**Лабораторная работа №7**

**“Прохождение основных этапов тестирования ”**

**Цель работы:** получить опыт работы в команде и научиться использовать на практике полученные знания о тестировании программного обеспечения.

**Продолжительность работы** - 8 часов.

**Введение**

Сотрудники ИТ-сферы отличаются особым способом мышления и подходом к решению тех или иных проблем. Очень важно не следовать клише, а развивать и модифицировать новые идеи и подходы.

Гибкая методология как раз способствует генерации новых взглядов и подходов к работе, как разработчиков, так и тестировщиков. Опытный QA-специалист всегда знает или может быстро разобраться в том, что и как надо протестировать.

В гибкой команде существует распределение обязанностей между всеми участниками проекта. **Тестирование ПО** или [**веб-тестирование**](http://ru.qatestlab.com/technologies/types-of-applications/web-application-testing/#веб-тестирование) с применением принципов гибкости характеризуется некоторыми особенностями, которые определяют таски самого тестировщика.



## Каковы обязанности тестировщика в гибкой команде?

Необходимо проводить [**исследовательское тестирование**](http://ru.qatestlab.com/services/No-Documentation/exploratory-testing/#исследовательское%20тестирование) на каждом спринте. С помощью такого нетривиального подхода можно обнаружить баги, которые не смогли бы найти даже автоматизированные тест-скрипты.

Следует выполнять [**ручное тестирование**](http://ru.qatestlab.com/services/We-Are-Professionals-in/Manual-Testing/#ручное%20тестирование), когда все пользовательские истории закончены. Это не только поможет обнаружить баги, но и покажет, какие тесты стоит автоматизировать на следующих спринтах.

Бизнес-аналитики часто нуждаются в помощи тестировщиков, чтобы формулировать критерии допустимости для дальнейших спринтов.

**Автоматизированное тестирование** играет ключевую роль для гибкой разработки. Автотесты предупреждают возникновение множества проблем и помогают сэкономить время.

Тестировщики принимают активное участие в планировании текущих и следующих спринтов.

АРI-автоматизация также входит в компетенцию тестировщика. Большинство специалистов владеют опытом работы с SoapUI, который может очень пригодиться для АРI-автоматизации.

Упомянутые выше обязанности являются основными для тестировщика в гибкой команде. Но не исключено, что специалист будет выполнять также и другие таски, характерные для разработки по гибкой методологии.

# Порядок выполнения лабораторной работы

1. Необходимо разбиться на команды по 2-3 человека
2. Запустить и ознакомиться с программой, которая представлена в папке M:\methodic\%%
3. Ознакомиться с таблицей 1, которая представлена ниже. В ней указаны критерии, по которым будут выставлены оценки в ходе выполнения лабораторной работы.
4. Написать подробный тест-план, оформить его в отчете по лабораторной работе.
5. Написать тест-кейсы для проекта (юнит тесты, нагрузочные, ручные). За тест-кейсы, которые будут уникальными и произведут впечатление, будет начислены дополнительные баллы.
6. Оптимизировать работу программы. В программе предоставлен метод с потоками и критическими секциями.
7. Найти ошибки в юнит-тестах, представленных в программе, и исправить их
8. Защитить перед преподавателем свой отчет.

**Критерии оценки выполнения лабораторной работы**

Каждая команда будет оценена по критериям, указанным в таблице 1.

Максимальное количество баллов по каждому из критериев - 10.

Минимальное количество баллов по каждому из критериев – 0.

Необходимо соблюдать порядок сдачи, который описан в вашем тест-плане.

За быстроту дополнительных баллов не предусматривается, важно **качество** работы.

Команда, набравшая по итогу наибольшее количество баллов, получает автомат.

Таблица 1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Ошибки в юнит-тестах | Тест-кейсы | Оптимизация программы | Взаимодействие участников команды друг с другом | Качество тест-плана |
| Команда 1 |  |  |  |  |  |
| Команда 2 |  |  |  |  |  |
| Команда 3 |  |  |  |  |  |