

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»**

Утверждаю:
Зав. кафедрой компьютерных
интеллектуальных технологий
проектирования

 М.И. Чижов
«21» декабря 2021 г.

**УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ
ПО ДИСЦИПЛИНЕ**

«Управление жизненным циклом программного обеспечения »

Направление подготовки: 09.04.01 Информатика и вычислительная техника

Направленность (профиль): Искусственный интеллект

Квалификация выпускника магистр

Нормативный период обучения 2 года / 2 года и 5 м.

Форма обучения очная / заочная

Год начала подготовки 2022

Составитель:

Составитель: Чижов Михаил Иванович, д.т.н., профессор, заведующий кафедрой
Компьютерных интеллектуальных технологий проектирования Воронежского
государственного технического университета (ВГТУ)

Ершов Евгений Валентинович, д.т.н., профессор, директор института информационных
технологий, зав. кафедрой МПО ЭВМ ЧГУ

г. Воронеж - 2021

Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

1. Аппело, Ю. Agile-менеджмент. Лидерство и управление командами=Managtmnt 3.0. Leading Agile Developers, Developing Agile Leaders : [12+] / Ю. Аппело ; науч. ред. А. Обухова ; ред. А. Черникова ; пер. с англ. А. Олейник. – Москва : Альпина Паблишер, 2018. – 534 с. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=570321>. – Библиогр. в кн. – ISBN 978-5-9614-6361-3. – Текст : электронный.

2. Лауферман, О. В. Разработка программного продукта: профессиональные стандарты, жизненный цикл, командная работа : учебное пособие / О. В. Лауферман, Н. И. Лыгина. — Новосибирск : Новосибирский государственный технический университет, 2019. — 75 с. — ISBN 978-5-7782-3893-0. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/99215.html>. — Режим доступа: для авторизир. пользователей

3. Организация самостоятельной работы обучающихся: методические указания для студентов, осваивающих основные образовательные программы высшего образования — бакалавриата, специалитета, магистратуры: методические указания / сост. В.Н. Почечихина, И.Н. Крючкова, Е.И. Головина, В.Р. Демидов; ФГБОУ ВО «Воронежский государственный технический университет». – Воронеж, 2020. – 14 с.

Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

Ресурсы информационно-телекоммуникационной сети «Интернет»:

- Образовательный портал ВГТУ
- <http://www.edu.ru/>
- <https://www.atlassian.com/>
- <https://github.com/>
- <https://app.diagrams.net/>

Информационно-справочные системы:

- <http://window.edu.ru>
- <https://wiki.cchgeu.ru/>

Современные профессиональные базы данных:

- <https://proglib.io>

Учебно-методические указания и рекомендации

к изучению тем лекционных и практических занятий, самостоятельной работе студентов

очная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Лаб. зан.	СРС	Всего, час
1	Модели жизненного цикла программного обеспечения	Понятие жизненного цикла программного обеспечения. Стандартизация. Традиционные модели жизненного цикла программного продукта. Преимущества и недостатки. Области применения	6	4	72	82
2	Принципы разработки программного обеспечения	Гибкая разработка. Agile. Lean.	4	4	72	80
3	Основные методы разработки программного обеспечения	Scrum. Kanban. RUP (Rational Unified Process). DSDM (Dynamic Systems Development Model), RAD (Rapid Application Development), XP (Extreme Programming)	4	4	72	80
4	Автоматизация управления жизненным циклом программно	Системы SLM (Software Life Management), ALM (Application Lifecycle Management)	4	6	72	82
Итого			18	18	288	324

заочная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Лаб. зан.	СРС	Всего, час
1	Модели жизненного цикла программного обеспечения	Понятие жизненного цикла программного обеспечения. Стандартизация. Традиционные модели жизненного цикла программного продукта. Преимущества и недостатки. Области применения	2	2	84	88
2	Принципы разработки программного обеспечения	Гибкая разработка. Agile. Lean.	2	2	84	88
3	Основные методы разработки программного обеспечения	Scrum. Kanban. RUP (Rational Unified Process). DSDM (Dynamic Systems Development Model), RAD (Rapid Application Development), XP (Extreme Programming)	-	2	84	86
4	Автоматизация управления жизненным циклом программно	Системы SLM (Software Life Management), ALM (Application Lifecycle Management)	-	2	87	89
Итого			4	8	339	351

Темы лабораторных работ

Лабораторная работа № 1. Моделирование предметной области. Первоначальное формирование требований. Анализ категорий пользователей. Выбор целевой платформы. Выполнить моделирование предметной области, разработать модели для решения задач диссертационного исследования, реализовать алгоритмические решения полученных моделей.

Контрольные вопросы:

1. Основные подходы, применяемые при моделировании предметной области.
2. Информационные и концептуальные модели предметной области.
3. Оценка качества концептуального моделирования предметной области.

4. Ваш проект является независимой разработкой или расширением существующей системой?
5. На каких платформах будет работать решений?
6. Какие категории пользователей будут в системе?
7. Какие планируются отличия в интерфейсе для разных пользователей?
8. С какими системами будет интегрироваться ваше решение?

Лабораторная работа № 2. Предметно-ориентированное проектирование. Определение модели жизненного цикла. Первоначальное планирование. Распределение ролей. Выполнить предметно-ориентированное проектирование, разработать многоуровневую архитектуру системы, реализующей методы обработки информации для решения задач диссертационного исследования.

Контрольные вопросы:

1. Классификация методов проектирования.
2. Классификация средств проектирования.
3. Классификация процессов жизненного цикла.
4. Основные процессы жизненного цикла.
5. Сервис-ориентированная архитектура.
6. Что такое жизненный цикл?
7. Почему выбрана такая модель жизненного цикла?
8. Как будут распределяться задачи в рамках вашего проекта?

Лабораторная работа № 3. Применение формальных систем и аналитических шаблонов. Подготовка среды проекта. Создание и настройка репозитория github.com. Создание и настройка проекта github.com. Описать возможность применения аналитических шаблонов, моделей и формальных систем для решения задач диссертационного исследования.

Контрольные вопросы:

1. Аналитические шаблоны: возможность применения и имеющиеся ограничения.
2. Особенности формальных систем.
3. Оценка эффективности применения аналитических шаблонов, моделей и формальных систем.
4. Какие типы репозиториев поддерживает github.com?
5. Как выполняется управление версиями?
6. Почему была выбрана такая структура репозитория?
7. Какая методология управления проектами используется github.com?

Лабораторная работа № 4. Проектирование предметно-ориентированных программных систем. Реализация программного средства. Проведение релиза. Управление версиями

проекта. Тестирование решения. Выполнить проектирование предметно-ориентированной программной системы, предложить структурно-функциональную организацию, разработать архитектуру системы для решения задач диссертационного исследования.

Контрольные вопросы:

1. Состав и содержание работ на предпроектной стадии.
2. Состав и содержание работ на стадии технического и рабочего проектирования.
3. Состав и содержание работ на стадиях внедрения, эксплуатации и сопровождения.
4. Методы проведения испытаний.
5. По каким критериям были выбраны средства реализации?
6. Как организовано взаимодействие среды разработки и системы управления версиями?
7. Какой этап разработки оказался самым сложным?

Средства контроля качества обучения

Вопросы к экзамену

1. Жизненный цикл программного обеспечения.
2. Стандартизация жизненного цикла ПО.
3. Традиционные модели жизненного цикла программного продукта.
4. Преимущества и недостатки традиционных моделей ЖЦ ПО.
5. Области применения моделей ЖЦ ПО.
6. Гибкая разработка ПО.
7. Методология Agile.
8. Методология Lean.
9. Методология Scrum.
10. Методология Kanban.
11. Методология RUP (Rational Unified Process).
12. Методология DSDM (Dynamic Systems Development Model).
13. Методология RAD (Rapid Application Development)
14. Методология XP (Extreme Programming)
15. SLM и ALM системы. Основные возможности.
16. SLM и ALM системы. Типовая архитектура.
17. Средства коллективной разработки программного обеспечения.
18. Системы управления версиями ПО.