TEST AUTOMATION ACADEMY
AUTOMATION BASICS

ARTIN
SOLUTIONS

TEST AUTOMATION

> Not only "automatic run of tests"

> Using tools that not only run but also prepare environment, get results

> Software that run tests have to be other than test product

TEST AUTOMATION - CONTEXT

> It have to be based on building solution for software testing
>> Supported systems
>> Documentation
>> Test cases
>> Data
>> Monitoring
>> reporting

TEST AUTOMATION - APPROACH

> It is not only GUI / APP testing

> We need to look at the product using more than one interface

GOALS OF TEST AUTOMATION

> Increase of testing effectivity

> Increase test coverage

> Run tests that are hard for manual tester to automate

> Increase frequency of testing

DISADVATAGES OF AUTOMATION

> It is not free

> Big start costs

> Testing and programming skills are needed

> Support of automation testing is needed

> Complexity of tests can be too big

> No manual testing (focus only on automation)

LIMITATIONS

> Not all can be automated

> We can only check expected results, that are checkable by machines

TEST AUTOMATION
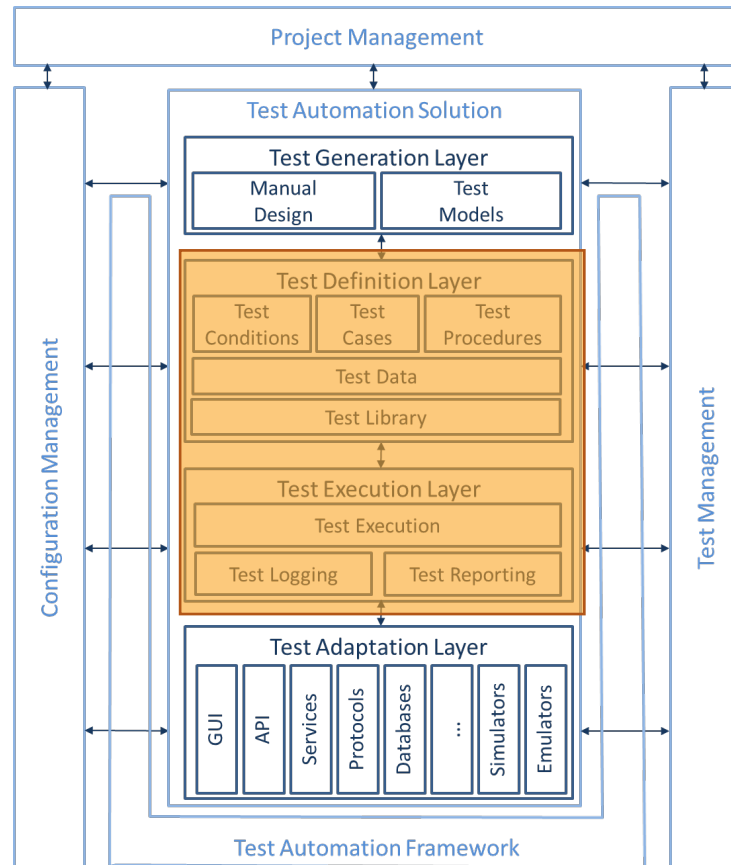
Generation
- How to choose what to test

Definition
- How to write data, conditions etc..

Execution
- Reporting and logging

Adaptation
- Connectors that are used for test automation

TEST AUTOMATION TYPES

> Capture/playback

> Linear scripting

> Structured scripting

> Data driven testing

> Keyword driven

> Process driven

> Model based testing

CAPTURE / PLAYBACK

> Record scripts
> Make changes
> Run tests

> When SUT changes, record and playback again

LINEAR SCRIPTING

> Create test automation scripts

> Every test is set of repeated steps

> New test -> new scripts

> On change only part of scripts is changed, but in every script

STRUCTURED SCRIPTING

> Create test automation scripts

> Start to use functions for repeated steps

> New test -> new set of functions
  > On change only functions are changed

> When I change functions, I'll fix all tests at once

DATA DRIVEN TESTING

> Create test automation scripts

> Start to use functions for repeated steps

> New test -> new data for test, definition is not changed

> On change functions can be changed, or simply data added

KEYWORD DRIVEN TESTING

> Create test automation scripts

> Start to use functions for repeated steps

> DSL will be defined for test creation. DSL is easy recognized by tester, manager or everbody who gets tests for work with.

> On change, keywords are reworked. I'm chaning only specific keywords, not full implementation

PROCESS DRIVEN TESTING

Workflow

Log In
CalculateTax
Log Out

Keyword file

login (name, password): boolean
calculate_tax(): float
logout()

Test data

Log In
    Pooh, 123: true
    Piglet, 456: true
    Roo, 789: false

CalculateTax
    Pooh: 125,7
    Piglet: 80,9
    Roo: none

Process-driven script

Automatic combination of data,
workflow and keywords.

MODEL BASED TESTING

> Model of system is the basis of testing

> This model is translated using tool

> Tests are designed using this model

> On change, model is changed and tests are regenerated using new model

APPLICATION SEEN BY TESTERS

ARTIN
SOLUTIONS

MANUAL TESTER

> Editboxes

> Links

> Buttons

> Translations

> Pictures

> Screen layout

DATA FOR TESTING

> Editboxes
  > Valid/invalid input

> Links
  > Valid/invalid links

> Buttons
  > Translation chage
  > usage

> Pictures
  > Layout
  > colors

AUTOMATION TESTER

> Input

> Href

> Button

> Text

> Img

- xpath
- id
- name

- Language

- Data generation

- Framework

- Any automation at all?

- CI / CD

- Environment

- HW resources

WHAT IS TEST AUTOMATION

> Programming using design patterns

> Focused on e2e testing or any other kind of testing

> If manual tester see application as end user, automation
tester have to see under the hood

> Don't fear to start again

MAIN GOALS OF TEST AUTOMATION TESTER

> We can never replace manual testing

> Our goal is to fill in the gap, regression testing and let manual testers do their job

> Let tester use our scripts to create data or to use keyword driven approach without us

> Using test analytics to create 'early warning' system

FUNCTIONAL VS NON-FUNCTIONAL TESTING

ARTIN
SOLUTIONS

TOOLS OVERVIEW

ARTIN
SOLUTIONS

MOST KNOWN TOOLS

> Selenium
>> Outside of browser automation

> Cypress
>> In browser automation tool

> Puppeteer
>> Outside of browser automation

> Ranorex
>> Outside of browser automation