



Professional Solutions
Around The Globe



Qulix Systems

**КУРС «ВВЕДЕНИЕ В АВТОМАТИЗИРОВАННОЕ
ФУНКЦИОНАЛЬНОЕ ТЕСТИРОВАНИЕ
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ»**

Программа теоретических и практических занятий

Название документа	Программа курса: Введение в автоматизированное функциональное тестирование ПО
Версия документа	1.0.0

Содержание

ПРОГРАММА ТЕОРЕТИЧЕСКИХ ЗАНЯТИЙ.....	3
ПРОГРАММА ПРАКТИЧЕСКИХ ЗАНЯТИЙ.....	4

ПРОГРАММА ТЕОРЕТИЧЕСКИХ ЗАНЯТИЙ

Лекция 1. Общие сведения об автоматизации тестирования программного обеспечения.

- 1.1 Место автоматизации функционального тестирования в жизненном цикле процесса контроля качества ПО;
- 1.2 Цели и задачи автоматизации функционального тестирования ПО;
- 1.3 Проблемы автоматизированного тестирования;
- 1.4 Принятие решения о внедрении автоматизации;
- 1.5 "Плюсы" и "минусы" автоматизированного тестирования.

Лекция 2. Этапы автоматизированного функционального тестирования.

- 2.1 Место автоматизированного тестирования в процессе разработки.

Лекция 3. Этап исследования и "пилотный проект".

- 3.1 Выбор инструмента автоматизации;
- 3.2 Планирование и реализация "пилотного проекта".

Лекция 4. Этап анализа и планирования автоматизации тестирования.

- 4.1 Особенности разработки скриптов автоматизации тестирования в зависимости от тестируемого приложения;
- 4.2 Подходы к автоматизации тестирования;
- 4.3 Документы, сопровождающие автоматизированное тестирование.

Лекция 5. Этап разработки архитектуры тестов.

- 5.1 Создание базовых функций;
- 5.2 Структура хранения скриптов;
- 5.3 Данные и параметризация.

Лекция 6. Этап создания автоматизированного сценария.

- 6.1 Запись/отладка скриптов;
- 6.2 Контрольные точки;
- 6.3 Библиотеки функций;
- 6.4 Библиотеки объектов;
- 6.5 Исключительные ситуации.

Лекция 7. Этап эксплуатации скриптов.

Лекция 8. Практическая работа на примере WinRunner и QTP.

ПРОГРАММА ПРАКТИЧЕСКИХ ЗАНЯТИЙ

Задание 1. Запись и воспроизведение (**playback**) теста на WR для тестового приложения

- 1.1. Выбор режима записи теста: отличия контекстно-зависимого (Context Sensitive mode) и аналогового (Analog mode) режимов
 - 1.1.1. Запись теста в контекстно-зависимом режиме
 - 1.1.2. Запись теста в аналоговом режиме
- 1.2. Запуск тестов и просмотр тестовых результатов

Задание 2. Понимание карты объектов

- 2.1. Объяснение различий и преимуществ глобальной и индивидуальной для каждого теста карты объектов
- 2.2. Использование **RapidTest Script Wizard (Insert > RapidTest Script Wizard)**
- 2.3. Подключение карты объектов к тесту
- 2.4. Изменение свойств объектов и добавление объектов в карту во время прохождения теста (run time)
- 2.5. Использование полного описания объекта для обращения к соответствующему элементу управления на форме без использования карты объектов (object full description)
- 2.6. Использование регулярных выражений в карте объектов

Задание 3. Отладка автоматического теста (**Debug Run Mode**)

- 3.1. Окно просмотра значений переменных (**watch list**)
- 3.2. Точки прерывания выполнения теста (**break points**)
- 3.3. Цепочки вызовов (**call chain**)

Задание 4. Написание теста с использованием контрольных точек (check points)

- 4.1. Использование стандартных мастеров (**wizards**) WR
 - 4.1.1. Контрольные точки для элементов управления (**GUI checkpoints**)
 - 4.1.2. Контрольные точки для графических объектов (**Bitmap checkpoints**)
- 4.2. Ознакомление с основными функциями для использования контрольных точек: win_exists; obj_exists; obj_get_desc; win_get_desc

Задание 5. Синхронизация действий в автоматических тестах

- 5.1. Изменение настроек синхронизации WR (**Tools > General Options; Run > Synchronization**)
- 5.2. Использование стандартного мастера WR (**Insert > Synchronization Point**)
- 5.3. Использование функций wait; web_sync

Задание 6. Написание теста управляемого данными (**data-driven**)

6.1. Использование встроенного мастера WR (**Table > Data Driver Wizard**)

6.2. Разбор известных проблем работы с управляемыми данными тестами, описанных в файле Mercury Interactive\WR\dat\readme.wri из стандартной поставки WR

Задание 7. Организация структуры тестов

7.1. Создание библиотек (**compiled module**). Подключение библиотек к автоматическому тесту. Принципы организации и дифференциации библиотек

7.2. Организация хранилища (repository) автоматических тестов на уровне файловой системы. Формирование логически выделенных наборов (suite) тестов для запуска.

Задание 8. Запуск тестов

8.1. Запуск тестов в пакетном режиме (**batch mode**)

8.2. Запуск тестов из командной строки

8.3. Создание теста с параметрами

Задание 9. Тестирование web-приложений

Задание 10. Работа с виртуальными объектами

Задание 11. Определение и использование сценариев восстановления (**recovery scenarios**)

11.1. Определение простого сценария восстановления (simple recovery scenarios)

11.2. Определение составного сценария восстановления (compound recovery scenarios)

Задание 12. Решения типовых задач

12.1. Запуск наборов тестов (suite), используя технику тестов управляемых данными в пакетном режиме из командной строки (Shell script)

12.2. Авто заполнение формы: написание универсального теста для заполнения любой формы приложения с использованием техники тестов управляемых данными

12.3. Основы работы с БД через инструменты автоматизированного тестирования

Задание 13. Разбор готовых примеров скриптов на SilkTest. Сравнение механизмов работы SilkTest и WR

Задание 14. Разбор готовых примеров скриптов на QuickTest Professional. Сравнение механизмов работы QTP и WR