



Автономная некоммерческая организация
дополнительного профессионального образования
(АНО ДПО «Инфосфера»)
Центр профессиональной подготовки
ИНСТИТУТ ПРОГРАММНЫХ СИСТЕМ

Рабочая программа дисциплины
«Контроль качества ПО»

Разработал:
преподаватель ИПС
АНО ДПО «Инфосфера»
А. Смоленцева

Йошкар-Ола, 2017

Пояснительная записка

Целью курса является изучение основных аспектов тестирования программного обеспечения, виды, уровни и стадии тестирования. Будущие тестировщики познакомятся со статическим и динамическим, ручным и автоматизированным тестированием. Также изучается тестирование юзабилити и тестирование безопасности. Значительное внимание уделяется проектной документации. По окончании обучения студенты смогут самостоятельно составлять чек листы, тест кейсы, создавать отчеты об ошибках и отчеты о результатах тестирования. Студенты также получат навыки работы с баг треккинговой системой.

Категория слушателей: Слушателями курса могут быть лица от 18 лет, имеющие начальное, среднее или высшее профессиональное образование, а также выпускники общеобразовательных школ.

Срок обучения: 4 месяца (36 академических часа)

Режим занятий: 4 ак. часа в неделю

Планируемые результаты обучения

Разработка тестовых наборов данных предполагает формулирование правил создания, структуры и требований к тестовым наборам данных, подготовка наборов данных, используемых в процессе проверки работоспособности. Решаются задачи с элементами проектирования. Не требуется взаимодействие с другими программистами, системным аналитиком и архитектором программного обеспечения. Полученные результаты представляются руководителю разработки программного обеспечения.

Проверка работоспособности программного обеспечения осуществляется программистом на основании функциональных требований и технических спецификаций на программное обеспечение самостоятельно путем разработки и исполнения сценариев проверки с применением методов и технологий тестирования и ревьюирования кода.

В ходе проверки работоспособности осуществляется анализ нарушений требований к программному обеспечению, принимаются решения и вносятся изменения в программный код. Программист несет ответственность за решение поставленных задач или результат деятельности группы работников. Полученные результаты представляются руководителю разработки программного обеспечения.

Рефакторинг и оптимизация программного кода осуществляется на основании функциональных требований и технических спецификаций на программное обеспечение, в том числе с использованием специализированных программных средств. Программист несет

ответственность за решение поставленных задач или результат деятельности группы работников. Полученные результаты представляются руководителю разработки программного обеспечения.

Исправление дефектов, зафиксированных в базе данных дефектов сводится к воспроизведению дефектов, зафиксированных в базе данных дефектов, установлению причин возникновения дефектов, внесению изменений в программный код для устранения выявленных дефектов. Программист несет ответственность за решение поставленных задач или результат деятельности группы работников. Полученные результаты представляются руководителю разработки программного обеспечения.

Учебно-тематический план

№	Наименование разделов	Всего часов	В том числе		Форма контроля
			Лекции	Практ. занятия	
1	Введение в тестирование ПО	1	0.5	0.5	Фронтальный опрос
2	Тестирование ПО и его связь с жизненным циклом ПО	1	0.5	0.5	Контрольная работа
3	Планирование тестовых испытаний	1	0.5	0.5	Контрольная работа
4	Тестирование документации и требований	1	0.5	0.5	Контрольная работа
5	Разработка тестов	1	0.5	0.5	Контрольная работа
6	Поиск и документирование дефектов	2	1	1	Контрольная работа
7	Документирование результатов тестирования	2	1	1	Контрольная работа
8	Автоматизация тестирования	2	1	1	Контрольная работа
9	Тестирование безопасности	2	1	1	Контрольная работа
10	Тестирование юзабилити	2	1	1	Контрольная работа
11	Модульное тестирование	2	1	1	Контрольная работа

12	Итоги и перспективы	1	0.5	0.5	Фронтальный опрос
	Итого	36	18	18	

Содержание курса

Тема 1. Введение в тестирование ПО

- Введение
- Тестирование и качество
- Направления тестирования
- Этапы тестирования
- Методы тестирования (метод чёрного, белого и серого ящика)
- Функциональное тестирование
- Виды тестирования
- Уровни тестирования

Тема 2. Тестирование ПО и его связь с жизненным циклом ПО

- Стадии процесса тестирования
- Жизненный цикл ПО и его связь с тестированием
- Сроки тестирования (когда начинать и прекращать тестирование)
- Основные сложности тестирования
- Психологические аспекты тестирования
- Необходимые знания тестировщика

Тема 3. Планирование тестовых испытаний

- Введение
- Связь планирования тестовых испытаний с жизненным циклом ПО
- Области компетенции тестировщиков
- Задачи планирования и тестового плана
- Необходимые действия на стадии планирования
- Arteфакты, создаваемые на стадии планирования
- Сложности планирования
- Риски
- Секции тестового плана
- Критерии хорошего тестового плана
- Преимущества хорошего тестового плана
- Рассмотрение пример тестового плана

Тема 4. Тестирование документации и требований

- Введение
- Определение понятия “требование”

- Требования к продукту и процессу
- Важность требований
- Распределение дефектов по рабочим продуктам
- Виды документации, подвергаемой тестированию
- Уровни требований
- Типы требований
- Группа функциональных требований
- Группа нефункциональных требований
- Кто создаёт и использует требования
- Связь требований и архитектуры проекта
- Связь анализа требований и потоков программной инженерии
- Источники требований
- Пути выявления требований (интервью, анкетирование, наблюдение, самостоятельное описание, семинары, прототипирование)
- Видение продукта и границы проекта
- Документирование требований. Секции документа “Требования”
- Тестирование требований
- Характеристики хорошего требования
- Характеристики хорошего набора требований
- Каких требований быть не должно
- Проблемы с требованиями
- Проблемы с наборами требований
- Работа с требованиями (техники и способы)

Тема 5. Разработка тестов

- Разновидности тестов
- Пример разработки тестов: задача о треугольнике
- Классы эквивалентности и граничные условия
- Рекомендации по разработке тестов
- Документирование тестов. Тест-кейсы
- Свойства тест-кейсов
- Язык написания тестов
- Критерии хорошего тест-кейса
- Тестовые сценарии
- Техники ускорения написания тестов
- Шаги разработки тестов
- Пример разработки тестов: Notepad

Тема 6. Поиск и документирование дефектов

- Введение

- Определения дефекта
- Отчёты об ошибках
- Примеры отчётов об ошибках
- Атрибуты отчётов об ошибках
- Плохие и хорошие отчёты об ошибках
- Рекомендации по написанию отчётов об ошибках
- Преимущества хороших отчётов об ошибках
- Баг-трекинг-системы
- Практическое задание

Тема 7. Документирование результатов тестирования

- Введение
- Определения отчёта о результатах тестирования
- Цели написания отчёта о результатах тестирования
- Структура отчёта о результатах тестирования, примеры.
- Потребности проектной команды в отчёте о результатах тестирования
- Финальный отчёт о результатах тестирования
- Рассмотрение примеров отчётов о результатах тестирования

Тема 8. Автоматизация тестирования

- Введение
- Понятие автоматизации
- Области автоматизации
- Наилучшие и наихудшие тесты для автоматизации
- Памятки автоматизации
- Почему не всё следует автоматизировать
- Условия успешной автоматизации
- Преимущества автоматизации
- Недостатки автоматизации
- Лучшие идеи автоматизации
- Технология Record&Playback
- Функциональная декомпозиция
- Тестирование под управлением данными
- Тестирование под управлением ключевыми словами
- Обзор средств автоматизации
- Средство автоматизации SilkTest

Тема 9. Тестирование безопасности

- Введение
- Определения тестирования безопасности

- Терминология тестирования безопасности
- Источники проблем информационной безопасности
- Принципы безопасности и защищённости
- Категории объектов защиты
- Уровни информационной безопасности
- Поиск уязвимостей
- Типы тестов уязвимости
- Атаки через среду
- Атаки через ввод
- Атаки через данные и логику
- Программное обеспечение для автоматизации тестирования безопасности
- Мифы безопасности
- Практическое рассмотрение средств тестирования безопасности

Тема 10. Тестирование юзабилити

- Введение
- Определения юзабилити и тестирования юзабилити
- Тезисы тестирования юзабилити
- Виды тестирования юзабилити
- Контекстные исследования
- Оценочные листы
- Плюралистическая проработка
- Протоколы самоотчёта
- Фокусные группы
- Экспертиза компонентов

Тема 11. Модульное тестирование

- Введение
- Термины и определения модульного тестирования
- Стратегия модульного тестирования
- Цель модульного тестирования
- Планирование модульного тестирования
- Распределение обязанностей при модульном тестировании
- Место модульного тестирования в программировании
- Уровни модульного тестирования
- Метрики покрытия
- Оболочки модульного тестирования
- Семейство XUnit
- Расширения XUnit

Тема 12. Итоги и перспективы

- Классические ошибки тестировщиков
- Портрет тестировщика
- Перспективы карьерного роста тестировщика

Методические рекомендации

Для реализации заявленных учебных целей используются методологические технологии, реализующие деятельностный, личностно-ориентированный, практико-ориентированный подходы. В ходе изучения курса используются технологии проблемного, эвристического, модульно-рейтингового, опережающего, знаково-контекстного, проектного, дифференцированного, группового, личностно-ориентированного обучения, информационные и дистанционные технологии.

Основными видами учебной работы (стратегическими технологиями) являются лекции. Также делается упор на практические занятия. Лекция классическая предусматривает (ЛК) сообщение темы, плана лекции, списка литературы, изложение информации под запись слушателям (монолог, диалог в проблемном изложении), фронтальная проверка знаний, подведение итогов, выводы, ответы на вопросы слушателей, возникшие в ходе лекции или по итогам самостоятельной работы над теоретическим материалом). Практические работы предусматривают последовательное выполнение заданий слушателями в классе, направленных на упорядочивание, систематизацию теоретических знаний; выполнение упражнений на запоминание, осмысление и оперирование языковой терминологией; перевод знаний на уровень практических умений и навыков. Самостоятельная работа включает процедуры самообучения слушателей курса, инициируемые и управляемые со стороны преподавателя в режиме их подготовки к лекциям, семинарам, практическим работам, сдаче экзаменов и зачетов; процедуры самообучения в условиях свободы выбора заданий для самостоятельного освоения новых знаний, овладения умениями, используя весь арсенал современных источников информации (учебники, учебные пособия, ресурсы интернета, собственный опыт); консультирование преподавателя.

При организации учебных занятий используются активные методы обучения (работа в группах, взаимообучение, самоконтроль, индивидуальные задания дифференцированной сложности).

В процессе обучения возможно использование следующих тактических технологий: лекция классическая, лекция проблемная, лекция-визуализация, лекция-диалог, аудиторно-практическое занятие классическое, практикум-лабораторная работа, самообучение.

Пособия по изучению курса

1. Lessons Learned in Software Testing: A Context-Driven Approach. Cem Kaner, 2001. – 320 p.
2. Beautiful Testing: Leading Professionals Reveal How They Improve Software (Theory in Practice). Tim Riley, Adam Goucher, 2009. – 352 p.
3. Software Testing (2nd Edition). Ron Patton, 2005. – 408 p.

4. Testing Computer Software, 2nd Edition. Cem Kaner, Jack Falk, Hung Q. Nguyen, 1999. – 480 p.
5. Managing the Testing Process: Practical Tools and Techniques for Managing Hardware and Software Testing. Rex Black, 2009. – 672 p.
6. Software Testing: A Craftsman's Approach, Third Edition. Paul C. Jorgensen, 2008. – 440 p.
7. Implementing Automated Software Testing: How to Save Time and Lower Costs While Raising Quality. Elfriede Dustin, Thom Garrett, Bernie Gauf, 2009. – 368 p.
8. Agile Testing: A Practical Guide for Testers and Agile Teams. Lisa Crispin, Janet Gregory, 2009. – 576 p.

Контрольные задания

Контрольные вопросы для самостоятельной оценки качества освоения дисциплины

1. Введение. Основные понятия. Виды тестирования. Уровни тестирования.
2. Процесс тестирования. Жизненный цикл ПО.
3. Планирование тестовых испытаний. Риски.
4. Тестирование документации и требований.
5. Разновидности тестов. Тест-кейсы. Тестовые сценарии. Разработка тестов.
6. Отчеты об ошибках.
7. Документирование результатов тестирования.
8. Автоматизация тестирования.
9. Средства тестирования безопасности.
10. Тестирование юзабилити.
11. Модульное тестирование.
12. Классические ошибки тестировщиков.

По итогам обучения выдается документ следующего образца