

**МИНОБРНАУКИ РОССИИ**  
**федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**  
**Ярославский государственный университет им. П.Г.Демидова**

**УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ**  
**ПО ДИСЦИПЛИНЕ**

**Основы промышленной backend и TDD разработки на java-технологиях**

Направление подготовки (специальность):  
09.04.03 ПРИКЛАДНАЯ ИНФОРМАТИКА

Образовательная программа  
Искусственный интеллект в корпоративных информационных системах

**очная форма обучения**

Составитель:  
Парамонов И.В., к.ф.-м.н., доцент

г. Ярославль

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

### **Основная литература:**

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.

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

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-undev.html> - Лекции о системе управления версиями Git
3. [http://lib.custis.ru/Continuous\\_Integration](http://lib.custis.ru/Continuous_Integration) - Непрерывная интеграция
4. [http://www.redmine.org/projects/redmine/wiki/RusUser\\_Guide](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

**Учебно-методические указания и рекомендации  
к изучению тем лекционных и практических занятий, самостоятельной  
работе студентов**

**Очная форма обучения**

**Практические занятия**

№	Тема практического занятия	Количество часов
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	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
3	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.	3
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
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
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
<b>Итого:</b>		<b>18</b>

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.

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.