

МИНОБРНАУКИ РОССИИ

Ярославский государственный университет им. П.Г. Демидова

Кафедра компьютерных сетей

УТВЕРЖДАЮ

Декан факультета ИВТ

 Д.Ю. Чалый

«24» мая 2022 г.

Рабочая программа дисциплины

«Основы промышленной backend и TDD разработки на java-технологиях»

Направление подготовки

09.04.03 прикладная информатика

Профиль

«Искусственный интеллект в корпоративных информационных системах»

Квалификация выпускника

Магистр

Форма обучения

очная

Программа рассмотрена на заседании кафедры от «22» марта 2022 г., протокол № 7

Программа одобрена НМК факультета ИВТ протокол № 6 от «18» апреля 2022 г. года

Ярославль
2022

1. Цели освоения дисциплины

Дисциплина «Основы промышленной backend и TDD разработки на Java-технологиях» относится к дисциплинам вариативной части.

Для освоения данной дисциплиной студенты должны владеть базовыми знаниями по дискретной математике, математической логике, знать основы проектирования информационных систем, иметь базовую подготовку по теории формальных моделей информационных систем и понимать основные принципы функционирования облачных сред.

2. Место дисциплины в структуре ОП магистратуры

Дисциплина «Основы промышленной backend и TDD разработки на Java-технологиях» относится к вариативной части (дисциплина по выбору) ОП магистратуры.

Главной особенностью данного курса является ориентация не на стандарты и модели процессов разработки, а на реальные и эффективные практики взаимодействия конкретных участников процесса разработки (разработчиков, менеджеров, тестировщиков), применяемые в современной ИТ-индустрии.

Содержание курса тесно связано фактически со всеми дисциплинами, которые изучались студентами. Освоению данной программы предшествуют учебные курсы по программированию и современным информационным технологиям.

Дисциплина «Основы промышленной backend и TDD разработки на Java-технологиях» обеспечивает закрепление и углубление теоретических знаний и практических навыков по основным дисциплинам ИТ-цикла. Дисциплина позволит уже на уровне университета подготовить специалистов, способных не только решать конкретные задачи разработки программного обеспечения, но и самостоятельно и гармонично вписываться в бизнес-процессы компании, максимально полно реализуя свои способности как в интересах компании, так и в интересах собственного профессионального развития.

3. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОП магистратуры

Процесс изучения дисциплины направлен на формирование следующих элементов компетенций в соответствии с ФГОС ВО, ОП ВО и приобретения следующих знаний, умений, навыков и (или) опыта деятельности:

Код компетенции	Формулировка компетенции	Перечень планируемых результатов обучения
Профессиональные компетенции		
ПК-5. Способен руководить проектами по созданию систем искусственного интеллекта с применением новых методов и алгоритмов машинного обучения со стороны заказчика	ПК-5.1. Руководит разработкой архитектуры комплексных систем искусственного интеллекта со стороны заказчика	Знать: - концептуальные и теоретические модели решаемых научных проблем и задач проектной и производственно-технологической деятельности. Уметь: - управлять проектами, планировать научно-исследовательскую деятельность, анализировать риски, управлять командой

		проекта; Владеть: - практическими навыками планирования работ по разработке проекта, подходами к управлению рисками
--	--	---

4. Объем, структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 1 зач. ед., 36 акад. час.

№ п/п	Темы (разделы) дисциплины, их содержание	Семестр	Виды учебных занятий, включая самостоятельную работу студентов, и их трудоемкость (в академических часах)						Формы текущего контроля успеваемости Форма промежуточной аттестации (по семестрам)
			Контактная работа						
			лекции	практические	лабораторные	консультации	аттестационные испытания	самостоятельная работа	
1.	Извлечение требований к ПО. Понятие требования. Роль требований в процессе промышленной разработки. Источники и пути выявления требований. Свойства качественных требований. Анализ и тестирование требований.	3		2				3	
2.	Управление требованиями к ПО. Характеристика процесса управления требованиями и его роль в промышленной разработке. Жизненный цикл требований. Основные задачи управления требованиями. Инструментальные средства управления требованиями.	3		2				3	
3.	Проектирование пользовательских интерфейсов. Общая	3		2		1		3	

	характеристика задачи проектирования пользовательских интерфейсов и её место в процессе разработки ПО. Понятие user experience (UX). Этапы разработки пользовательского интерфейса. Методы и инструментальные средства проектирования интерфейсов. Методы и инструментальные средства тестирования интерфейсов.								
4.	Гибкая методология разработки ПО. Основные положения гибкой методологии. Отличия гибкой методологии разработки от традиционной. Возможности и область применимости гибкой методологии. Отдельные методики в рамках гибкой методологии: XP, Scrum, Kanban, Lean.	3		2				3	
5.	Планирование выпусков ПО в рамках гибкой методологии разработки. Долгосрочное и краткосрочное планирование выпусков. Методики оценки трудозатрат на выполнение конкретных задач: покер планирования, определение размера футболки, оценка относительных масс, оценка на стене.	3		4				3	
6.	Использование трекера задач в промышленной разработке. Задачи, решаемые трекером задач в рамках процесса разработки. Жизненный цикл задачи в трекере. Трекер задач как средство регламентации деятельности разработчика.	3		4				2,8	
Всего за 3 семестр				18				17,8	Зачет
Всего				18				17,8	

Содержание разделов дисциплины:

Извлечение требований к ПО. Понятие требования. Роль требований в процессе промышленной разработки. Источники и пути выявления требований. Свойства качественных требований. Анализ и тестирование требований.

<p>Управление требованиями к ПО. Характеристика процесса управления требованиями и его роль в промышленной разработке. Жизненный цикл требований. Основные задачи управления требованиями. Инструментальные средства управления требованиями.</p>
<p>Проектирование пользовательских интерфейсов. Общая характеристика задачи проектирования пользовательских интерфейсов и её место в процессе разработки ПО. Понятие user experience (UX). Этапы разработки пользовательского интерфейса. Методы и инструментальные средства проектирования интерфейсов. Методы и инструментальные средства тестирования интерфейсов.</p>
<p>Гибкая методология разработки ПО. Основные положения гибкой методологии. Отличия гибкой методологии разработки от традиционной. Возможности и область применимости гибкой методологии. Отдельные методики в рамках гибкой методологии: XP, Scrum, Kanban, Lean.</p>
<p>Планирование выпусков ПО в рамках гибкой методологии разработки. Долгосрочное и краткосрочное планирование выпусков. Методики оценки трудозатрат на выполнение конкретных задач: покер планирования, определение размера футболки, оценка относительных масс, оценка на стене.</p>
<p>Использование трекера задач в промышленной разработке.</p> <p>Задачи, решаемые трекером задач в рамках процесса разработки. Жизненный цикл задачи в трекере. Трекер задач как средство регламентации деятельности разработчика.</p>

5. Образовательные технологии, используемые при осуществлении образовательного процесса по дисциплине

В процессе обучения используются следующие образовательные технологии:

Лекция-беседа или «диалог с аудиторией», является наиболее распространенной и сравнительно простой формой активного вовлечения студентов в учебный процесс. Эта лекция предполагает непосредственный контакт преподавателя с аудиторией. Преимущество лекции-беседы состоит в том, что она позволяет привлекать внимание студентов к наиболее важным вопросам темы, определять содержание и темп изложения учебного материала с учетом особенностей студентов.

Мастер-класс – это особая форма учебного занятия, когда преподаватель-мастер передает свой опыт путем прямого и комментированного показа последовательности действий, методов, приемов и форм педагогической деятельности. Целью проведения мастер-класса является профессиональное, интеллектуальное и эстетическое воспитание студентов, и прежде всего, развитие в ходе мастер-класса способности студента самостоятельно и нестандартно мыслить.

Практическое занятие – занятие, посвященное освоению конкретных умений и навыков и закреплению полученных на лекции знаний по предложенному алгоритму.

6. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения и информационных справочных систем (при необходимости)

В процессе осуществления образовательного процесса используются: для разработки документов, презентаций, для работы с электронными таблицами

OfficeStd 2013 RUS OLP NL Acdmc 021-10232

LibreOffice (свободное)

издательская система LaTeX;

– для поиска учебной литературы библиотеки ЯрГУ – Автоматизированная библиотечная информационная система "БУКИ-NEXT" (АБИС "Буки-Next").

7. Перечень основной и дополнительной учебной литературы, ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

а) основная:

1. Парамонов И. В. Инженерия программных систем и комплексов на основе гибкой методологии разработки: учеб.-метод. пособие для студентов, обучающихся по направлению Прикладная математика и информатика. / И. В. Парамонов; Яросл. гос. ун-т им. П. Г. Демидова, Науч.-метод. совет ун-та - Ярославль: ЯрГУ, 2015. - 47 с.

2. Парамонов И. В. Инженерия программных систем и комплексов на основе гибкой методологии разработки: учеб.-метод. пособие для студентов, обучающихся по направлению Прикладная математика и информатика. [Электронный ресурс]/ И. В. Парамонов; Яросл. гос. ун-т им. П. Г. Демидова, Науч.-метод. совет ун-та - Ярославль: ЯрГУ, 2015. - 47 с. <http://www.lib.uniyar.ac.ru/edocs/iuni/20150402.pdf>

Дополнительная литература:

1. Макконнелл, С., Совершенный код. Мастер-класс : пер. с англ / С. Макконнелл, М., Русская редакция, 2013, 868с

2. Мацяшек Л. А. Практическая программная инженерия на основе учебного примера. / Л. А. Мацяшек, Б. Л. Лионг; пер. с англ. А. М. Епанешникова, В. А. Епанешникова - М.: Бином. Лаборатория знаний, 2013. - 956 с.: ил.

3. «Интернет»:

1. <http://parallels.nsu.ru/~fat/subversion.ppt> - Лекции по инструментам управления конфигурацией, Иртегов Д.В

2. [http://zzet.org/git/learning/undev/coursify/2014/02/09/lecture-1-git-course-](http://zzet.org/git/learning/undev/coursify/2014/02/09/lecture-1-git-course-undev.html)

[undev.html](http://zzet.org/git/learning/undev/coursify/2014/02/09/lecture-1-git-course-undev.html) - Лекции о системе управления версиями Git

3. http://lib.custis.ru/Continuous_Integration - Непрерывная интеграция

4. http://www.redmine.org/projects/redmine/wiki/RusUser_Guide -

Руководство пользователя системы Redmine

5. <http://svnbook.red-bean.com/index.ru.html> - Управление версиями в Subversion

8. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине включает в свой состав специальные помещения:

- учебные аудитории для проведения занятий лекционного типа и практических занятий (семинаров);
- учебные аудитории для проведения лабораторных занятий;
- учебные аудитории для проведения групповых и индивидуальных консультаций,
- учебные аудитории для проведения текущего контроля и промежуточной аттестации;
- помещения для самостоятельной работы;
- помещения для хранения и профилактического обслуживания технических средств обучения.

Специальные помещения укомплектованы средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения занятий лекционного типа предлагаются наборы демонстрационного оборудования и учебно-наглядных пособий, хранящиеся на электронных носителях и обеспечивающие тематические иллюстрации, соответствующие рабочим программам дисциплин.

Помещения для лабораторных занятий и самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации.

Число посадочных мест в лекционной аудитории больше либо равно списочному составу потока, а в аудитории для практических занятий (семинаров) – списочному составу группы обучающихся.

Автор(ы) : Доцент кафедры КС И. В. Парамонов

**Приложение №1 к рабочей программе дисциплины
«Основы промышленной backend и TDD разработки на Java-
технологиях»**

**Фонд оценочных средств
для проведения текущей и промежуточной аттестации студентов
по дисциплине**

1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

1.1. Контрольные задания и иные материалы, используемые в процессе текущей аттестации

Контрольная работа по теме «Использование систем управления версиями»

1. Из каких подсистем состоит система контроля версий?
2. Чем отличаются централизованная и распределённая системы контроля версий?
3. Перечислите сценарии решения типичных задач промышленной разработки с использованием систем контроля версий.

Контрольная работа по теме «Проблемы управления конфигурацией и инструменты для их решения»

1. Дайте определение модели данных, модели архитектуры и модели подсистем.
2. Поясните понятия модуля и модульности. Зачем используют модули?
3. Роль управления конфигурации в управлении проектом.

Критерии оценивания:

Оценка	Критерии
Отлично Уровень формирования компетенций: высокий	Демонстрирует полное знание материалы по всем трем вопросам При ответе на вопрос рассуждает уверенно, аргументированно, приводит примеры.
Хорошо Уровень	Демонстрирует полное знание материалы по крайней мере в двух вопросах, ответ на третий может быть не полным, но

формирования компетенций: продвинутый	без грубых ошибок или присутствуют неточности при ответе на вопросы При ответе на вопрос рассуждает уверенно, но не всегда аргументированно, примеры приводит не всегда.
Удовлетворительно Уровень формирования компетенций: пороговый	Демонстрирует полное знание материалы по крайней мере в одном вопросе, ответы на другие вопросы не полные, но без грубых ошибок или присутствуют ошибки при ответе на все вопросы, но основной материал описан правильно. При ответе на вопрос рассуждает неуверенно, не всегда аргументированно, злоупотребляет воспроизведением теоретического материала, слабо фокусируясь на поставленном вопросе.
Неудовлетворительн о	Не знает ответы на два из трех вопросов или допускает грубые ошибки в ответах на все вопросы. Не понимает или плохо понимает суть поставленного вопроса. При ответе на вопрос демонстрирует неспособность логично рассуждать, злоупотребляет воспроизведением теоретического материала, не фокусируясь на поставленном вопросе.

Список вопросов к зачету

1. Понятие требования к ПО.
2. Роль требований в процессе промышленной разработки.
3. Источники и пути выявления требований.
4. Свойства качественных требований.
5. Анализ и тестирование требований.
6. Характеристика процесса управления требованиями и его роль в промышленной разработке.
7. Жизненный цикл требований.
8. Основные задачи управления требованиями.
9. Инструментальные средства управления требованиями.
10. Задача проектирования пользовательских интерфейсов и её место в процессе разработки ПО.
11. Понятие user experience (UX).
12. Этапы разработки пользовательского интерфейса.
13. Методы и инструментальные средства проектирования интерфейсов.
14. Методы и инструментальные средства тестирования интерфейсов.
15. Основные положения гибкой методологии.

16. Отличия гибкой методологии разработки от традиционной.
17. Возможности и область применимости гибкой методологии.
18. Отдельные методики в рамках гибкой методологии: XP, Scrum, Kanban, Lean.
19. Планирование выпусков ПО в рамках гибкой методологии разработки.
20. Долгосрочное и краткосрочное планирование выпусков.
21. Методики оценки трудозатрат на выполнение конкретных задач: покер планирования, определение размера футболки, оценка относительных масс, оценка на стене.
22. Задачи, решаемые трекером задач в рамках процесса разработки.
23. Жизненный цикл задачи в трекере.
24. Трекер задач как средство регламентации деятельности разработчика.
25. Виды систем контроля версий: централизованные и распределённые.
26. Коллективная разработка с использованием систем контроля версий.
27. Интеграция систем контроля версий в проектную инфраструктуру.
28. Сценарии решения типичных задач промышленной разработки с использованием систем контроля версий.
29. Задачи систем автоматической сборки в промышленной разработке.
30. Возможности систем автоматической сборки.
31. Понятие непрерывной интеграции, её преимущества и недостатки, область применения.
32. Инструменты непрерывной интеграции и сценарии их использования в промышленной разработке.
33. Назначение процедуры профилирования кода и её область применения.
34. Принципы работы профайлеров.
35. Анализ профиля и техники оптимизации, основанные на нём.
36. Назначение процедуры статического анализа кода и её область применения.
37. Возможности существующих статических анализаторов.
38. Роль статического анализа в промышленной разработке.

39. Задача управления конфигурацией и её актуальность в промышленной разработке.
40. Проблема воспроизводимости процесса и подходы к её решению.
41. Роль управления конфигурации в управлении проектом.
42. Инструменты управления конфигурацией и основные сценарии их применения.
43. Жизненный цикл программного кода в промышленной разработке.
44. Связь читаемости кода и его сопровождаемости.
45. Роль корпоративных стандартов на код в промышленной разработке.
46. Методы и приёмы обеспечения читаемости и сопровождаемости кода.
47. Инструментальные средства проверки кода на соответствие стандартам.
48. Задачи документирования программного обеспечения.
49. Инструментальные средства управления знаниями (howto, wiki и т. п.).
50. Требования предъявляемые к документации.
51. Инструменты для автоматической генерации документации.
52. Организация взаимодействия с пользователями и ресурсы, необходимые для неё.
53. Техническая поддержка по электронной почте и через call-центры.
54. Инструменты автоматизации технической поддержки.

2. Перечень компетенций, этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкалы оценивания

2.1. Шкала оценивания сформированности компетенций и ее описание

Оценивание уровня сформированности компетенций в процессе освоения дисциплины осуществляется по следующей трехуровневой шкале:

Пороговый уровень - предполагает отражение тех ожидаемых результатов, которые определяют минимальный набор знаний и (или) умений и (или) навыков, полученных студентом в результате освоения дисциплины. Пороговый уровень является обязательным уровнем для студента к моменту завершения им освоения данной дисциплины.

Продвинутый уровень - предполагает способность студента использовать знания, умения, навыки и (или) опыт деятельности, полученные при освоении дисциплины, для решения профессиональных задач. Продвинутый уровень превосходит пороговый уровень по нескольким существенным признакам.

Высокий уровень - предполагает способность студента использовать потенциал интегрированных знаний, умений, навыков и (или) опыта деятельности, полученных при

освоении дисциплины, для творческого решения профессиональных задач и самостоятельного поиска новых подходов в их решении путем комбинирования и использования известных способов решения применительно к конкретным условиям. Высокий уровень превосходит пороговый уровень по всем существенным признакам.

2.2. Перечень компетенций, этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования

Шкала оценивания компетенций:

Оценка в 100-балльной шкале	Оценка в 5-ти балльной шкале	Уровень компетенций сформированности
0-54 баллов	неудовлетворительно (не зачтено)	недостаточный
55-69 баллов	удовлетворительно (зачтено)	базовый
70-85 баллов	хорошо (зачтено)	повышенный
86-100 баллов	отлично (зачтено)	

Критерии оценивания компетенций:

Индикаторы достижения компетенций	Критерии оценивания компетенций		
	Недостаточный уровень	Базовый уровень	Повышенный уровень
ПК-5.1. Руководит разработкой архитектуры комплексных систем искусственного интеллекта со стороны заказчика	Неудовлетворительно (не зачтено)	Удовлетворительно (зачтено)	Хорошо или отлично (зачтено)

3. Методические рекомендации преподавателю по процедуре оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Целью процедуры оценивания является определение степени овладения студентом ожидаемыми результатами обучения (знаниями, умениями, навыками и (или) опытом деятельности).

Процедура оценивания степени овладения студентом ожидаемыми результатами обучения осуществляется с помощью методических материалов, представленных в разделе «Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций»

3.1 Критерии оценивания степени овладения знаниями, умениями, навыками и (или) опытом деятельности, определяющие уровни сформированности компетенций

Пороговый уровень (общие характеристики):

- владение основным объемом знаний по программе дисциплины;
- знание основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы без существенных ошибок;
- владение инструментарием дисциплины, умение его использовать в решении стандартных (типовых) задач;
- способность самостоятельно применять типовые решения в рамках рабочей программы дисциплины;
- усвоение основной литературы, рекомендованной рабочей программой дисциплины;
- знание базовых теорий, концепций и направлений по изучаемой дисциплине;
- самостоятельная работа на практических и лабораторных занятиях, периодическое участие в групповых обсуждениях, достаточный уровень культуры исполнения заданий.

Продвинутый уровень (общие характеристики):

- достаточно полные и систематизированные знания в объёме программы дисциплины;
- использование основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать выводы;
- владение инструментарием дисциплины, умение его использовать в решении учебных и профессиональных задач;
- способность самостоятельно решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в базовых теориях, концепциях и направлениях по изучаемой дисциплине и давать им сравнительную оценку;
- самостоятельная работа на практических и лабораторных занятиях, участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

Высокий уровень (общие характеристики):

- систематизированные, глубокие и полные знания по всем разделам дисциплины;
- точное использование терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать обоснованные выводы;

- безупречное владение инструментарием дисциплины, умение его использовать в постановке и решении научных и профессиональных задач;
- способность самостоятельно и творчески решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- полное и глубокое усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в основных теориях, концепциях и направлениях по изучаемой дисциплине и давать им критическую оценку;
- активная самостоятельная работа на практических и лабораторных занятиях, творческое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

3.2 Описание процедуры выставления оценки

В зависимости от уровня сформированности каждой компетенции по окончании освоения дисциплины студенту выставляется оценка «зачтено», «незачтено».

Показатели и критерии, используемые при выставлении оценки подробно описаны в разделе «Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций».

Высокий уровень формирования компетенций соответствует оценке «отлично» за практические задания и тест.

Продвинутый уровень формирования компетенций соответствует оценке «хорошо» за практические задания и тест.

Пороговый уровень формирования компетенций соответствует оценке «удовлетворительно» за практические задания и тест.

Оценка «отлично» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована на высоком уровне.

Оценка «хорошо» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на продвинутом уровне.

Оценка «удовлетворительно» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «неудовлетворительно» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

Оценка «зачет» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «незачтено» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

Приложение №2 к рабочей программе дисциплины «Основы промышленной backend и TDD разработки на Java-технологиях»

Методические указания для студентов по освоению дисциплины

Основными видами учебных занятий по курсу являются лекции и семинары. В рамках лекций предполагается максимально уйти от репродуктивного стиля обучения и широко применять интерактивные элементы. В частности, предполагается использовать технику проблемных вопросов и диалогичность, позволяя студентам самостоятельно воссоздавать новое знание, а не пассивно воспринимать уже подготовленную информацию. Данный подход выглядит для рассматриваемого курса особенно оправданным в силу специфики предмета, выраженной в множестве возможных точек зрения на объекты его предметной области и необходимости постоянного нахождения компромиссов в ходе практической деятельности участников процесса разработки.

2-3 лекции в рамках курса предполагается проводить приглашёнными специалистами индустриального партнёра, а также выпускниками магистерской программы прошлых лет, готовыми поделиться своими личными историями успеха в области ИТ-индустрии.

В рамках семинаров предполагается рассмотрение проблемных ситуаций, специально разработанных в рамках проекта на основе анализа процессов индустрии программного обеспечения. При этом предполагается широко использовать ролевые игры и метод кейсов, в рамках которых студенты смогут представить себя в ситуации, максимально приближенной к реальной, принять в этой ситуации решение и увидеть его последствия. Другим полезным в рамках разрабатываемого курса подходом может быть мозговой штурм в командах с последующим обсуждением результатов. Такие методы позволят не только сориентировать студентов на восприятие нового материала, но также помогут им в построении логических цепочек между изучаемыми техниками и процессами.

Для закрепления знаний предполагается активно использовать практику дистанционного выполнения домашних заданий в одной из систем управления обучением (LMS) с последующей оценкой студентами работ других участников по заранее подготовленному списку критериев. Также возможно финальное обсуждение предложенных решений для обсуждения границ их применимости и типичных ошибок.

Особенностью предлагаемого курса является использование интерактивных форм обучения, которые подразумевают активное участие со стороны студентов. На основании проявленного ими интереса и степени их вовлечённости в рассматриваемые ситуации можно сделать выводы о доступности предлагаемого материала: его сложности и понимании контекста студентами. Помимо внешней оценки, проводимой преподавателям, студентам также будет дополнительно предложено самостоятельно провести оценку, дать советы по улучшению по улучшению материала, например, в рамках обсуждения результатов выполнения заданий.

Учебно-методическое обеспечение самостоятельной работы студентов по дисциплине

Для самостоятельной работы особенно рекомендуется использовать учебную литературу, указанную в разделе № 7 данной рабочей программы.

Также для подбора учебной литературы рекомендуется использовать широкий спектр интернет-ресурсов:

1. Электронно-библиотечная система «Университетская библиотека online» (www.biblioclub.ru) - электронная библиотека, обеспечивающая доступ к наиболее востребованным материалам-первоисточникам, учебной, научной и художественной литературе ведущих издательств (*регистрация в электронной библиотеке – только в сети

университета. После регистрации работа с системой возможна с любой точки доступа в Internet.).

2. Для самостоятельного подбора литературы в библиотеке ЯрГУ рекомендуется использовать:

1. Личный кабинет (http://lib.uniyar.ac.ru/opac/bk_login.php) дает возможность получения on-line доступа к списку выданной в автоматизированном режиме литературы, просмотра и копирования электронных версий изданий сотрудников университета (учеб. и метод. пособия, тексты лекций и т.д.) Для работы в «Личном кабинете» необходимо зайти на сайт Научной библиотеки ЯрГУ с любой точки, имеющей доступ в Internet, в пункт меню «Электронный каталог»; пройти процедуру авторизации, выбрав вкладку «Авторизация», и заполнить представленные поля информации.

2. Электронная библиотека учебных материалов ЯрГУ (http://www.lib.uniyar.ac.ru/opac/bk_cat_find.php) содержит более 2500 полных текстов учебных и учебно-методических материалов по основным изучаемым дисциплинам, изданных в университете. Доступ в сети университета, либо по логину/паролю.

3. Электронная картотека «Книгообеспеченность» (http://www.lib.uniyar.ac.ru/opac/bk_bookreq_find.php) раскрывает учебный фонд научной библиотеки ЯрГУ, предоставляет оперативную информацию о состоянии книгообеспеченности дисциплин основной и дополнительной литературой, а также цикла дисциплин и специальностей. Электронная картотека «Книгообеспеченность» доступна в сети университета и через Личный кабинет.