

Негосударственное аккредитованное некоммерческое
частное образовательное учреждение высшего образования
«АКАДЕМИЯ МАРКЕТИНГА И СОЦИАЛЬНО-ИНФОРМАЦИОННЫХ
ТЕХНОЛОГИЙ – ИМСИТ»
(г. Краснодар)

Кафедра математики и вычислительной техники

Рассмотрено и одобрено на заседании
кафедры Математики и вычислительной
техники Академии ИМСИТ, протокол
№8 от 19 марта 2018 года,
зав. кафедрой



Н.С. Нестерова

УТВЕРЖДЕНО

Научно-методическим советом академии
протокол №8 от 16 апреля 2018 года

Председатель ИМС,
проректор по учебной работе,
профессор



Н.Н. Павелко

Б1.В.08

ПРОЕКТИРОВАНИЕ И АРХИТЕКТУРА ПРОГРАММНЫХ СИСТЕМ

АННОТАЦИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

по направлению подготовки

09.03.04 Программная инженерия

Направленность (профиль) программы:

«Информационно-вычислительные системы»

Квалификация

Бакалавр

Краснодар
2018

<p>Цель изучения дисциплины:</p>	<p>Целью освоения дисциплины «Проектирование и архитектура программных систем» являются получение теоретических знаний о принципах, технологии, методах и средствах проектирования архитектуры программных систем, а также приобретение практических навыков в выполнении действий по различным фазам создания программных продуктов.</p> <p>Задачи курса:</p> <p>Приобретение знаний в области современных архитектур программных систем, технологий и парадигм их проектирования.</p>
<p>Краткая характеристика учебной дисциплины (основные блоки, темы)</p>	<ol style="list-style-type: none"> 1. Введение в дисциплину. Программные системы (ПС) как отрасль экономики. Проблемы создания ПС. Кризис программирования. Становление и развитие программной инженерии. Развитие технологий программирования. CASE-технологии. 2. Архитектуры программных систем. Понятие архитектуры программной системы. Что определяет и на что влияет архитектура. Архитектурные структуры и представления. Модульные структуры. Структуры “компонент и соединитель”. Структуры распределения. Отношения между структурами. Варианты архитектур программных систем. Архитектура, основанная на уровнях абстракций. Архитектуры, основанные на портах. Архитектуры независимых компонентов. Архитектуры, основанные на потоках данных. 3. Жизненный цикл программных систем (ПС). Понятие жизненного цикла ПС. Основные процессы ЖЦ ПС. Вспомогательные процессы ЖЦ ПС. Организационные процессы ЖЦ ПС. Взаимосвязь между процессами ЖЦ ПС. Состав и стадии жизненного цикла ПС. Модели ЖЦ ПС. Каскадная модель (классический жизненный цикл). Итерационная модель ЖЦ ПС. Макетирование. Стратегии конструирования ПС. Инкрементная модель. Спиральная модель. Спиральная модель ЖЦ ПС “Рациональный унифицированный процесс” (RUP). 4. Постановка требований к ПС. Модель проектирования ПС как последовательная трансляция требований, предъявляемых к системе. Методология решения задач проектирования по Г. Майерсу. Уровни требований к программным системам. Определение требований к программным системам. Определение требований к программным системам. Функциональные и нефункциональные (эксплуатационные) требования. Процесс определения целей продукта и проекта. Разработка технического задания. 5. Анализ требований и разработка внешних спецификаций. Разработка предварительного внешнего проекта. Процесс внешнего проектирования. Проектирование взаимодействия с пользователем.

	<p>Подготовка внешних спецификаций. Проверка правильности внешних спецификаций. Планирование изменений спецификаций. Анализ требований и определение спецификаций при структурном подходе к проектированию. Метод функционального моделирования. Функциональные диаграммы. Диаграммы потоков данных. Диаграммы переходов состояний. Анализ требований и определение спецификаций при объектном подходе. Анализ требований и определение спецификаций при объектном подходе к проектированию.</p> <p>6. Проектирование архитектуры программных систем. Методология проектирования. Методы проектирования компонентных и модульных архитектур программных систем. Структурное проектирование. Модульность и ее характеристики. Оценка сложности модульных иерархических структур. Слои программного продукта. Метод восходящей разработки (“снизу-вверх”). Метод нисходящей разработки (“сверху-вниз”). Замечания по структурному проектированию. Формальное описание методики разработки модульной архитектуры программных систем. Пример проектирования структуры программной системы. Проектирование и программирование модулей. Проектирование программных систем при объектном подходе. Рефакторинг архитектуры программных систем. Паттерны системного проектирования</p>
<p>Компетенции, формируемые в результате освоения учебной дисциплины:</p>	<p>ОПК – 3 готовностью применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов</p> <p>ПК – 5 владением стандартами и моделями жизненного цикла</p> <p>ПК -14 готовностью обосновать принимаемые проектные решения, осуществлять постановку и выполнение экспериментов по проверке их корректности и эффективности</p> <p>ПК -16 способностью формализовать предметную область программного проекта и разработать спецификации для компонентов программного продукта</p> <p>ПК -17 способностью выполнить начальную оценку степени трудности, рисков, затрат и сформировать рабочий график</p>
<p>Наименования дисциплин, необходимых для освоения данной учебной дисциплины</p>	<p>– Информатика.</p>
<p>Знания, умения и навыки, получаемые в процессе изучения дисциплины:</p>	<p>Знать:</p> <ul style="list-style-type: none"> – современные архитектуры программных систем; – представления и модели жизненного цикла программных систем; – методы, технологии и средства разработки

	<p>архитектуры сложных программных систем.</p> <p>Уметь:</p> <ul style="list-style-type: none"> – решать задачи, возникающие на различных фазах жизненного цикла программных систем, связанных с проектированием архитектуры программных систем. <p>Владеть:</p> <ul style="list-style-type: none"> – навыками использования современных CASE-средств, используемых на различных фазах проектирования архитектуры программных систем.
Формы проведения занятий, образовательные технологии	<p>Лекционные занятия: проблемные и интерактивные лекции, лекция – визуализация, лекция-беседа, лекция-дискуссия, лекция с запланированными ошибками.</p> <p>Практические занятия: самостоятельная разработка архитектуры программной системы, написание кода программной системы.</p>
Используемые инструментальные и программные средства:	<p>Компьютерный класс с установленным программным обеспечением, средства проекции (презентации).</p> <p><u>ВАРИАНТ</u></p> <p>Программное обеспечение: среда разработки программного обеспечения Microsoft Visual Studio, сайт кафедры М и ВТ, учебные программы и методические указания в электронном виде, электронные учебники.</p> <p>Данная дисциплина обеспечена: компьютерными классами, электронным курсом лекций, необходимым оборудованием для лекций.</p>
Формы промежуточного контроля:	<p>Текущие оценки знаний, тестирование, доклады, контрольные работы, рефераты</p>
Форма итогового контроля знаний:	<p>Зачёт, экзамен.</p>