

Теория:

1. Стандарты качества ПО: структура, назначение и применение в проектной деятельности.
2. Методы прогнозирования сроков и затрат на разработку.
3. Жизненные циклы ПО: сравнение моделей Waterfall, V-модель и Spiral.
4. Сравнительный анализ средств разработки: критерии выбора IDE и фреймворков
5. Система контроля версий Git: архитектура, основные команды, работа с ветками.
6. Организация работы в команде разработчиков: роли и распределение задач.
7. Метрики программного кода: LOC, сложность, связность. Практическое значение.
8. Управление рисками в IT-проектах: идентификация и планирование реагирования.
9. Требования к ПО: функциональные и нефункциональные. Правила формулирования.
10. Инструменты ревьюирования кода: цели, этапы, чек-листы.
11. Agile-методологии: Scrum, Kanban. Отличия от каскадных моделей.
12. Методы оценки бюджета разработки: базовые подходы.
13. Отладка программ: точки останова, пошаговое выполнение, анализ переменных.
14. Метрики качества процессов: скорость устранения дефектов, покрытие тестами.
15. Сравнительный анализ программных продуктов: критерии и оформление результатов.
16. Стандарты документирования ПО: виды и требования к структуре.
17. Жизненный цикл проекта: фазы, вехи, критический путь.

МДК.03.02 Управление проектами. Преподаватель: Здор Даниил Игоревич  
Московский приборостроительный техникум

18.Метрики прогноза сроков: базовые инструменты планирования.

19.Дизассемблирование и анализ бинарного кода: назначение и применение.

20.Подходы к менеджменту проектов: проектное, процессное, гибкое управление.

21.Инструменты статического анализа кода: принципы работы и типы дефектов.

22.Управление качеством в команде: код-стайл, линтеры, автоматические проверки.

23.Технические задания: требования к однозначности, проверяемости и полноте.

24.Методы организации командной работы: парное программирование, код-ревью.

25.Оптимизация кода: цели, базовые техники, влияние на производительность.

26.Сравнительный анализ методологий: когда применять Waterfall, Agile, Hybrid.

27.Метрики надёжности ПО: плотность дефектов, время между отказами.

28.Инструменты управления проектами: Jira, Trello, Redmine. Сравнение.

29.Git-воркфлоу: GitFlow, GitHub Flow. Преимущества и ограничения.

30.Стандарты качества процессов: CMMI, ISO 9001 в разработке ПО.

31.Анализ ПО из предметной области: методы и критерии оценки.

32.Прогнозирование качества ПО на основе метрик кода.

33.Управление изменениями в проекте: процесс запросов, влияние на сроки.

34.Метрики производительности: время отклика, использование ресурсов.

35.Рефакторинг: цели, базовые техники, влияние на читаемость.

36.Методы оценки рисков: качественные и количественные.

37.Сборка и отладка программы: этапы, логирование, базовые инструменты.

МДК.03.02 Управление проектами. Преподаватель: Здор Даниил Игоревич  
Московский приборостроительный техникум

38. Критерии выбора средств разработки: лицензирование, экосистема, поддержка.
39. Требования к безопасности ПО: конфиденциальность, целостность, доступность.
40. Управление командой разработчиков: мотивация и разрешение конфликтов.
41. Метрики сопровождаемости ПО: технический долг, дублирование кода.
42. Сравнительный анализ систем контроля версий: Git, SVN.
43. CI/CD: базовые этапы, автоматизация сборки и тестирования.
44. Прогнозирование сроков: метод трёх оценок (оптимистичная, реальная, пессимистичная).
45. Анализ зависимостей: внешние библиотеки, версии, совместимость.
46. Организация работы в распределённых командах: коммуникация и документирование.
47. Требования к производительности и масштабируемости: кэширование, балансировка.
48. Стандарты качества ПО: функциональность, надёжность, переносимость.
49. Метрики покрытия кода тестами: statement, branch coverage.
50. Управление конфигурацией проекта: версионирование, репродуцируемость.
51. Критерии приёмки ПО: соответствие ТЗ, результаты тестов, готовность к эксплуатации.
52. Методы оценки бюджета: резерв на риски, отслеживание фактических затрат.
53. Анализ кода специализированными средствами: линтеры, форматтеры.
54. Подходы к менеджменту проектов: PMI, PRINCE2, Agile. Сравнение принципов.
55. Метрики эффективности команды: velocity, lead time, cycle time.

МДК.03.02 Управление проектами. Преподаватель: Здор Даниил Игоревич  
Московский приборостроительный техникум

56. Сравнительный анализ ПО: бенчмарки, юзабилити, функциональный охват.

57. Документирование процессов разработки: регламенты, инструкции, wiki.

58. Методы оптимизации кода: алгоритмические, структурные.

59. Итоговая оценка качества проекта: критерии, отчётность, выводы.

60. Управление проектом инструментальными средствами: интеграция Git, трекеров, CI.

#### Практическая часть:

1. Анализ кода
2. Подсчёт количества точек принятия решений (ветвлений)
3. Заполнение таблиц для сравнения
4. Базовые команды Git Bash
5. Нахождение ошибок в коде (HTML, CSS, PHP)
6. Заполнение шаблона ТЗ
7. Сравнение фреймворков
8. Составление чек-листа для проверки функций языков программирования