

№	ТЕМА РЕФЕРАТА	Фамилия	Группа	Дата	Продолжительность
1	Категории программных ошибок. Распространённые программные ошибки.				
2	Визуальный контроль. Статическое и динамическое тестирования.				
3	Организация процесса тестирования программного продукта. Методика тестирования				
4	Модульное тестирование, тестирование элементов				
5	Интеграционное тестирование. Сравнительная характеристика методов тестирования.				
6	Монолитное тестирование				
7	Восходящее и нисходящее тестирования (драйвера)				
8	Нисходящее тестирования (заглушки)				
9	Модифицированный нисходящий метод)				
10	Метод скачка. Метод Сандвича. Модифицированный метод Сандвича.				
11	Эволюция методологий тестирования, таких как Agile, Waterfall, и их влияние на качество программных продуктов.				
12	Основные принципы Agile-тестирования, учитываемые при планировании проекта				
13	Методы тестирования наиболее эффективные в Agile-проектах				
14	Программирование на основе тестов (TDD)				
15	Инструменты для Agile-тестирования				
16	Юнит тест. NUnit, MSUnit (C#). Сравнить их возможности.				
17	Юнит тест. JUnit (Java).				

18	Юнит тест. Vocha, TestCaje, Chai (JavaScript).				
19	Юнит тест. Mock (C#).				
20	Юнит тест. QUnit (JavaScript).				
21	Тестирование изменений. Регрессионное тестирование. Инструменты регрессионного тестирования				
22	Системное тестирование. Категории тестов системного тестирования.				
23	Нефункциональное тестирование				
24	Тестирование правильности.				
25	Альфа-тестирование.				
26	Бета-тестирование.				
27	Гамма-тестирование.				
28	Тестирование производительности.				
29	Тестирование восстановления.				
30	Тестирование надежности.				
31	Тестирование стабильности.				
32	Нагрузочное тестирование. Основные принципы нагрузочного тестирования				
33	Тестирование на расширяемость функциональности.				
34	Стрессовое тестирование. Тестирование чувствительности.				
35	Тестирование безопасности				
36	Тестирование совместимости.				
37	Позитивное и негативное тестирования.				
38	Тестирование серого ящика.				
39	Инструменты и рамки для тестирования пользовательского интерфейса				
40	Тестирование конфигурации.				

41	Тестирование пользовательского интерфейса (оценить элементы дизайна: макет, цвет, шрифты, размеры шрифта, метки, текстовые поля, форматирование текста, титры, кнопки, списки, значки, ссылки и контент). FitNesse, iMacros, Coded UI.				
42	Ручное, автоматизированное и полуавтоматизированное тестирования (semiautomated testing). Выгоды и риски автоматизации				
43	Стохастическое тестирование. Критерии стохастического тестирования.				
44	Объектно-ориентированное тестирование.				
45	Отладка программного продукта.				
46	Надежность программы.				
47	Веб-тестирование - основные подходы и методы. Специфика тестирования веб-приложений, включая функциональное, нагрузочное тестирования и тестирование на совместимость.				
48	Бесплатный онлайн-сервис проверки сайта				
49	Веб-стандарты.				
50	Система отслеживания ошибок (Bug tracking).				
51	Система отслеживания ошибок с веб-интерфейсом Bugzilla				
52	Тестирование мобильных приложений. Специфика тестирования мобильных приложений, включая адаптацию к различным платформам и устройствам.				
53	Тестирование мобильных приложений на базе ОС Android. Анализ методов тестирования мобильных приложений на базе ОС Android				

	и результаты анализа их эффективности.				
54	Инструменты для тестирования мобильных приложений (эмуляторы: Android SDK and AVD Manager).				
55	Инструменты для тестирования мобильных приложений (эмуляторы: Mobi One Developer).				
56	Инструменты для тестирования мобильных приложений (эмуляторы: Electric Mobile Studio 2012).				
57	Инструменты для тестирования мобильных приложений (Сервисы для бета-тестирования: The Beta Family).				
58	Инструменты для тестирования мобильных приложений (сбор статистики: Google Analytics, Flurry, BugSence, Apsalar).				
59	Частые дефекты в тестировании мобильных приложений.				
60	Лучшие инструменты для тестирования интерфейса: REST-Assured (Java); Postman (для Mac, Linux, Windows).				
61	Инструменты автоматизированного тестирования (Selenium (C#, Java, Ruby, Python, JavaScript)).				
62	Инструменты автоматизированного тестирования (Telerik Test studio (C#)).				
63	Инструменты автоматизированного тестирования (Telerik Testing Framework (C#)).				
64	Инструменты автоматизированного тестирования (Coded UI (C#)).				
65	Инструменты автоматизированного тестирования (Sikuli (Ruby, Python))				

66	Инструменты, используемые в тестировании. Выгоды и риски использования типовых инструментов				
67	Тестирование антивирусного программного обеспечения. Анализ методов тестирования антивирусного программного обеспечения и результаты анализа их эффективности.				
68	Тестирование в облачных средах. Описание подходов и методов тестирования программных продуктов в облачных средах, таких как Amazon Web Services (AWS), Microsoft Azure и Google Cloud Platform				
69	Тестирование и развертывание облачных приложений. Вызовы, связанные с тестированием и развертыванием программного обеспечения в облачных средах				
70	Тестирование искусственного интеллекта. Подходы и методы тестирования программных продуктов, которые используют искусственный интеллект, такие как машинное обучение, глубокие нейронные сети и другие технологии.				
71	Тестирование искусственного интеллекта и машинного обучения. Особенности тестирования программ, использующих технологии искусственного интеллекта и машинного обучения.				
72	Методы тестирования, используемые в быстрорастущих проектах с гибкими методиками разработки				
73	Тестирование итеративных и гибких методологий разработки. Специфика тестирования в контексте гибких методологий, таких как Scrum или Kanban.				

74	Тестирование игрового программного обеспечения. Уникальные аспекты тестирования компьютерных игр, включая функциональное и игровое тестирования, тестирование на совместимость.				
75	Тестирование виртуальной реальности. Описание подходов и методов тестирования программных продуктов, которые используют виртуальную реальность, такие как виртуальные реальности Oculus, HTC Vive и другие устройства.				
76	Использование искусственного интеллекта и машинного обучения для автоматизации создания и выполнения тестов, а также прогнозирования потенциальных ошибок				
77	Low-Code и No-Code решения				
78	Тестирование Low-Code и No-Code приложений				
79	Типы приложений, лучше всего подходящие для разработки ПО с использованием low-code				
80	Инструменты тестирования low-code наиболее популярные на рынке				
81	Кроссплатформенное тестирование				