

## СИЛЛАБУС

Название и код дисциплины	Проектирование ПО II		Учебный год, семестр	2021-2022 уч. год, 4 семестр
Трудоемкость курса	<b>4 кредита</b> Всего -120 ч.	Структура занятий	Лабораторные занятия -64 ч. СРС-56 ч.	
Данные о преподавателе	Мусина Индира Рафиковна, доц. кафедры "Программное обеспечение компьютерных систем" КГТУ им. И. Раззакова, г.Бишкек, 56-38-53(кафедра), +996 550 00 58 98; E-mail: Musina-indira@yandex.ru			
Цель и задачи дисциплины	<p>Целями дисциплины являются закрепление и расширение будущими программными инженерами знаний об этапах жизненного цикла создаваемого программного обеспечения и подходах к созданию ПО, полученных при обучении дисциплины «Проектирование ПО I»; изучение объектно-ориентированного подхода к проектированию ПО.</p> <p>Основные задачи дисциплины:</p> <ul style="list-style-type: none"> <li>- Изучение языка моделирования UML;</li> <li>- Приобретение практических навыков по анализу и формирований требований, а также проектированию ПО с помощью графических языков программирования (UML).</li> </ul>			
Описание курса	<p>Дисциплина «Проектирование ПО II» относится к обязательным дисциплинам вариативной части профессионального цикла. Она читается в весеннем семестре студентам второго курса.</p> <p>Предметом изучения дисциплины являются теоретические и практические вопросы проектирования сложных программных систем с использованием объектно-ориентированного подхода на основе применения диаграмм унифицированного языка UML. Кроме того, в рамках данного курса изучаются модели жизненного цикла ПО, вопросы анализа и формирования требований.</p> <p>По итогам изучения дисциплины студенты сдают экзамен. Текущий контроль и самоконтроль усвоения курса осуществляется посредством выполнения лабораторных и самостоятельных работ. Рубежный контроль – письменная работа (модуль I) и командная разработка ПО, включающая разработку требований и проектирование (модуль II)</p>			
Пререквизиты	«Информатика»; «Проектирование ПО I»;	Постреквизиты	Процессы проектирования и разработки ПО	
Краткое содержание дисциплины	Процессы программной инженерии. Модели жизненного цикла. Анализ и формирование требований.. Разработка ТЗ. Объектно-ориентированный подход к анализу, проектированию и кодированию. Язык UML. Виды диаграмм UML. Статические диаграммы (классов, пакетов, компонентов). Динамические диаграммы (Use Case, деятельности, последовательности, состояний). Связи в UML. Проектирование БД. Образцы проектирования. Программная реализация проекта.			
Основная литература	<ol style="list-style-type: none"> <li>1. А. Орлов., Б.Я.Цилькер. «Технологии разработки программного обеспечения». Учебник. - СПб: Питер, 2016.</li> <li>2. Йан Соммервилл «Инженерия программного обеспечения». Издат. Дом Москва, 2002г. (или на англ. яз. Изд- е от 2016г.)</li> <li>3. Мусина И.Р., Семенов А.С. Процессы проектирования и разработки программного обеспечения: Учебно – методическое пособие /КГТУ им. И.Раззакова, Бишкек: ИЦ «Техник», 2016.-64с.</li> </ol>			
Дополнительная литература	<ol style="list-style-type: none"> <li>1. Микки Госсе, Брайан Келлер, Аджой Кришнамурти и т.д. «Управление жизненным циклом приложений с Visual Studio 2010. - 2012.</li> <li>2. Мартин Р. Чистая архитектура. Искусство разработки программного обеспечения. — СПб.: Питер, 2018. — 352 с.: ил. — (Серия «Библиотека программиста»). ISBN 978-5-4461-0772-8/</li> <li>3. Л.Г. Гагарина, Е.В. Кокорева, Б.Д.Виснадул. Технология разработки программного обеспечения. Москва, 2008.</li> <li>4. М. М. Меженная, Т. В. Гордейчук, М. М. Борисик, О. С. Медведев, И.Ф. Киринович. Тестирование, оценка программного обеспечения. Учебно-методическое пособие.— Минск: БГУИР, 2016. – 64 с. : ил.</li> </ol>			
Информация по оценке	<p>Сумма баллов: 61-73 удовлетворительно; 74-86 хорошо; 87-100 отлично</p> <p>Успешность изучения дисциплины в системе кредитных технологий оценивается суммой набранных баллов (из 100 возможных):</p>			

Политика выставления баллов	<ol style="list-style-type: none"> <li>1. Выполнение лабораторных работ: 20-30 баллов.</li> <li>2. Выполнение самостоятельной работы студента: 20-30 баллов.</li> <li>3. Рубежный контроль – по 4 балла на каждый модуль.</li> <li>4. Итоговый контроль (бланочное или on-line тестирование, устный опрос): 0-32 баллов.</li> </ol> Штрафные баллы: <ol style="list-style-type: none"> <li>1. Несвоевременная сдача лаб. работ - -2 балла за каждую работу,</li> <li>2. Пропуск занятия без уважительной причины, т.е 1 нб = -1балл ,</li> <li>3. Плагиат - -30 баллов.</li> </ol>
Политика курса	Перечень правил, которых должен придерживаться студент: <ol style="list-style-type: none"> <li>a) Обязательная защита лабораторных и самостоятельных работ</li> <li>b) Обязательное присутствие экзамене.</li> </ol> Недопустимо: <ol style="list-style-type: none"> <li>a) Плагиат</li> <li>d) Несвоевременная сдача заданий без уважительной причины.</li> </ol>
Права студента	Права студента в случае несогласия с действиями, оценкой преподавателя: возможность подать на апелляцию.

### Лабораторные занятия

№	Наименование работ	Содержание	Кол-во час.	Примечание
<b>Модуль 1</b>				
1	Разработка ТЗ на программный продукт	<ol style="list-style-type: none"> <li>1. Проанализировать автоматизируемый объект;</li> <li>2. Определить бизнес-требования;</li> <li>3. Определить функциональные и нефункциональные требования.</li> </ol>	4	Показать документ, подготовленный в WORD
2	Построение диаграммы USE CASE.	<ol style="list-style-type: none"> <li>1. Выделить варианты использования и действующие лица.</li> <li>2. Установить связи между вариантами использования.</li> </ol>	4	Использовать одно из CASE-средств (MS Visio, он-лайн UMLредактор)
3	Построение диаграммы USE CASE со связями обобщения между актерами и между вариантами использования	В диаграмме USE CASE установить связи обобщения между актерами и между вариантами использования	4	Использовать одно из CASE-средств (MS Visio, он-лайн UMLредактор)
4	Описание вариантов использования	Описать каждый вариант использования из диаграммы использования (UCM)	4	Использовать шаблон для описания
5	Алгоритмическое проектирование. Построение диаграммы деятельности	Построить диаграммы деятельности для каждого варианта использования	4	Использовать одно из CASE-средств
6	Построение диаграммы деятельности для параллельных процессов	Построить диаграммы с распараллеливанием и слиянием процессов	4	Использовать одно из CASE-средств (MS Visio, он-лайн UMLредактор)
7	Построение диаграммы деятельности с дорожками	Построить диаграммы деятельности с распределением деятельностей по различным дорожкам (модулям и актерам)	4	Использовать одно из CASE-средств (MS Visio, он-лайн UMLредактор)
8	Модульная работа: разработка ТЗ и проектирование для контрольного примера	<ol style="list-style-type: none"> <li>1. Разработать ТЗ</li> <li>2. Построить диаграммы: вариантов использования и деятельности.</li> </ol>	4	Использовать одно из CASE-средств (MS Visio, он-лайн UMLредактор)

<b>Модуль 2</b>				
9	Проектирование архитектуры. Организация классов. Построение диаграммы классов. Атрибуты, методы, связи	Конструирование классов. Определение атрибутов и методов. Определение отношений между классами: ассоциация, зависимость, агрегация, композиция, обобщение	4	Использовать одно из CASE-средств
10	Построение диаграммы последовательности	Построение взаимосвязи между объектами классов через передачу сообщений	4	Использовать одно из CASE-средств
11	Построение диаграммы пакетов.	Объединение классов в пакеты. Установление связи между пакетами	4	Использовать одно из CASE-средств
12	Построение диаграммы состояний	Определение состояний, переходов, триггеров. Построение диаграммы состояний.	4	Использовать одно из CASE-средств
13	Проектирование БД	Спроектировать БД. Построение логической и физической модели БД.	4	Использовать любое средство моделирования БД (Visio, All fusion и т.д.)
14	Программная реализация проекта	Провести кодирования, используя любой язык программирования.	4	Использовать любое средство для кодирования
15	Тестирование программы. Разработка документации	Тестирование разработанного программного продукта методом черного ящика. Отладка. Разработка документации на продукт	4	Использовать любое средство для кодирования
16	Защита проекта (по самостоятельной работе)	Подготовить отчет по самостоятельной работе с демонстрацией программы	4	Отчет должен быть подготовлен в виде текстового документа.
<b>ИТОГО</b>			<b>64</b>	

### Самостоятельная работа

№	Темы занятий	Задания на СРС	Цель и содержание заданий	Реком. литература	Форма контроля	Срок и сдачи	Макс балл
<b>Модуль 1</b>							
1	Модели жизненного цикла ПО (4 час.)	Рассмотреть различные виды ЖЦ ПО, выявить их достоинства и недостатки	Студент должен знать в каких случаях применять ту или иную модель ЖЦ ПО	[1]	Отчет по работе.	2-3 недели	2
2	Agile (гибкая) разработка ПО (4 час.)	Описать особенности гибкой разработки	Студент должен знать в каких условиях и как применять гибкую разработку ПО	[1,2]	Отчет по работе.	2-3 недели	2
3	Разработка технического задания для индивидуального проекта (6 час.)	Разработать ТЗ для индивидуального проекта, согласованного с преподавателем. Построить UCM.	Студент должен уметь самостоятельно анализировать и выстраивать требования для проекта по разработке ПО	[6]	Отчет с предоставлением UCM в среде для построения UML диаграмм	4-5 недели	3

4	Документирование вариантов использования для индивидуального проекта (8 час)	Провести описание вариантов использования по шаблону в виде таблицы	Студент должен уметь самостоятельно правильно документировать варианты использования ПО	[7]	Отчет	6-7 недель	4
5	Построение диаграммы деятельности для индивидуального проекта (6 час).	Построить диаграммы деятельности с использованием CASE-технологий	Умение самостоятельно применить CASE-технологии для алгоритмического проектирования	[3, 6]	Отчет с предоставлением диаграмм	6-8 недели	4
<b>Итого по модулю I (28ч.)</b>							<b>15</b>
<b>Модуль 2</b>							
1	Построение диаграммы классов для индивидуального проекта (4 час)	Построить диаграмму классов с использованием CASE-технологий	Умение самостоятельно применить CASE-технологии для проектирования классов для реального объекта	[3, 7]	Отчет с предоставлением диаграмм	9-10 недели	3
2	Проектирование архитектуры ПО для индивидуального проекта (4 час)	Построение диаграммы пакетов с использованием CASE-технологий	Умение самостоятельно применить CASE-технологии для проектирования архитектуры ПО реального объекта	[3,2,7]	Отчет с предоставлением диаграмм	11-12 недели	3
3	Разработка модели данных для индивидуального проекта (4 час)	Построение диаграммы ERD для разрабатываемого ПО с использованием CASE-технологий	Умение самостоятельно анализировать данные и моделировать БД для реального объекта	[3,7]	Отчет с предоставлением диаграмм	13-14 недели	2
4	Построение диаграмм последовательности и состояний для индивидуального проекта (6 час)	Построить диаграммы последовательности и состояний с использованием CASE-технологий	Студент должен уметь самостоятельно строить диаграммы последовательности и состояний для реального объекта	[1,2]	Отчет с предоставлением диаграмм	15-16 недели	3
5	Программная реализация для индивидуального проекта (10 час)	Реализовать проект в виде работающего программного продукта	Студент должен уметь самостоятельно реализовать проект и провести его валидацию	[1,2]	Полный отчет в рукописном виде	15-16 недели	4
<b>Итого по модулю II:28 час.</b>							<b>15</b>