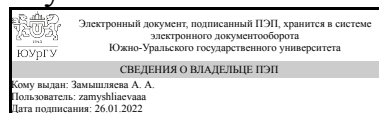


ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ:
Директор института
Институт естественных и точных
наук



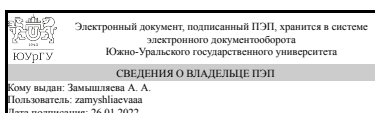
А. А. Замышляева

РАБОЧАЯ ПРОГРАММА

дисциплины 1.Ф.02 Анализ требований и проектирование ПО
для направления 01.03.02 Прикладная математика и информатика
уровень Бакалавриат
форма обучения очная
кафедра-разработчик Прикладная математика и программирование

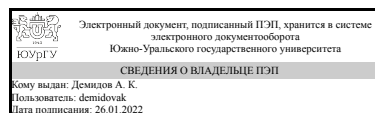
Рабочая программа составлена в соответствии с ФГОС ВО по направлению подготовки 01.03.02 Прикладная математика и информатика, утверждённым приказом Минобрнауки от 10.01.2018 № 9

Зав.кафедрой разработчика,
д.физ.-мат.н., проф.



А. А. Замышляева

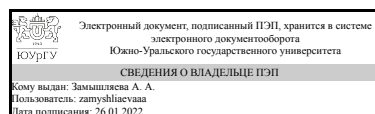
Разработчик программы,
доцент



А. К. Демидов

СОГЛАСОВАНО

Руководитель направления
д.физ.-мат.н., проф.



А. А. Замышляева

1. Цели и задачи дисциплины

Преподаваемая дисциплина посвящена изучению двух важных этапов жизненного цикла программного обеспечения - анализу предметной области и проектированию архитектуры программных систем. Преподавание и изучение дисциплины следует рассматривать как полезную составляющую профессиональной подготовки. Целью преподавания дисциплины является обучение студентов методам выявления, анализа и разработки требований и методам проектирования сложных программных систем. Задачи дисциплины заключаются в том, чтобы студенты овладели навыками анализа предметной области, создания и описания объектно-ориентированных моделей предметной области, выполнения системного анализа и разработки на его основе архитектуры, алгоритмических и программных решений системного и прикладного программного обеспечения; навыками создания спецификаций, как для всей системы в целом, так для отдельных подсистем и модулей.

Краткое содержание дисциплины

Основы управления требованиями к программному обеспечению. Методы выявления требований и заинтересованных лиц. Спецификация требований. Требования к типичным системам. Согласование требований и управление рисками. Проблемы разработки сложных программ. Структурный подход в проектировании ПО и классификация структурных методологий. Архитектура ПО, влияние архитектуры на свойства ПО. Унифицированный язык моделирования UML. Применения шаблонов проектирования к прикладным задачам. Исследование существующих архитектур промежуточного программного обеспечения. Компонентное проектирование. Классификация CASE-систем и их сравнительная характеристика.

2. Компетенции обучающегося, формируемые в результате освоения дисциплины

| Планируемые результаты освоения ОП ВО (компетенции) | Планируемые результаты обучения по дисциплине |
|--|---|
| ПК-1 Способен активно участвовать в разработке системного и прикладного программного обеспечения | Знает: методы выявления, анализа и разработки требований при проектировании сложных программных систем Умеет: создавать и описывать объектно-ориентированные модели предметной области Имеет практический опыт: создания спецификаций как для всей системы в целом, так для отдельных подсистем и модулей |

3. Место дисциплины в структуре ОП ВО

| Перечень предшествующих дисциплин, видов работ учебного плана | Перечень последующих дисциплин, видов работ |
|---|---|
| 1.Ф.10 Web-программирование | Не предусмотрены |

Требования к «входным» знаниям, умениям, навыкам студента, необходимым при освоении данной дисциплины и приобретенным в результате освоения предшествующих дисциплин:

| Дисциплина | Требования |
|-----------------------------|---|
| 1.Ф.10 Web-программирование | Знает: основные этапы, методологию и технологию построения Web-систем Умеет: разрабатывать программное обеспечение, основанное на Web-интерфейсе Имеет практический опыт: разработки Web-приложений |

4. Объём и виды учебной работы

Общая трудоемкость дисциплины составляет 2 з.е., 72 ч., 52,25 ч. контактной работы

| Вид учебной работы | Всего часов | Распределение по семестрам в часах | |
|--|-------------|------------------------------------|--|
| | | Номер семестра | |
| | | 8 | |
| Общая трудоёмкость дисциплины | 72 | 72 | |
| <i>Аудиторные занятия:</i> | 48 | 48 | |
| Лекции (Л) | 24 | 24 | |
| Практические занятия, семинары и (или) другие виды аудиторных занятий (ПЗ) | 24 | 24 | |
| Лабораторные работы (ЛР) | 0 | 0 | |
| <i>Самостоятельная работа (СРС)</i> | 19,75 | 19,75 | |
| с применением дистанционных образовательных технологий | 0 | | |
| Изучение теоретического материала к практическим занятиям | 11,75 | 11.75 | |
| Подготовка к дифференцированному зачету | 8 | 8 | |
| Консультации и промежуточная аттестация | 4,25 | 4,25 | |
| Вид контроля (зачет, диф.зачет, экзамен) | - | диф.зачет | |

5. Содержание дисциплины

| № раздела | Наименование разделов дисциплины | Объем аудиторных занятий по видам в часах | | | |
|-----------|--|---|----|----|----|
| | | Всего | Л | ПЗ | ЛР |
| 1 | Разработка и анализ требований | 24 | 12 | 12 | 0 |
| 2 | Проектирование архитектуры и разработка спецификаций модулей | 24 | 12 | 12 | 0 |

5.1. Лекции

| № лекции | № раздела | Наименование или краткое содержание лекционного занятия | Кол-во часов |
|----------|-----------|---|--------------|
|----------|-----------|---|--------------|

| | | | |
|----|---|---|---|
| 1 | 1 | Основы управления требованиями к программному обеспечению. Процессы инженерии требований: выявление требований, спецификация, анализ и управление. | 2 |
| 2 | 1 | Типы требований: функциональные, нефункциональные, атрибуты качества. Потребности и функции продукта. | 2 |
| 3 | 1 | Выявление требований: определение потребностей, целей и требований. Заказчики и другие заинтересованные лица. Интервью, наблюдения, совещания, мозговой штурм, раскадровки, прототипирование. | 2 |
| 4 | 1 | Спецификация требований: текстовые и графические нотации и языки (UML, нотации пользовательских требований). Методы написания высококачественных требований. Стандарты документирования. | 2 |
| 5 | 1 | Анализ требований: инспекция, аттестация, завершенность, обнаружение конфликтов и несоответствий. Анализ взаимодействия элементов функциональности и разрешение противоречий. | 2 |
| 6 | 1 | Требования к типичным системам: встроенным системам, потребительским системам, web-системам, бизнес-системам, научным системам и другим инженерным системам. Согласование требований и управление рисками. Интеграция анализа требований и процессов разработки программного обеспечения (включая Agile-процессы). | 2 |
| 7 | 2 | Проблемы разработки сложных программ. Структурный подход в проектировании ПО и классификация структурных методологий. Диаграммы «сущность-связь» (ERD), диаграммы потоков данных (DFD), SADT-модели (стандарт IDEF0). | 2 |
| 8 | 2 | Архитектура ПО, влияние архитектуры на свойства ПО. Особенности разработки сложных программных систем: иерархичность, групповая разработка, сборочное проектирование. Преимущества и недостатки объектно-ориентированного подхода. | 2 |
| 9 | 2 | Основные понятия унифицированного языка моделирования (UML). Диаграммы прецедентов, диаграммы классов, диаграммы взаимодействий, диаграммы последовательности действий, диаграммы состояний, компонентные диаграммы. | 2 |
| 10 | 2 | Углубленное изучение проектирования программного обеспечения, шаблонов проектирования, сред разработки и архитектур. Применения шаблонов проектирования к прикладным задачам. | 2 |
| 11 | 2 | Проектирование распределенных систем с использованием промежуточного программного обеспечения (middleware). Изучение архитектур COM, CORBA и .NET. Исследование существующих архитектур промежуточного программного обеспечения. Компонентное проектирование. | 2 |
| 12 | 2 | Классификация CASE-систем и их сравнительная характеристика. Тенденции развития объектно-ориентированных инструментальных средств. Поддержка графических моделей. Репозитарий и контроль ошибок. Теория измерения и использование метрик в проектировании. Проектирование с учетом таких качеств, как производительность, безопасность, защищенность, возможность повторного использования, надежность и т.д. Измерение внутренних параметров и сложности программного обеспечения. | 2 |

5.2. Практические занятия, семинары

| № занятия | № раздела | Наименование или краткое содержание практического занятия, семинара | Кол-во часов |
|-----------|-----------|---|--------------|
| 1 | 1 | Анализ множества существующих программных систем: измерение качества и восстановление требований (reverse engineering) по программе. Использование инструментов управления требованиями. | 2 |

| | | | |
|-------|---|---|---|
| 2 | 1 | Анализ проблемы. Выявление причин проблемы. | 2 |
| 3 | 1 | Моделирование, прототипирование и анализ требований с помощью средств UML | 2 |
| 4 | 1 | Создание документа-концепции для учебного проекта | 2 |
| 5,6 | 1 | Создание спецификации для учебного проекта | 4 |
| 7 | 2 | Разработка архитектуры учебного проекта | 2 |
| 8 | 2 | Разработка диаграммы классов для учебного проекта | 2 |
| 9 | 2 | Создание спецификации для нескольких модулей учебного проекта | 2 |
| 10 | 2 | Разработка диаграмм взаимодействий, диаграмм последовательности действий для учебного проекта | 2 |
| 11,12 | 2 | Проектирование распределенных систем с использованием промежуточного программного обеспечения. Использование CASE-средств | 4 |

5.3. Лабораторные работы

Не предусмотрены

5.4. Самостоятельная работа студента

| Выполнение СРС | | | |
|---|--|---------|--------------|
| Подвид СРС | Список литературы (с указанием разделов, глав, страниц) / ссылка на ресурс | Семестр | Кол-во часов |
| Изучение теоретического материала к практическим занятиям | ЭУМД, осн. лит. 1. с. 6-250, ЭУМД, доп. лит. 2, гл. 4, 5, 8, 17, 18, 20, ЭУМД, осн. лит. 3, гл. 3, 4, 6-10 | 8 | 11,75 |
| Подготовка к дифференцированному зачету | ЭУМД, осн. лит. 1. с. 6-250, ЭУМД, осн. лит. 3, гл. 3, 4, 6-10 | 8 | 8 |

6. Текущий контроль успеваемости, промежуточная аттестация

Контроль качества освоения образовательной программы осуществляется в соответствии с Положением о балльно-рейтинговой системе оценивания результатов учебной деятельности обучающихся.

6.1. Контрольные мероприятия (КМ)

| № КМ | Се-местр | Вид контроля | Название контрольного мероприятия | Вес | Макс. балл | Порядок начисления баллов | Учитывается в ПА |
|------|----------|------------------|-----------------------------------|-----|------------|--|--------------------------|
| 1 | 8 | Текущий контроль | Задание 1 | 1 | 10 | Выполнена регистрация участников группы в СУТ — 1 балл Зафиксированы не менее 5 существующих требований к ПО (reverse engineering) — 5 баллов (по 1 баллу за каждое требование) Предложено | дифференцированный зачет |

| | | | | | | | |
|---|---|------------------|-----------|---|----|---|--------------------------|
| | | | | | | изменение требования, улучшение функциональности ПО - 2 балла, иначе 0 баллов Выполнена классификация требований — 2 балла, иначе 0 баллов | |
| 2 | 8 | Текущий контроль | Задание 2 | 1 | 10 | Поставлена реальная проблема(ы) - 1 балл Указано не менее 5 причин для проблем - 5 баллов (по 1 баллу за причину) Есть причины причин - 1 балл Есть причины, которые можно решить программным путем - 1 балл Есть список из не менее 2 конкурирующих продуктов, которые будут использованы для сравнения (ссылки) - 2 балла (по 1 баллу за продукт) | дифференцированный зачет |
| 3 | 8 | Текущий контроль | Задание 3 | 1 | 10 | Нарисована диаграмма вариантов использования для всех прямых пользователей - 3 балла (оценка снижается на 1 балл за каждого отсутствующего пользователя) Варианты использования соответствуют потребностям и целям пользователей - 2 балла (оценка снижается на 1 балл за каждую ошибку) Есть описание одного из вариантов использования - 3 балла, иначе 0 баллов В описании нет противоречий - 2 балла (оценка снижается на 1 балл за каждую ошибку) | дифференцированный зачет |

| | | | | | | | |
|---|---|------------------|-----------|---|----|--|--------------------------|
| 4 | 8 | Текущий контроль | Задание 4 | 1 | 10 | <p>Структура документа-концепции соответствует шаблону - 2 балла, иначе 0 баллов</p> <p>Достаточное описание пользователя и его потребностей (пп 2.2,2.3,2.4) - 3 балла (оценка снижается на 1 балл за каждую ошибку)</p> <p>Рассмотрено не менее 2 альтернатив - 2 балла, 1 альтернатива - 1 балл, иначе 0 баллов</p> <p>В п.3 указано достаточное обоснование важности проекта - 1 балл</p> <p>Указано не менее 5 функций - 2 балла, от 3 до 4 функций - 1 балл, менее 3 функций - 0 баллов</p> | дифференцированный зачет |
| 5 | 8 | Текущий контроль | Задание 5 | 1 | 10 | <p>Структура спецификации соответствует шаблону - 2 балла, иначе 0 баллов</p> <p>Выполняется трассируемость функций из документа концепции - 1 балл</p> <p>Имеется подробная спецификация не менее 5 функциональных требований - 2 балла, 3-4 требования - 1 балл, менее 3 - 0 баллов</p> <p>Имеется подробная спецификация не менее 3 нефункциональных требований - 2 балла, 2 требования - 1 балл, , менее 2 - 0 баллов</p> <p>Указана спецификация для интерфейсов пользователя (не менее 3 диалогов) - 2 балла, 2 диалога - 1 балл, , менее 2 - 0 баллов</p> <p>Указана спецификация</p> | дифференцированный зачет |

| | | | | | | | |
|---|---|------------------|-----------|---|----|--|--------------------------|
| | | | | | | для хотя бы 1 программного интерфейса (API) - 1 балл | |
| 6 | 8 | Текущий контроль | Задание 6 | 1 | 10 | <p>Выбран язык программирования - 1 балл</p> <p>Выбран фреймворк, библиотеки для реализации интерфейса пользователя - 2 балла, устаревший фреймворк - 1 балл, не выбран - 0 баллов</p> <p>Выбрана СУБД - 1 балл</p> <p>Сделано обоснование выбора - 1 балл</p> <p>Нарисована диаграмма компонентов (подсистем) - 1 балл</p> <p>Обозначения на диаграмме соответствуют стандарту UML - 2 балла, оценка снижается на 1 балл за каждую ошибку</p> <p>Подсистемы и компоненты обеспечивают выполнение требований (функций) ПО - 2 балла, , оценка снижается на 1 балл за каждую ошибку</p> | дифференцированный зачет |
| 7 | 8 | Текущий контроль | Задание 7 | 1 | 10 | <p>Разработана диаграмма классов - 3 балла, иначе 0 баллов</p> <p>Правильно использованы обозначения UML - 3 балла, оценка снижается на 1 балл за каждую ошибку</p> <p>Каждому компоненту (задание 5) соответствует хотя бы один класс - 2 балла, отсутствует 1 компонент - 1 балл, отсутствует более 1 компонента - 0 баллов</p> <p>Направление связей в иерархии классах</p> | дифференцированный зачет |

| | | | | | | | |
|----|---|------------------|------------|---|----|---|--------------------------|
| | | | | | | обеспечивают устойчивость компонент - 2 балла, иначе 0 баллов | |
| 8 | 8 | Текущий контроль | Задание 8 | 1 | 10 | Нарисована диаграмма классов (не менее 2 классов) - 3 балла, иначе 0 баллов Использованы корректные графические обозначения UML для классов и связей - 2 балла, оценка снижается на 1 балл за каждую ошибку В диаграмме классов поля имеют правильные описания - 1 балл В диаграмме классов методы имеют правильные описания - 1 балл Указаны спецификации не менее 3 методов - 3 балла (по 1 баллу за каждую спецификацию, но не более 3 баллов) | дифференцированный зачет |
| 9 | 8 | Текущий контроль | Задание 9 | 1 | 10 | Нарисована диаграмма - 3 балла, иначе 0 баллов Использованы правильные обозначения UML - 4 балла, оценка снижается на 1 балл за каждую ошибку Соответствует выбранному процессу - 3 балла, оценка снижается на 1 балл за каждую ошибку | дифференцированный зачет |
| 10 | 8 | Текущий контроль | Задание 10 | 1 | 10 | Разработан API для задачи (не менее 4 услуг) — 4 балла (1 балл за каждую услугу, но не более 4 баллов) API описан с помощью WDSL/OpenAPI – 4 балла, оценка снижается на 1 балл за | дифференцированный зачет |

| | | | | | | | |
|----|---|--------------------------|--------------------------|---|----|--|--------------------------|
| | | | | | | каждую ошибку в описании Выполнено тестирование API – 2 балла, иначе 0 баллов | |
| 11 | 8 | Промежуточная аттестация | дифференцированный зачет | - | 10 | <p>Это контрольное мероприятие проводится в форме собеседования. Задаются два вопроса по пройденным темам. В первую очередь предлагаются вопросы по темам, которые были оценены на "неудовлетворительно" по текущему контролю. Каждый ответ оценивается от 0 до 5 баллов в зависимости от полноты ответа, знания терминов.</p> <p>Шкала оценивания</p> <p>Полный, правильный ответ - 5 баллов</p> <p>Одна неточность, неправильный термин - 4 балла</p> <p>Частичный ответ - 3 балла</p> <p>В ответе есть некоторые правильные определения - 2 балла</p> <p>Нет ответа - 0 баллов</p> <p>Оценка ставится как сумма баллов за оба ответа.</p> | дифференцированный зачет |
| 12 | 8 | Бонус | Бонус-рейтинг | - | 15 | <p>Активность на занятиях, посещаемость 100% посещение (допускаются пропуски по уважительной причине) - 3 балла</p> <p>85-99% посещение - 2 балла</p> <p>Работа у доски 1 балл за решение задачи у доски, но не более 5 баллов за семестр</p> | дифференцированный зачет |

6.2. Процедура проведения, критерии оценивания

| Вид промежуточной аттестации | Процедура проведения | Критерии оценивания |
|------------------------------|---|---|
| дифференцированный зачет | Возможно определение рейтинга студента по дисциплине по результатам текущего контроля в соответствии с п.2.6. Если студент не набрал необходимый рейтинг по текущему контролю, то проводится устное собеседование. Студент должен ответить на 2 вопроса из вопросов к дифференцированному зачету. В первую очередь предлагаются вопросы по темам, которые были оценены на "неудовлетворительно" по текущему контролю. | В соответствии с пп. 2.5, 2.6 Положения |

6.3. Оценочные материалы

| Компетенции | Результаты обучения | № КМ | | | | | | | | | | | | |
|-------------|--|------|---|---|---|---|---|---|---|---|----|----|----|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
| ПК-1 | Знает: методы выявления, анализа и разработки требований при проектировании сложных программных систем | + | + | + | + | + | + | | | | | | + | + |
| ПК-1 | Умеет: создавать и описывать объектно-ориентированные модели предметной области | | | | + | + | + | + | + | | | | + | + |
| ПК-1 | Имеет практический опыт: создания спецификаций как для всей системы в целом, так для отдельных подсистем и модулей | | | + | + | + | | | | + | + | + | + | |

Фонды оценочных средств по каждому контрольному мероприятию находятся в приложениях.

7. Учебно-методическое и информационное обеспечение дисциплины

Печатная учебно-методическая документация

а) *основная литература:*

Не предусмотрена

б) *дополнительная литература:*

Не предусмотрена

в) *отечественные и зарубежные журналы по дисциплине, имеющиеся в библиотеке:*

1. Программирование : науч. журн. /Рос. акад. наук, Отд-ние информатики, вычисл. техники и автоматизации, Моск. гос. ун-т

г) *методические указания для студентов по освоению дисциплины:*

1. Методические указания для СРС по выполнению лабораторных работ (на сервере кафедры Н:\Учебные материалы)

из них: учебно-методическое обеспечение самостоятельной работы студента:

1. Методические указания для СРС по выполнению лабораторных работ (на сервере кафедры Н:\Учебные материалы)

Электронная учебно-методическая документация

| № | Вид литературы | Наименование ресурса в электронной форме | Библиографическое описание |
|---|----------------|--|--|
| 1 | Основная | Электронно- | Новиков, Ф.А. Учебно-методическое пособие по |

| | | | |
|---|---------------------------|---|---|
| | литература | библиотечная система издательства Лань | дисциплине «Анализ и проектирование на UML». [Электронный ресурс] — Электрон. дан. — СПб. : НИУ ИТМО, 2007. — 286 с. http://e.lanbook.com/book/43540 |
| 2 | Дополнительная литература | Электронно-библиотечная система издательства Лань | Буч, Г. Язык UML. Руководство пользователя. [Электронный ресурс] / Г. Буч, Д. Рамбо, И. Якобсон. — Электрон. дан. — М. : ДМК Пресс, 2008. — 496 с. http://e.lanbook.com/book/1246 |
| 3 | Основная литература | Электронно-библиотечная система издательства Лань | Косяков, А. Системная инженерия. Принципы и практика. [Электронный ресурс] / А. Косяков, У. Свит. — Электрон. дан. — М. : ДМК Пресс, 2014. — 624 с. http://e.lanbook.com/book/66484 |

Перечень используемого программного обеспечения:

1. -WhiteStarUML (инструмент работы с диаграммами UML)(бессрочно)
2. Microsoft-Office(бессрочно)
3. Microsoft-Visio(бессрочно)

Перечень используемых профессиональных баз данных и информационных справочных систем:

Нет

8. Материально-техническое обеспечение дисциплины

| Вид занятий | № ауд. | Основное оборудование, стенды, макеты, компьютерная техника, предустановленное программное обеспечение, используемое для различных видов занятий |
|----------------------|----------|--|
| Лабораторные занятия | 332 (3б) | Компьютеры, редактор для UML диаграмм (UMLet, WhiteStar UML или MS Visio), MS Office |