и перечень дидактических единиц	Трудоемкость, часов
	1	МУЛЬТИМЕДИА В ИНТЕРАКТИВНЫХ ПРИЛОЖЕНИЯХ	
1		Тема 1.1 Технологии мультимедиа	
		1.1.1 <i>Введение</i> . Дисциплина «Компьютерные технологии мультимедиа», его задачи и место в подготовке магистров.	2
		1.1.2 <i>Современные технологии мультимедиа</i> . HTML, JavaScript, Flash, MS Silverlight.	
2		Тема 1.2 Знакомство с MS Silverlight	
		1.2.1 <i>Назначение и основы применения MS Silverlight</i> . Назначение MS Silverlight, состав, характеристики.	2
		1.2.2 <i>Использование MS Silverlight</i> . Взаимодействие с пользователем. Архитектура MS Silverlight. MS Silverlight и язык разметки XAML. Использование MS Silverlight с MS Expression Studio.	
3		Тема 1.3 Использование MS Silverlight с MS Visual Studio	
		1.3.1 <i>Установка инструментов Silverlight для Visual Studio</i> . Подключение MS Silverlight к проекту Visual Studio.	2
		1.3.2 <i>Использование MS Silverlight</i> . Взаимодействие с пользователем. Объекты MS Silverlight.	
	2	ОСНОВЫ XAML MS SILVERLIGHT	
4		Тема 2.1 XAML – язык описания интерфейса	
		2.1.1 <i>Основные конструкции</i> . Связь XAML с XML. Теги. Синтаксис тегов. Иерархия.	2
		2.1.2 <i>Графические ресурсы и объекты</i> . Элемент управления Canvas. Цвета, кисти. Фигуры в XAML.	
5		Тема 2.2 Элементы управления в XAML	
		2.2.1 <i>Элементы компоновки</i> . Элемент управления StackPanel. Элемент управления Grid.	2
		2.2.2 <i>Базовые элементы управления</i> . Класс Control. Кнопки. Текстовые элементы управления. Элементы на основе диапазона значений.	
	3	ОСНАЩЕНИЕ ПРИЛОЖЕНИЙ МУЛЬТИМЕДИА-СРЕДСТВАМИ	
6		Тема 3.1 Геометрические объекты и графика	
		3.1.1 <i>Кисти XAML</i> . SolidColorBrush, LinearGradientBrush, RadialGradientBrush. Системные цвета.	2
		3.1.2 <i>Фигуры в XAML</i> . Line, Rectangle, Ellipse. Использование геометрических объектов.	
7		Тема 3.2 Трансформация и анимация	
		3.2.1 <i>Трансформация</i> . Основные виды трансформации. Плоско-параллельное перемещение с помощью TranslateTransform/ Вращение с помощью RotateTransform. Масштабирование с помощью ScaleTransform. Совмещение преобразований.	2

№ лекции	Номер раздела	Тема лекции и перечень дидактических единиц	Трудоемкость, часов
		3.2.2 <u>Анимация</u> . Основные виды анимации. Запуск анимации. Параметры анимации. Анимация с помощью ключевых кадров.	
8		Тема 3.3 Работа с аудио и видео	
		3.3.1 <u>Работа с аудио</u> . Автоматическое воспроизведение звука. Управление воспроизведением звука.	2
		3.3.2 <u>Работа с видео</u> . Управление текущим состоянием видео. Управление положением курсора воспроизведения. Использование маркеров временной шкалы мультимедиа.	
9		Тема 3.4 Ресурсы и стили приложений	
		3.4.1 <u>Ресурсы</u> . Ресурсы приложения. Ресурсы объектов. Выделение ресурсов объектов в отдельные файлы.	1
		3.4.2 <u>Стили</u> . Понятие стиля. Создание стиля. Шаблоны стилей.	
Итого:			17

Практические занятия

Практические занятия учебным планом не предусмотрены.

Таблица 6

Лабораторные работы

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

№ лаб. раб.	Номер раздела	Наименование лабораторной работы и перечень дидактических единиц	Трудоемкость, часов
		Использование маркеров временной шкалы мультимедиа.	
9	3	Создание стиля приложения Создание стиля. Шаблоны стилей.	4
Итого:			34

Содержание отчетов о каждой лабораторной работе, конкретные задания на разработку программ приведены в методических указаниях к ним.

Таблица 7

Самостоятельная работа студента

Раздел дисциплины	Под-раздел	Вид самостоятельной работы студента (СРС) и перечень дидактических единиц	Трудо-емкость, час.
Разделы 1, 2	1.1, 2.2	Подготовка к лабораторной работе № 1 Подключение среды Silverlight к MS Visual Studio. Разработка программного кода приложения.	3
	1.1	Оформление отчета по лабораторной работе № 1	1
Раздел 3	2.2	Подготовка к лабораторной работе № 2 Элементы управления Silverlight Класс Control. Кнопки. Текстовые элементы управления. Элементы на основе диапазона значений.	3
	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.	3
	3.2	Оформление отчета по лабораторной работе № 3	2
	3.2	Подготовка к лабораторной работе № 4 Трансформация Плоско-параллельное перемещение, масштабирование, вращение, совмещение преобразований	3
	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.	3
	3.2	Оформление отчета по лабораторной работе № 5	2
	3.3	Подготовка к лабораторной работе № 6 Воспроизведение аудио Автоматическое воспроизведение звука. Управление воспроизведением звука.	3
	3.3	Оформление отчета по лабораторной работе № 6	2
	3.3	Подготовка к лабораторной работе № 7 Воспроизведение видео Управление текущим состоянием видео.	3
3.3	Оформление отчета по лабораторной работе № 7	2	

Раздел дисциплины	Под-раздел	Вид самостоятельной работы студента (СРС) и перечень дидактических единиц	Трудо-емкость, час.
	3.3	Подготовка к лабораторной работе № 8 Создание индикатора загрузки и воспроизведения видео Управление положением курсора воспроизведения.	3
	3.3	Оформление отчета по лабораторной работе № 8	2
	3.4	Подготовка к лабораторной работе № 9 Создание стиля приложения Создание стиля. Шаблоны стилей.	3
	3.4	Оформление отчета по лабораторной работе № 9	2
Контактная внеаудиторная работа			4
Подготовка к экзамену			45
Итого:			93

4. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

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. - ISBN 978-5-91359-079-4

Методические указания в т.ч. для самостоятельной работы обучающихся и методические указания для обучающихся по освоению дисциплины приводятся в Приложениях к рабочей программе

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Таблица 8

Интерактивные образовательные технологии, используемые в аудиторных занятиях

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

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

- *выполнение лабораторных работ;*
- *защита лабораторных работ.*

Промежуточный контроль по результатам семестра по дисциплине проходит в форме письменного экзамена, который включает в себя ответ на теоретические вопросы.

Фонд оценочных средств, перечень заданий для проведения промежуточной аттестации, а также методические указания для проведения промежуточной аттестации приводятся в Приложении 3.

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. ОСНОВНАЯ, ДОПОЛНИТЕЛЬНАЯ И УЧЕБНО-МЕТОДИЧЕСКАЯ ЛИТЕРАТУРА

Таблица 9

Учебно-методическое обеспечение

№ п/п	Учебник, учебное пособие (приводится библиографическое описание учебника, учебного пособия)	Ресурс НТБ СамГТУ	Кол-во экз.
Основная литература			
1	Крапивенко А.В. Технологии мультимедиа и восприятие ощущений [Электронный ресурс]: учебное пособие/ Крапивенко А.В.— Электрон. текстовые данные.— М.: БИНОМ. Лаборатория знаний, 2012.— 271 с.— Режим доступа: http://www.iprbookshop.ru/6475 .	ЭБС «IPRbooks»	
2	Стативко Р.У. Информационные технологии [Электронный ресурс]: учебное пособие/ Стативко Р.У., Рыбакова А.И.— Электрон. текстовые данные.— Белгород: Белгородский государственный технологический университет им. В.Г. Шухова, ЭБС АСВ, 2012.— 168 с.— Режим доступа: http://www.iprbookshop.ru/28346 .	ЭБС «IPRbooks»	
Дополнительная литература			
1	Дьяконов В.П. Новые информационные технологии [Электронный ресурс]: учебное пособие/ Дьяконов В.П.— Электрон. текстовые данные.— М.: СОЛОН-ПРЕСС, 2008.— 640 с.— Режим доступа: http://www.iprbookshop.ru/8663 .	ЭБС «IPRbooks»	
2	Тузовский А.Ф. Проектирование и разработка WEB-приложений [Электронный ресурс]: учебное пособие/ Тузовский А.Ф.— Электрон. текстовые данные.— Томск: Томский политехнический университет, 2014.— 219 с.— Режим доступа: http://www.iprbookshop.ru/34702 .	ЭБС «IPRbooks»	
3	Бердышев С.Н. Искусство оформления сайта.- 2-е изд. [Электронный ресурс]: практическое пособие/ Бердышев С.Н.— Электрон. текстовые данные.— М.: Дашков и К, Ай Пи Эр Медиа, 2012.— 101 с.— Режим доступа: http://www.iprbookshop.ru/5968 .	ЭБС «IPRbooks»	
4	Аббасов И.Б. Основы графического дизайна на компьютере в Photoshop CS6 [Электронный ресурс]: учебное пособие/ Аббасов И.Б.— Электрон. текстовые данные.— М.: ДМК Пресс, 2013.— 238 с.— Режим доступа: http://www.iprbookshop.ru/29256 .	ЭБС «IPRbooks»	
5	Байдачный С.С. Silverlight 4 [Электронный ресурс]: создание насыщенных Web-приложений/ Байдачный С.С.— Электрон. текстовые данные.— М.: СОЛОН-ПРЕСС, 2010.— 288 с.— Режим доступа: http://www.iprbookshop.ru/20851 .	ЭБС «IPRbooks»	

Учебно-методическая литература			
1	Компьютерные технологии мультимедиа: методические указания к лабораторным работам / Сост. А.И. Пугачев. Самара: Самар. гос. техн. ун-т, 2015. – 49 с.: ил.	ЭБС СамГТУ	НТБ

8. РЕСУРСЫ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ»

Российские

1. Электронная библиотека диссертаций РГБ (Просмотр полных текстов диссертаций возможен только с компьютеров, установленных в научно-библиографическом отделе НТБ СамГТУ)
2. ВИНИТИ
3. eLIBRARY.RU (НЭБ - Научная электронная библиотека)
4. Сетевой журнал «Компьютерная графика и мультимедиа» <http://cgm.computergraphics.ru/>
5. Уроки программирования на OpenGL <http://pmg.org.ru/nehe/>

Зарубежные

6. ScienceDirect (Elsevier) - естественные науки, техника, медицина и общественные науки.
7. Scopus - база данных рефератов и цитирования
8. УИС РОССИЯ - Университетская информационная система РОССИЯ - <http://www.cir.ru/index.jsp>
9. Библиотека компьютерной литературы. - <http://it.eup.ru/>
10. Электронная библиотека РФФИ. - <http://www.rfbr.ru/rffi/ru/>
11. Сайт Microsoft SilverLight Toolkint <http://silverlight.codeplex.com/>
12. Сайт компании Microsoft <http://silverlight.net/showcase/>

9. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

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

10. 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).

ОДОБРЕНА на заседании методической комиссии по УГС " ____ " _____ 20__ г."

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

_____	_____	_____	_____
<i>шифр</i>	<i>наименование</i>	<i>личная подпись</i>	<i>расшифровка подписи</i>

СОГЛАСОВАНО:

Заведующий выпускающей кафедрой _____
наименование кафедры личная подпись расшифровка подписи дата

Декан _____
наименование факультета, где производится обучение, личная подпись расшифровка подписи дата

Начальник УВО _____
личная подпись расшифровка подписи дата

**Аннотация рабочей программы
по дисциплине «Компьютерные технологии мультимедиа»**

направление 09.04.01 «Информатика и вычислительная техника» профиль
«Программное обеспечение интеллектуальных систем и технологий»

Дисциплина «Компьютерные технологии мультимедиа» относится к вариативной части блока 1 учебного плана профессионального блока дисциплин магистерской подготовки по направлению 09.04.01 «Информатика и вычислительная техника». Дисциплина реализуется кафедрой «Вычислительная техника» на факультете автоматизации 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-6, способность проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности;
- ОПК-5, владение методами и средствами получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях
- ОПК-6, способность анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями;
- ПК-7, применение перспективных методов исследования и решения профессиональных задач на основе знания мировых тенденций развития вычислительной техники и информационных технологий.

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 (17 часов), лабораторные работы (34 часа), самостоятельная работа (93 часа).

Методические указания для самостоятельной работы обучающихся

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 существу 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 существу 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	Разработать приложение в Visual Studio с	0,2

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
формирования умений	использованием Silverlight согласно методическим указаниям.	

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 систематизации знаний	<i>Работа с конспектом лекций:</i> изучить лекцию № 7 , тема 3.3.2 Анимация. Ответить на контрольные вопросы по лабораторной работе № 5 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 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

Вид заданий	Содержание задания на самостоятельную работу	Время на СР, час
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 C# для профессионалов, 4-е изд. : Пер. с англ. - М. : ООО «И.Д. Вильямс», 2013. - 848 с. : ил. - ISBN 978-5-8459-1784-3.

Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

«Самарский государственный технический университет»

Факультет автоматике и информационных технологий

Кафедра «Вычислительная техника»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

текущего контроля и промежуточной аттестации

дисциплины: **Б1.В.ДВ.5.2 Компьютерные технологии мультимедиа**

в составе основной образовательной программы по направлению подготовки
(специальности): 09.04.01, «Информатика и вычислительная техника»

направленность (профиль) программы: «Программное обеспечение
интеллектуальных систем и технологий»

уровень высшего образования: магистратура

Разработчик ФОС

«__» _____ 20__ г. _____
(подпись)

Пугачев А.И.
(Ф.И.О.)

Заведующий кафедрой «Вычислительная техника»

«__» _____ 20__ г. _____
(подпись)

Орлов С.П.
(Ф.И.О.)

Самара 2015

Паспорт фонда оценочных средств

по дисциплине «Компьютерные технологии мультимедиа»

№ п/п	Этапы формирования компетенции	Код и наименование формируемой компетенции	Наименование оценочного средства
1	Раздел 1. Мультимедиа в дизайне интерактивных приложений	ОК-6, способность проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности У (ОК-6) III, В (ОК-6) III, З (ОК-6) III	проверка рефератов, экзаменационные билеты
2	Раздел 2. Основы XAML MS SilverLight	ОПК-5, владение методами и средствами получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях У1 (ОПК-5) III, В1 (ОПК-5) III, З1 (ОПК-5) III ОПК – 6: способность анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями У1 (ОПК-6) II, В1 (ОПК-6) II, З1 (ОПК-6) II	проверка рефератов, экзаменационные билеты
3	Раздел 3. Оснащение приложений мультимедиа-средствами	ПК-7: применение перспективных методов исследования и решения профессиональных задач на основе знания мировых тенденций развития вычислительной техники и информационных технологий У1 (ПК-7) I, В1 (ПК-7) I, З1 (ПК-7) I	проверка рефератов, экзаменационные билеты

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 _____ Пугачев А. И.
(подпись)

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-6; ОПК-5; ОПК-6, ПК-7.

Разработчик _____ Пугачев А. И.

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="OK"/>  
</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].

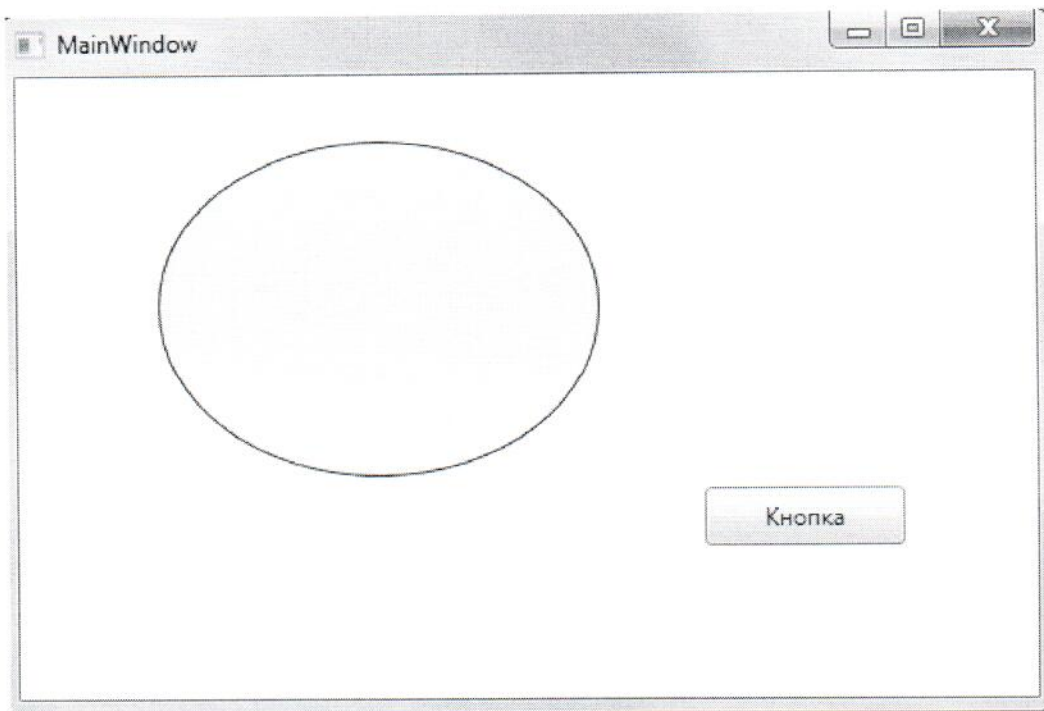


Рис. 1

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:

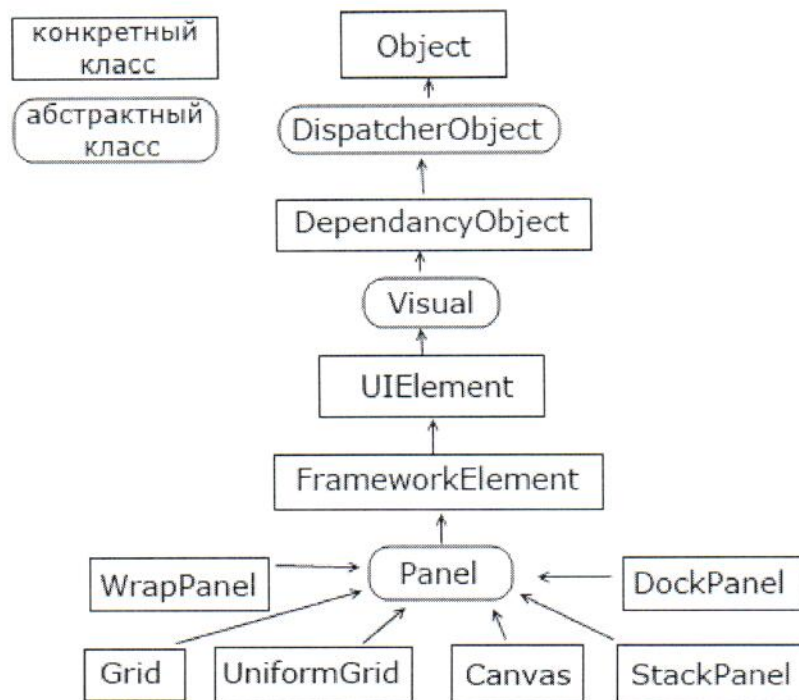


Рис. 2

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.

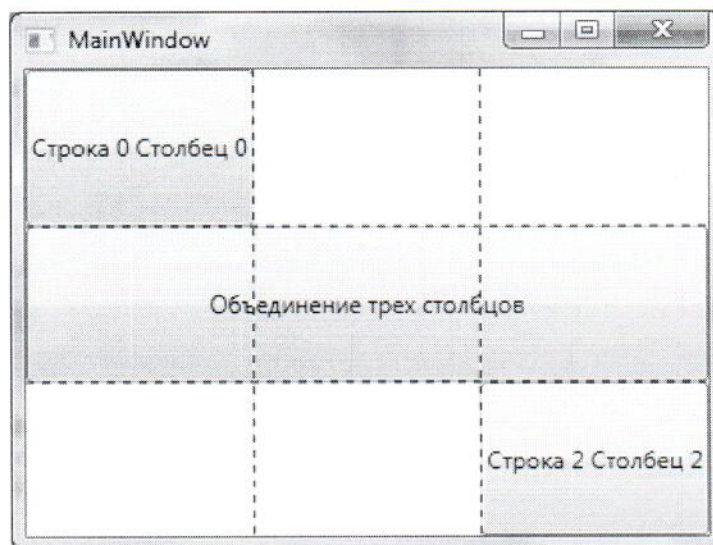


Рис. 3

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.

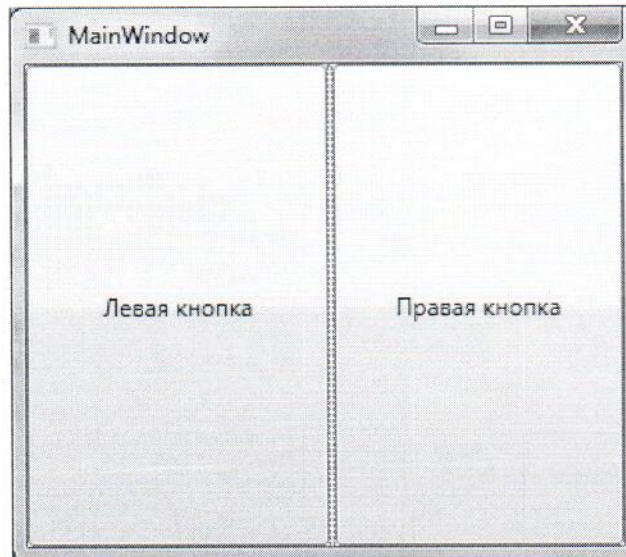


Рис. 4

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" />
    </DockPanel>
  </Grid>
```

```

<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.

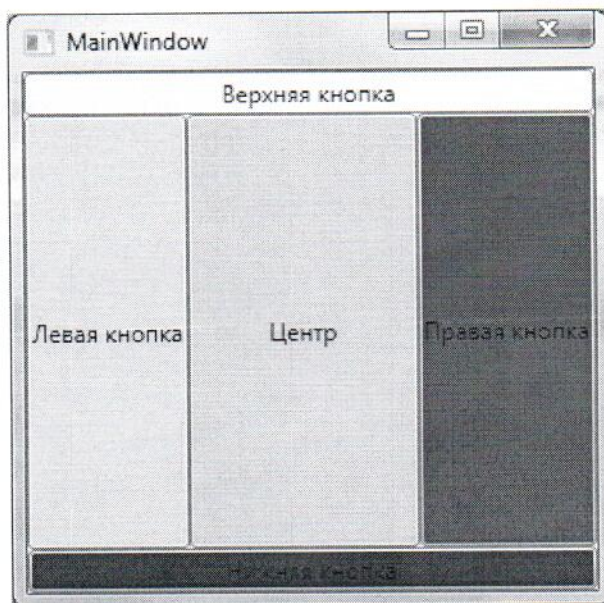


Рис. 5

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 - ScrollView, 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 (Закреть), ContextMenu (Контекстное меню), Copy (Копировать), CorrectionList (Список исправлений), Cut (Вырезать), Delete (Удалить), Find (Найти), Help (Справка), New (Создать), Open (Открыть), Paste (Вставить), Print (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) (AlignCenter (Выравнивание по центру), AlignRight, 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 будет 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>
```



```

0,0,1 1,0,1 0,1,1 1,1,1"
    <MeshGeometry3D Positions="0,0,0 1,0,0 0,1,0 1,1,0
    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.

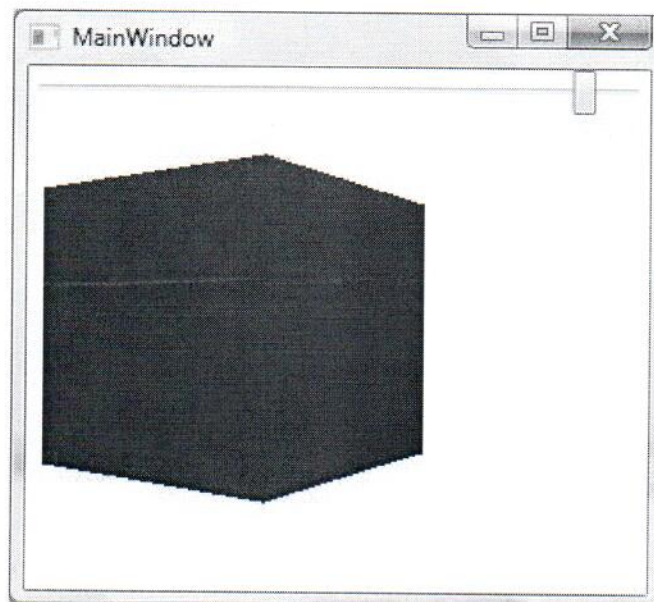


Рис. 6

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.

**Матрица соответствия оценочных средств запланированным результатам обучения
по дисциплине «Компьютерные технологии мультимедиа»**

		Структурные элементы заданий по дисциплине			
		Отчет по лабораторным работам	Реферат	Вопрос № 1	Вопрос № 2
Перечень результатов обучения					
<p>ОК-6 Способность проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности.</p> <p>ОПК-5 Владение методами и средствами получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях.</p> <p>ОПК-6 Способность анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями.</p> <p>ПК-7 Применение перспективных методов исследования и решения профессиональных задач на основе знания мировых тенденций развития вычислительной техники и информационных технологий.</p>	Виды СРС, предусмотренные рабочей программой дисциплины	Вопросы по реферату	Вопросы к экзамену		
	У (ПК-6) III В (ПК-6) III	3 (ПК-6) III	3 (ПК-6) III		
	У1 (ОПК-5) III В1 (ОПК-5) III	31 (ОПК-5) III		31 (ОПК-5) III	
	У1 (ОПК-6) II В1 (ОПК-6) II	31 (ОПК-6) II	31 (ОПК-6) II		31 (ОПК-6) II
	У1 (ПК-7) I В1 (ПК-7) I	31 (ПК-7) I		31 (ПК-7) I	

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 экзамена.

Критерии оценки достижений студентов запланированных результатов освоения дисциплины в ходе текущего контроля и промежуточной аттестации

Итоговая оценка промежуточной аттестации в форме экзамена выставляется по пятибалльной шкале по результатам этапов освоения целевых компетенций в ходе изучения дисциплины с учетом критериев оценки уровней достижения запланированных результатов обучения в соответствии с картами компетенций ОПОП, матрицей соответствия оценочных средств запланированным результатам обучения:

- «отлично»: более 70 % дескрипторов компетенций в соответствии с установленными картами компетенций уровнями их сформированности оцениваются на уровне «5»; оценки на уровне ниже «4» отсутствуют»;
- «хорошо»: более 60 % дескрипторов оцениваются на уровне «4» и/или «5»;
- «удовлетворительно»: 50 % дескрипторов оцениваются на уровне «3» и ниже.

Неудовлетворительная аттестация приравнивается к академической задолженности.

Оценка	Обобщенная характеристика результатов изучения дисциплины
«отлично»	Студент показал прочные знания основных положений фактического материала, умение самостоятельно решать конкретные практические задачи повышенной сложности, свободно использовать справочную литературу, делать обоснованные выводы из результатов расчетов
«хорошо»	Студент показал прочные знания основных положений фактического материала, умение самостоятельно решать конкретные практические задачи, предусмотренные рабочей программой, ориентироваться в рекомендованной справочной литературе, умеет правильно оценить полученные результаты расчетов
«удовлетворительно»	Студент показал знание основных положений фактического материала, умение получить с помощью преподавателя правильное решение конкретной практической задачи из числа предусмотренных рабочей программой, знакомство с рекомендованной справочной литературой
«неудовлетворительно»	При ответе студента выявились существенные пробелы в знаниях основных положений фактического материала, неумение с помощью преподавателя получить правильное решение конкретной практической задачи из числа предусмотренных рабочей программой учебной дисциплины

Преподаватель _____ А. И. Пугачев _____ « _____ » _____ 20 _____ г.

Методические указания для обучающихся по освоению дисциплины

Вид учебных занятий	Организация деятельности студента
Лекция	<p>При написании конспекта лекций необходимо кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометать важные мысли, выделять ключевые слова, термины. Обозначить вопросы, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на лабораторных занятиях.</p> <p>Уделить внимание следующим понятиям:</p> <ul style="list-style-type: none"> – язык разметки XAML; – синтаксис тэгов; – элементы управления StackPanel, Grid; – графические инструменты и графические объекты; – трансформация и анимация графических объектов; – управление аудио- и видео-ресурсами.
Индивидуальные задания	<p>Знакомство с основной и дополнительной литературой, включая справочные издания, зарубежные источники, конспект основных положений, терминов, сведений, требующих для запоминания и являющихся основополагающими в этой теме. Составление аннотаций к прочитанным литературным источникам и др.</p>
Реферат	<p><i>Реферат:</i> Поиск литературы и составление библиографии, использование от 3 до 5 научных работ, изложение мнения авторов и своего суждения по выбранному вопросу; изложение основных аспектов проблемы.</p> <p>Изучение научной, учебной, нормативной и другой литературы. Отбор необходимого материала; формирование выводов и разработка конкретных рекомендаций по анализируемой теме.</p> <p>Ознакомиться со структурой и оформлением реферата.</p>
Лабораторная работа	<p>Методические указания по выполнению лабораторных работ:</p> <p>Компьютерные технологии мультимедиа [электронный ресурс]: методические указания к лабораторным работам / Сост. А.И. Пугачев. Самара: Самар. гос. техн. ун-т, 2015</p>
Подготовка к экзамену (зачету)	<p>При подготовке к экзамену необходимо ориентироваться на конспекты лекций, рекомендуемую литературу и др.</p>