Управление разработкой программного обеспечения

**Разделы:**

Жизненный цикл программного обеспечения

Управление рисками, задачами и дефектами, системы контроля версий, лицензирование

Основы проектирования ПО

Архитектурные стили

**Содержание:**

1. Обзор управления разработкой ПО - включает в себя определение основных понятий, таких как управление разработкой ПО, процессы разработки, методологии и жизненный цикл разработки.
2. Процессы разработки ПО - классификация и описание основных процессов разработки программного обеспечения, таких как анализ требований, проектирование, кодирование, тестирование, развертывание и поддержка.
3. Методологии разработки ПО - обзор и сравнение наиболее популярных методологий, включая Agile, Scrum, Kanban, Waterfall и другие.
4. Жизненный цикл разработки ПО - описание основных этапов жизненного цикла разработки программного обеспечения: инициация, планирование, разработка, тестирование, внедрение и поддержка.
5. Оценка требований к ПО - процесс определения и анализа потребностей пользователей и функциональных требований к программному обеспечению.
6. Управление требованиями к ПО - методы сбора, документирования, анализа, оценки и управления требованиями для обеспечения успешного завершения проекта.
7. Анализ требований и спецификаций ПО - принципы и техники анализа требований для создания функциональной и нефункциональной спецификации программного обеспечения.
8. Проектирование ПО - определение и реализация архитектуры, структуры и интерфейсов программного обеспечения для удовлетворения требований и оптимизации разработки и поддержки.
9. Тестирование ПО - планирование, проведение и оценка качества тестирования для обнаружения и исправления дефектов и ошибок в программном обеспечении.
10. Управление проектами разработки ПО - техники и инструменты для планирования, контроля и координации работы над проектом, включая управление ресурсами, временем и бюджетом.
11. Управление рисками в разработке ПО - идентификация, анализ и управление потенциальными рисками, угрожающими успешному завершению проекта, такими как технические, организационные и финансовые риски.
12. Качество ПО и контроль качества - обеспечение и измерение качества программного обеспечения с использованием различных метрик и методов, таких как тестирование, контроль версий и анализ кода.
13. Разработка и внедрение стандартов и практик - создание и внедрение корпоративных стандартов и практик для команды разработчиков, включая стандарты кодирования, шаблоны проектирования, инструменты и среды разработки.
14. Организационные аспекты управления разработкой ПО - управление человеческими ресурсами и коммуникациями в команде разработки, включая роли и ответственность, коммуникации и разрешение конфликтов.
15. Этика и профессиональная ответственность - обсуждение этических аспектов разработки программного обеспечения и ответственности разработчика перед заказчиком, коллегами и обществом.

**Вопросы к зачету/экзамену**

1. Что такое управление разработкой программного обеспечения (ПО)?

2. Какие процессы включает в себя разработка ПО?

3. Какие методологии разработки ПО вы знаете и в чем их суть?

4. Из каких этапов состоит жизненный цикл разработки ПО?

5. Как происходит оценка требований к ПО?

6. Что включает в себя управление требованиями к ПО?

7. Как проводится анализ требований и спецификаций ПО?

8. Какие этапы включает в себя проектирование ПО?

9. Какие виды тестирования ПО существуют и в чем они заключаются?

10. Что включает в себя процесс управления проектами разработки ПО?

11. Как осуществляется управление рисками в разработке ПО?

12. Что такое качество ПО и как оно контролируется?

13. Для чего разрабатываются и внедряются стандарты и практики в команде разработчиков?

14. Какие организационные аспекты нужно учитывать при управлении разработкой ПО?

15. В чем заключается этика и профессиональная ответственность разработчика ПО?

16. Как проводится оценка требований к ПО в Agile-методологии?

17. Какие особенности имеет Scrum-фреймворк для управления разработкой ПО?

18. Как работает Kanban-метод в управлении разработкой ПО?

19. В чем отличие методологии Waterfall от других методологий разработки ПО?

20. Какие основные роли присутствуют в команде разработки ПО по Agile-методу?

21. Какие основные артефакты создаются в процессе Scrum-разработки ПО?

22. Какие инструменты используются для управления проектами в Kanban-методе?

23. В чем особенность методологии V-Model и где она применяется?

24. Что входит в обязанности менеджера проекта в области разработки ПО?

25. Какие техники применяются для управления рисками в разработке ПО?

26. В каких случаях применяется функциональное тестирование ПО и какие виды тестов используются?

27. Что такое регрессионное тестирование и когда оно проводится?

28. Что такое юнит-тестирование и для чего оно используется?

29. Какие метрики качества ПО вы знаете и как они используются?

30. В чем заключаются этические аспекты разработки программного обеспечения?

**Возможные практические задания**

1. Разработка плана проекта по созданию нового ПО с использованием Agile-метода.

2. Создание и ведение бэклога требований в Scrum для нового проекта по разработке ПО.

3. Применение Kanban-доски для управления задачами и ресурсами в процессе разработки ПО.

4. Планирование и проведение тестирования ПО с использованием разных видов тестирования.

5. Определение и анализ рисков в проекте разработки ПО, предложение стратегий их минимизации.

6. Оценка качества разработанного ПО с использованием метрик и инструментов контроля качества.

7. Разработка и применение стандартов кодирования и шаблонов проектирования для команды разработчиков ПО.

8. Управление командой разработки ПО, включая распределение задач, коммуникацию и разрешение конфликтов.

9. Оценка и улучшение этических аспектов разработки ПО, соблюдение профессиональной ответственности.

**Продвинутая веб-разработка**

**Содержание разделов дисциплины**

1. Архитектура высоконагруженных систем.

2. Масштабирование нагрузки.

3. Мониторинг.

4. Системы автоматизации развертывания, инструменты отладки.

5. Типовые архитектурные решения.

6. Знакомство с Linux системами.

7. Виртуализация, контейнеризация.

8. DevOps.

**Разработка высоконагруженных систем**

| Раздел 1. Введение в высокие нагрузки. | Тема 1. Проблемы высоких нагрузок. Нагрузочное тестирование. |
| --- | --- |
| Раздел 2. СУБД в высоконагруженных проектах. | Тема 2. Индексы. Репликация. |
| Раздел 2. СУБД в высоконагруженных проектах. | Тема 4. Транзакции в реляционных СУБД. Очереди и отложенное выполнение. In- Memory СУБД. |
| Раздел 3. Разработка высоконагруженных сервисов. | Тема 5. Микросервисы и протоколы. |
| Раздел 3. Разработка высоконагруженных сервисов. | Тема 6. Балансировка и отказоустойчивость. Мониторинг и алертинг. |
| Раздел 3. Разработка высоконагруженных сервисов. | Тема 8. Примеры оптимизации. |