

**Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Национальный исследовательский университет «МЭИ»**

Направление подготовки/специальность: 01.03.02 Прикладная математика и информатика

**Наименование образовательной программы: Математическое и программное обеспечение
вычислительных машин и компьютерных сетей**

Уровень образования: высшее образование - бакалавриат

Форма обучения: Очная

**Оценочные материалы
по дисциплине
Ведение крупных программных проектов**

**Москва
2022**

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ РАЗРАБОТАЛ:

Преподаватель

(должность)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Куриленко И.Е.
	Идентификатор	R73df8d6c-KurilenkoIY-5c331b90

(подпись)

И.Е.

Куриленко

(расшифровка
подписи)

СОГЛАСОВАНО:

Руководитель
образовательной
программы

(должность, ученая степень,
ученое звание)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Маран М.М.
	Идентификатор	R7be141f2-MaranMM-804b01e2

(подпись)

М.М. Маран

(расшифровка
подписи)

Заведующий
выпускающей кафедры

(должность, ученая степень,
ученое звание)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Варшавский П.Р.
	Идентификатор	R9a563c96-VarshavskyPR-efb4bbd

(подпись)

П.Р.

Варшавский

(расшифровка
подписи)

ОБЩАЯ ЧАСТЬ

Оценочные материалы по дисциплине предназначены для оценки: достижения обучающимися запланированных результатов обучения по дисциплине, этапа формирования запланированных компетенций и уровня освоения дисциплины.

Оценочные материалы по дисциплине включают оценочные средства для проведения мероприятий текущего контроля успеваемости и промежуточной аттестации.

Формируемые у обучающегося компетенции:

- ПК-1 Способен выполнять все этапы жизненного цикла программного обеспечения
ИД-1 Демонстрирует знания современных CASE-средств для выполнения этапов жизненного цикла ПО
ИД-3 Формирует архитектуру ПО
- ПК-2 Способен проектировать и реализовать базы данных, разработать интерфейсы пользователя к базам данных
ИД-3 Выбирает СУБД на основе их характеристики
ИД-4 Формирует и реализует базы данных в СУБД и организовать программный доступ к данным

и включает:

для текущего контроля успеваемости:

Форма реализации: Компьютерное задание

1. Автоматизация документирования программного продукта (Лабораторная работа)
2. Автоматизация сборки программного проекта. Применение системы компонентной сборки. ()
3. Использование систем календарного планирования. (Лабораторная работа)
4. Использование систем управления задачами и заявками (Лабораторная работа)
5. Построение структурной декомпозиции работ и диаграммы Гантта (Лабораторная работа)
6. Практическое применение систем контроля версий (Лабораторная работа)
7. Практическое применение систем контроля качества кода на основе метрик на примере Microsoft Visual Studio. (Лабораторная работа)
8. Применение средства контроля качества кода на базе правил (Лабораторная работа)
9. Разработка шаблонов для генерации кода (Лабораторная работа)

Форма реализации: Проверка задания

1. Расчет показателей качества кода (Реферат)
2. Расчет сетевого графика (Контрольная работа)

БРС дисциплины

8 семестр

Раздел дисциплины	Веса контрольных мероприятий, %											
	Индекс КМ:	КМ -1	КМ -2	КМ -3	КМ -4	КМ -5	КМ -6	КМ -7	КМ -8	КМ -9	КМ -10	КМ -11
	Срок КМ:	1	2	3	4	5	6	7	8	10	12	15
Управление крупным проектом												

по разработке программного обеспечения												
Методы сетевого планирования	+	+	+	+								
Гибкое планирование	+	+	+	+								
Техника снижения трудоемкости крупного проекта по разработке программного обеспечения												
Системы контроля версий и конфигурационного управления.						+				+		
Автоматизация сборки.							+	+				
Генерация кода по шаблону.						+	+			+		
Контроль работоспособности кода.									+		+	+
Контроль качества.									+		+	+
Автоматизация тестирования.									+			
Автоматическое документирование и средства корпоративной памяти.											+	+
Вес КМ:	10	8	8	8	8	8	10	10	10	10	10	10

\$Общая часть/Для промежуточной аттестации\$

СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ТЕКУЩЕГО КОНТРОЛЯ

I. Оценочные средства для оценки запланированных результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Индекс компетенции	Индикатор	Запланированные результаты обучения по дисциплине	Контрольная точка
ПК-1	ИД-1 _{ПК-1} Демонстрирует знания современных CASE-средств для выполнения этапов жизненного цикла ПО	Знать: современные методологии разработки программного обеспечения Уметь: применять современные CASE-средства	Построение структурной декомпозиции работ и диаграммы Ганта (Лабораторная работа) Расчет сетевого графика (Контрольная работа) Использование систем управления задачами и заявками (Лабораторная работа) Использование систем календарного планирования. (Лабораторная работа) Практическое применение систем контроля версий (Лабораторная работа) Автоматизация документирования программного продукта (Лабораторная работа)
ПК-1	ИД-3 _{ПК-1} Формирует архитектуру ПО	Знать: современные архитектурные концепции, применяемые при разработке программных систем Уметь: применять типовые архитектурные решения	Разработка шаблонов для генерации кода (Лабораторная работа) Автоматизация сборки программного проекта. Применение системы компонентной сборки.
ПК-2	ИД-3 _{ПК-2} Выбирает СУБД на основе их характеристики	Знать: современную технику планирования работ в проекте по разработке программного обеспечения	Практическое применение систем контроля качества кода на основе метрик на примере Microsoft Visual Studio. (Лабораторная работа) Применение средства контроля качества кода на базе правил (Лабораторная работа)

		<p>Уметь: анализировать архитектуру программных систем, оценивать уместность применения той или иной разновидности решения и технологии доступа к данным</p>	
ПК-2	ИД-4 _{ПК-2} Формирует и реализует базы данных в СУБД и организовать программный доступ к данным	<p>Знать: технику снижения трудоемкости крупного проекта по разработке программного обеспечения</p> <p>Уметь: создавать и внедрять стандарты кодирования и проектирования</p>	<p>Расчет показателей качества кода (Реферат) Практическое применение систем контроля качества кода на основе метрик на примере Microsoft Visual Studio. (Лабораторная работа)</p>

II. Содержание оценочных средств. Шкала и критерии оценивания

КМ-1. Построение структурной декомпозиции работ и диаграммы Ганта

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Проводится в форме выполнения и защиты лабораторной работы.

Краткое содержание задания:

Цель работы

Научиться разрабатывать иерархическую структуру работ (WBS).

Подготовка к работе

Ознакомиться с основными техниками и инструментами сетевого планирования;

Скачать и установить систему планирования WBS Chart Pro (опционально);

Ознакомиться с целью и назначением диаграмм Ганта;

Ознакомиться с принципом построения диаграмм Ганта;

Ознакомиться с целью и назначением PERT-диаграмм;

Установить и ознакомиться с возможностями системы Microsoft Project.

Порядок выполнения работы

Разбить проект на этапы. Построить первый уровень WBS

Для каждого этапа составить список работ. Внести эти работы на диаграмму

Для каждой недекомпозированной комплексной работы в WBS осуществить декомпозицию (Декомпозицию работ следует проводить до уровня на котором могут быть оценены сроки выполнения работ и требуемые ресурсы).

Оценить сроки выполнения работ

Создать новый проект в Microsoft Project.

Добавить в проект сотрудников и их роли.

Внести на диаграмму структурную декомпозицию работ.

Установить зависимости между работами.

Привязать работы ко времени.

Привязать работы к сотрудникам.

Убедиться в корректности установки нагрузки на исполнителей (чтобы суммарная загрузка сотрудников в единицу времени не превышала 100%).

Построить PERT-диаграмму.

Контрольные вопросы/задания:

Знать: современные методологии разработки программного обеспечения	1.Что такое структурная декомпозиция работы? 2.Каков рекомендуемый предел декомпозиции работ? 3.По какому принципу можно декомпонировать работы?
--	--

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-2. Расчет сетевого графика

Формы реализации: Проверка задания

Тип контрольного мероприятия: Контрольная работа

Вес контрольного мероприятия в БРС: 8

Процедура проведения контрольного мероприятия: Выдача заданий на построение сетевого графика работ и вычисление критического пути, выполняемых письменно. Сбор решений и проверка.

Краткое содержание задания:

Дана СДР в форме списка работ с оценками по времени и список зависимостей работ. Необходимо изобразить сетевой график и вычислить критический путь.

Контрольные вопросы/задания:

Знать: современные методологии разработки программного обеспечения	1.Что такое сетевой график? 2.Что такое критический путь?
--	--

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-3. Использование систем управления задачами и заявками

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 8

Процедура проведения контрольного мероприятия: Проводится в форме выполнения и защиты лабораторной работы.

Краткое содержание задания:

Цель работы.

Получить навыки работы в системе управления задачами и заявками. Изучить применение гибкого планирования при управлении проектами по разработке программного обеспечения.

Контрольные вопросы/задания:

Знать: современные методологии разработки программного обеспечения	1. Что такое гибкое планирование? 2. Чем гибкое планирование отличается от традиционного планирования?
--	---

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-4. Использование систем календарного планирования.

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 8

Процедура проведения контрольного мероприятия: Проводится в форме выполнения и защиты лабораторной работы.

Краткое содержание задания:

Научиться применять техники календарного планирования: строить структурную декомпозицию работ, сетевые графики и диаграммы Ганта, верстать календарные планы и распределять ресурсы.

Подготовка к работе

1. Ознакомиться с целью и назначением диаграмм Ганта;
2. Ознакомиться с принципом построения диаграмм Ганта;
3. Ознакомиться с целью и назначением PERT-диаграмм;

Порядок выполнения работы

1. Создать новый проект.
2. Добавить в проект сотрудников и их роли.
3. Внести на диаграмму структурную декомпозицию работ.
4. Установить зависимости между работами.
5. Привязать работы ко времени.
6. Привязать работы к сотрудникам.
7. Убедиться в корректности установки нагрузки на исполнителей (чтобы суммарная загрузка сотрудников в единицу времени не превышала 100%).

8. Построить PERT-диаграмму.

Контрольные вопросы/задания:

Знать: современные методологии разработки программного обеспечения	1.Зачем нужна диаграмма Ганта. 2.Зачем нужна PERT-диапаграмма.
--	---

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-5. Практическое применение систем контроля версий

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 8

Процедура проведения контрольного мероприятия: Проводится в форме выполнения и защиты лабораторной работы.

Краткое содержание задания:

Получить опыт практической работы с системой контроля версий.

Порядок выполнения работы

1. Создать локальный репозиторий.
2. Выложить в репозиторий тестовый проект Visual Studio.
3. Добавить в тестовый проект новый класс. Изменить существующий код. Выложить в репозиторий.
4. Осуществить откат к старой версии выложенного в репозиторий проекта.
5. Удалить локальную копию проекта и скачать последнюю версию из репозитория

Контрольные вопросы/задания:

Уметь: применять современные CASE-средства	1.Опишите назначение и аргументы команды git push 2.Опишите назначение и аргументы команды git commit 3.Опишите назначение и аргументы команды git add
--	--

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-6. Разработка шаблонов для генерации кода

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 8

Процедура проведения контрольного мероприятия: Проводится в форме выполнения и защиты лабораторной работы.

Краткое содержание задания:

Разработать шаблон для генерации новых проектов

Контрольные вопросы/задания:

Знать: современные архитектурные концепции, применяемые при разработке программных систем	1.Зачем нужны шаблоны кода?
Уметь: применять типовые архитектурные решения	1.Опишите возможности пакета text template transformation toolkit. 2.Из чего состоит шаблон text template transformation toolkit. 3.Может ли генерация кода по шаблону быть совмещена с процессом компиляции (встроена в него).

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-7. Автоматизация сборки программного проекта. Применение системы компонентной сборки.

Формы реализации: Компьютерное задание

Тип контрольного мероприятия:

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Проводится в форме выполнения и защиты лабораторной работы.

Краткое содержание задания:

Получить навыки развертывания системы автоматической сборки, разработки скриптов автоматизации сборки проекта, внедрения системы компонентной сборки.

Вариант 1:

Автоматизация сборки с применением Microsoft Build (MSBuild).

Вариант 2:

Автоматизация сборки с применением Apache Maven.

Контрольные вопросы/задания:

Уметь: применять типовые архитектурные решения	1.Опишите структуру скрипта MSBuild.
--	--------------------------------------

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-8. Расчет показателей качества кода

Формы реализации: Проверка задания

Тип контрольного мероприятия: Реферат

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Выдача заданий на построение системы показателей качества кода. Сбор решений и проверка.

Краткое содержание задания:

Выполнить обзор различных показателей качества кода, выполнить анализ их преимуществ и недостатков. Сформировать систему показателей качества кода и представить итог в форме реферата.

Контрольные вопросы/задания:

Уметь: создавать и внедрять стандарты кодирования и проектирования	1.Предложите методику оценки степени документированности библиотеки классов C#
--	--

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-9. Автоматизация документирования программного продукта

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Проводится в форме выполнения и защиты лабораторной работы.

Краткое содержание задания:

Выполнить документирование библиотеки классов.

С помощью генератора документации получить документацию к этой библиотеке.

Контрольные вопросы/задания:

Уметь: применять современные CASE-средства	1.Перечислите известные вам генераторы документации.
--	--

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-10. Практическое применение систем контроля качества кода на основе метрик на примере Microsoft Visual Studio.

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Проводится в форме выполнения и защиты лабораторной работы.

Краткое содержание задания:

Дан тестовый проект (библиотека классов на языке C#). Вычислить с помощью Visual Studio показатели качества кода. Предложить варианты улучшения кода для улучшения показателей качества.

Контрольные вопросы/задания:

Знать: технику снижения трудоемкости крупного проекта по разработке программного обеспечения	1.Какие метрики качества кода вы знаете?
Уметь: анализировать архитектуру программных систем, оценивать уместность применения той или иной разновидности решения и технологии доступа к данным	1.Приведите формулу по которой считается показатель качества кода в Visual Studio. 2.Как запустить подсчет метрик кода?

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-11. Применение средства контроля качества кода на базе правил

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Проводится в форме выполнения и защиты лабораторной работы.

Краткое содержание задания:

Дан тестовый проект (библиотека классов на языке C# или C++). Настроить автоматическую проверку качества кода по правилам во время компиляции. Сконфигурировать профиль проверки (какие предупреждения считать ошибками). Предложить варианты улучшения кода для улучшения показателей качества. Создать свое собственное правило проверки качества кода.

Контрольные вопросы/задания:

Знать: современную технику планирования работ в проекте по разработке программного обеспечения	1.Укажите отличие технологий статического и динамического анализа.
--	--

<p>Уметь: анализировать архитектуру программных систем, оценивать уместность применения той или иной разновидности решения и технологии доступа к данным</p>	<p>1.Опишите как настроить автоматическую проверку качества кода по правилам во время компиляции.</p>
--	---

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто, выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

8 семестр

Форма промежуточной аттестации: Экзамен

Пример билета

1. Проект. Характеристики проекта.
2. Автоматическая сборка.

Процедура проведения

Экзамен устный, по билетам. Студенту дается 30 минут на подготовку, во время ответа задаются дополнительные вопросы на темы, рассмотренные в рамках курса.

1. Перечень компетенций/индикаторов и контрольных вопросов проверки результатов освоения дисциплины

1. Компетенция/Индикатор: ИД-1ПК-1 Демонстрирует знания современных CASE-средств для выполнения этапов жизненного цикла ПО

Вопросы, задания

1.

Проект. Управление проектами

1. Цели и задачи курса.
2. Что такое проект?
3. Характеристики проекта.
4. Отличия проекта от процесса.
5. Окружение проекта.
6. Жизненный цикл проекта.
7. Задачи управляющего проектом.
8. Успешность проекта. Критерии успешности проекта.

2.

Управление IT-проектами

1. Особенности управления IT-проектами
2. Управление проектом и жизненный цикл
3. Типичные ошибки в управлении IT-проектом
4. Области знаний в управлении
5. Методы управления проектами

3.

Классический подход к управлению проектами

1. 5 этапов традиционного менеджмента
2. Фазы управления
3. Схема управления проектом
4. Жизненный цикл проекта
5. Сильные стороны классического проектного менеджмента
6. Слабые стороны классического проектного менеджмента

4.

[Жизненный цикл программного обеспечения](#)

1. Классификация технологических подходов ведения ЖЦ ПО
2. Классический набор технологических процессов
3. Каскадная модель. Преимущества и недостатки
4. Поэтапная модель “с возвратами”
5. Спиральная модель. Преимущества и недостатки
6. Итеративная модель
7. Преимущества итеративного подхода
8. Модель процессов MSF
9. Процессы жизненного цикла по стандарту ISO 12207

5.

[Планирование работ](#)

1. Суть задачи планирования
2. Сетевое планирование. Методы сетевого планирования
3. Календарное планирование. Схема календарного планирования
4. Детерминированные методы сетевого планирования: метод критического пути, диаграммы Ганта
5. Вероятностные методы
6. Иерархическая структура работ
7. Сетевой график
8. Ресурсное планирование

6.

[Microsoft Solutions Framework](#)

1. Что такое MSF
2. Модель процессов MSF. Как работает модель процессов MSF.
3. Этапы модели процессов MSF
4. Модель команд MSF
5. Требования к модели команды
6. Ключевые концепции используемые при формировании команды
7. Роли в модели команд MSF
8. Совмещение ролей
9. Дисциплины MSF
10. Управление рисками
11. Управление готовностью
12. Управление проектом
13. Управление областью действия проекта
14. Треугольник компромиссов
15. Microsoft Operations Framework
16. Базовые принципы MSF в управлении
17. Области управления проектами
18. Масштабируемость MSF
19. Масштабирование команд MSF

7.

[Гибкие методологии](#)

1. Что такое адаптивные методологии разработки программного обеспечения
2. Жизненный цикл agile
3. Основные концепции гибкой методологии
4. Манифест agile: ценности и принципы
5. Виды гибких методологий
6. Особенности гибких методологий

8.

[Гибкое планирование. Системы управления задачами и заявками](#)

1. Гибкое планирование.
2. Предпосылки появления систем управления задачами и заявками.
3. Системы отслеживания ошибок (Bug-tracking). Основные понятия. Обзор.
4. Система отслеживания ошибок Bugzilla.
5. Системы управления задачами и заявками. Основные понятия. Обзор.
6. Система управления задачами JIRA.
7. Система управления задачами TracStudio.

9.

[Экстремальное программирование](#)

1. Схема процесса разработки
2. Живое планирование
3. Частая смена версий
4. Рефакторинг
5. Разработка на основе тестирования
6. Коллективное владение кодом
7. Постоянная интеграция
8. Использование кода как средства коммуникации
9. Особенности взаимодействия с заказчиком
10. Достоинства методологии XP
11. Недостатки методологии XP

10.

[Методология Scrum](#)

1. Принципы Scrum
2. Жизненный цикл Scrum
3. Упрощенная модель итерации
4. Роли в scrum
5. Практики
6. Основные документы
7. Достоинства
8. Недостатки

11.

[Методология OpenUP](#)

1. Отличия OpenUP от RUP
2. Жизненный цикл OpenUP
3. Схема планирования
4. Модель команды

5. Преимущества и недостатки

12.

Функционально-ориентированная разработка

1. Обзор методологии.
2. Жизненный цикл.
3. Модель команды.
4. Преимущества FDD.
5. Недостатки FDD.

13.

Сравнение методологий RUP, MSF и гибких методологий

1. Определение критериев для сравнения методологий.
2. Сравнение методологий RUP, MSF и гибких методологий

Материалы для проверки остаточных знаний

1. Приведите характеристики проекта.
2. Что такое цель проекта?
3. Укажите чем проект отличается от процессной деятельности.
4. Что из перечисленного не является преимуществом проектной организационной структуры?
5. Укажите задачи, решаемые на фазе инициации проекта.
6. Что такое предметная область проекта.
7. Как используется метод критического пути?
8. Что такое структурная декомпозиция работ?
9. Что такое жизненный цикл проекта?
10. Какие виды жизненных циклов проекта вы знаете?

2. Компетенция/Индикатор: ИД-3ПК-1 Формирует архитектуру ПО

Вопросы, задания

1. Зачем нужна архитектура ПО. Виды архитектур ПО
- Многослойная архитектура (Layered Architecture).
 - Многоуровневая архитектура (Tiered Architecture).
 - Сервис-ориентированная архитектура (Service Oriented Architecture — SOA).
 - Микросервисная архитектура (Microservice Architecture).

Материалы для проверки остаточных знаний

1. Укажите отличия многослойной архитектуры от многоуровневой архитектуры.
2. Приведите отличия сервис-ориентированной архитектуры от микро-сервисной архитектуры.
3. Что такое микро-сервис

3. Компетенция/Индикатор: ИД-3ПК-2 Выбирает СУБД на основе их характеристики

Вопросы, задания

1. Системы контроля версий и конфигурационного управления.
 - 1 Модель работы с изменяющейся информацией. Контроль версий.
 - 2 Модели версионирования.
 - 3 Система управления версиями Subversion.

- 4 Система контроля версий GIT. Процесс Git Flow.
- 5 Система управления проектом Microfocus StarTeam.

2. Автоматизация сборки.

- 1 Задача автоматизации сборки. Требования к системам автоматизации сборки.
- 2 Типы систем автоматизации сборки.
- 3 Apache Ant.
- 4 MS Build.
- 5 Компонентная сборка.
- 6 Apache Maven.
- 7 JetBrains TeamCity.
- 8 Jenkins.
- 9 Component Builder.

3. Генерация кода по шаблону.

- 1 Стандартизация в разработке больших программных комплексов. Генерация кода. Методы генерации кода по шаблону.
- 2 Генерация кода на основе диаграмм.
- 3 Прямое визуальное проектирование.
- 4 Использование макропроцессоров для генерации кода.
- 5 Технология XSLT.
- 6 Создание шаблонов приложений в Embarcadero RAD Studio.
- 7 Шаблоны проектов и генерация каркасов типовых приложений в Eclipse.
- 8 Генерация кода с применением Java Emitter Templates.
- 9 Шаблоны проектов и генерация типовых приложений в Microsoft Visual Studio.
- 10 Разработка шаблонов с применением Text template transformation toolkit.

Материалы для проверки остаточных знаний

1. Чем отличается децентрализованная система контроля версий от централизованной?
2. Укажите какие задачи решает система автоматизации сборки.
3. Для чего используются средства генерации кода?
4. Какие модели версионирования вы знаете?

4. Компетенция/Индикатор: ИД-4ПК-2 Формирует и реализует базы данных в СУБД и организовать программный доступ к данным

Вопросы, задания

1.

[Выбор и внедрение методологии. Настройка процесса разработки](#)

1. Критерии, определяющие выбор методологии;
2. Выбор и учет альтернатив;
3. Адаптация методологии под нужды и реалии конкретной команды;
4. Настройка процесса разработки.

2. Контроль работоспособности кода.

- 1 Работоспособность кода.
- 2 Средства контроля работоспособности кода.
- 3 Модульное тестирование.
- 4 Разработка посредством тестирования.
- 5 Разработка модульных тестов в среде Visual Studio.

3. Контроль качества.

- 1 Качество кода. Оценка качества кода.
- 2 Метод метрик. Основные метрики кода.
- 3 Обнаружение и исправление ошибок в коде C/C++, C#, Java.

- 4 Контроль качества кода на базе правил.
- 5 Средства статического анализа кода. FxCop. Roslyn Analyzers.
- 6 Применение расширений Visual Assist X и JetBrains ReSharper.
- 4. Автоматизация тестирования.
 - 1 Средства автоматизации тестирования. Автоматизация тестирования приложений через графический интерфейс пользователя.
 - 2 Применение планировщиков для автоматизации тестирования. Специализированные программы-планировщики. Планировщик xStarter. Планировщик Macro Scheduler.
 - 3 Автоматизация тестирования приложений с помощью Microfocus Silk Test.
 - 4 Автоматизация тестирования программ с помощью IBM Rational Functional Tester.
 - 5. Автоматическое документирование и средства корпоративной памяти.
 - 1 Автоматическая генерация документации.
 - 2 Системы корпоративной памяти и накопления знаний.
 - 3 Системы накопления знаний, организованные по принципу «единого источника».

Материалы для проверки остаточных знаний

- 1. Перечислите основные метрики кода.
- 2. Приведите характеристики качественного кода.
- 3. Чем качественный код отличается от работоспособного кода

II. Описание шкалы оценивания

Оценка: 5

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Работа выполнена в рамках "продвинутого" уровня. Ответы даны верно, четко сформулированные особенности практических решений

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Работа выполнена в рамках "базового" уровня. Большинство ответов даны верно. В части материала есть незначительные недостатки

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Работа выполнена в рамках "порогового" уровня. Основная часть задания выполнена верно. на вопросы углубленного уровня

III. Правила выставления итоговой оценки по курсу

Оценка определяется в соответствии с Положением о балльно-рейтинговой системе для студентов НИУ «МЭИ» на основании семестровой и аттестационной составляющих.