

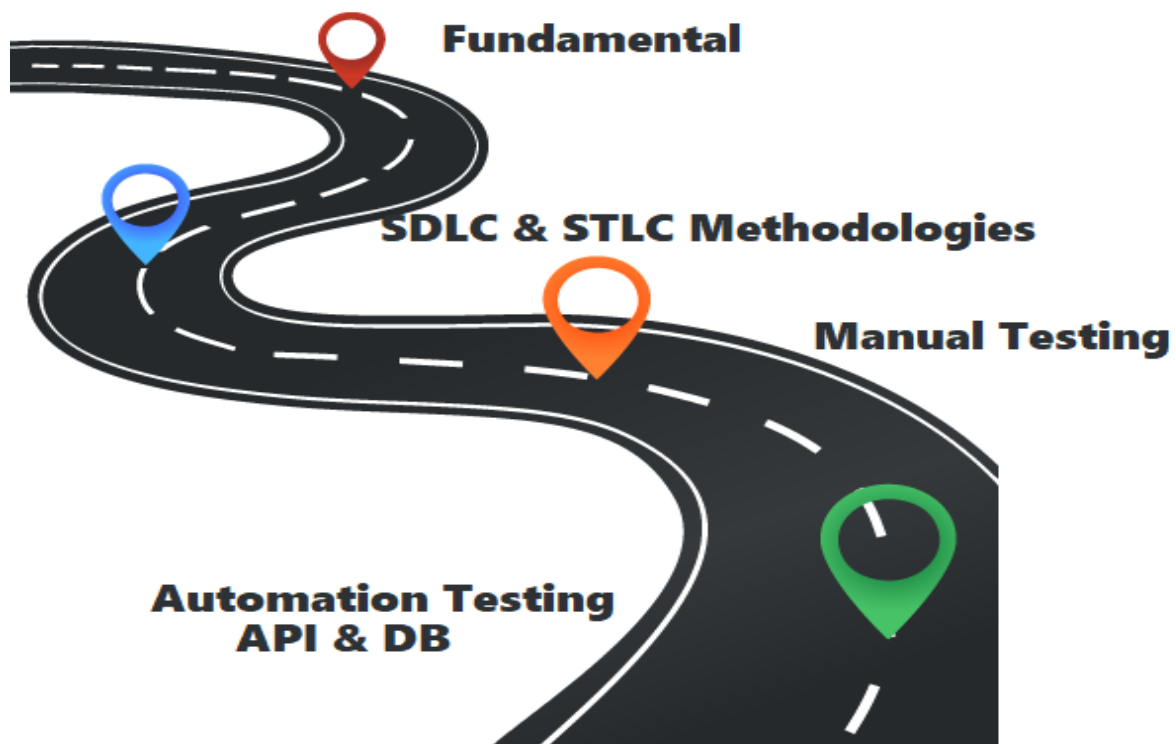
Road-map to become a SQA (SQA Base)



Why do you need strong base to be an SQA?

My interest grew on SQA because of Selenium Web driver. One of my friends showed me how his codes were checking through all the features and I found it interesting. Very next day I googled and searched "How to check features automatically?" and I started watching tutorial of few Indians. I had a brief knowledge about software testing. So, I kept watching and trying to learn selenium but somewhere I was lacking. I didn't think about giving up instead I was looking for ways to improve. Finally, I realised my base is not as strong as it need to be. So, my quest leads me to basic to start again. Testing methodologies and Manual testing are core of all software testing activities, rich base will grant smooth journey on QA world.

In my opinion If you are determined and eagerly trying to be a SQA but finding no path then you can follow the road-map that I am going to discuss, hope it will be helpful:



Fundamental:

Start with very fundamental topic like what is Software Quality Assurance & Testing, role and responsibilities of a SQA Engineer, importance of SQA engineering. It will straighten your journey and give you clear idea of why and what you are going to do ahead.

Testing strategies: Main target of SQA/Testing is to get rid of error and improving ultimate quality of final product/software. Test strategies are classified as

- **Black-box Testing** - Testing based on system behaviour, inputs and outputs.
- **White-box Testing** - Testing based on system structure and processing.

Experienced-based Testing - Often combined with black-box and white-box testing and depends on developers, testers, users experience.

Test Types:

Typically, software testing is classified into three categories:

1.Functional Testing - Testing features and functionalities of system.

Types of Functional Testing:

- Unit Testing
- Smoke Testing
- Integration Testing
- Sanity Testing
- User Acceptance Testing and so on

2. Non-Functional Testing - Testing non-functional aspects like performance, reliability, usability, security etc. Of a system. Types of Non-Functional Testing:

- Performance Testing
- Load Testing
- Stress Testing
- Security Testing
- Volume Testing
- Usability Testing and so on

3. Maintenance Testing - Testing final build and maintainability of system.

Types of Maintenance Testing:

- Maintenance Testing
- Regression Testing
- Test Levels:

Four levels of software testing:

- Unit Testing
- Integration Testing
- System Testing
- Acceptance Testing

SDLC & STLC Methodologies:

SDLC (Software development life cycle) and **STLC (Software testing life cycle)** must follow to ensure high-quality software system which helps you to meet the customer expectations.

SDLC - is a way to develop system/software through a phased manner in the following order:

- Requirements Analysis
- Design the software Architecture
- Build the Software
- Test
- Deployment & Maintenance

STLC - is the testing process of a system/software that is executed in a well-planned manner. STLC stages are:

- Requirement Analysis
- Test Plan
- Test Case Develop
- Environment setup
- Test Case Execution
- Test Cycle Closure
- Development and Testing models-

Agile Model - continues and iterative method

V Model - sequential method

Waterfall Model - hands off approach

Bug & Bug life Cycle:

Error (mistake)/defect (tester found)/bug (developer assigned) is the cause of a system/software failure. There is a life cycle in STLC to keep tracking and fixing bug this is known as Bug life cycle. Bug life cycle has following stages:

- New
- Assigned
- Open
- Fixed
- Re-test
- Verified
- Re-open
- Closed

- Duplicate
- Rejected
- Deferred
- Not a bug

Management Tools:

Test management tools are used to store information on how testing is to be done, plan testing activities and report the status of quality assurance activities. The tools have different approaches to testing and thus have different sets of features. When you have covered the above topics with good understanding, it is good to start using different management tools like:

- qTest
- TestLink
- TestRail
- Jira
- Trello etc.

which will be great help to get better jobs also increase your mastery in the field of SQA.

Manual Testing:

After completing all the previous tasks and learning, start applying your knowledge on demo or real projects which will add extra points in your bag. So, start testing a software manually, in this phase you will learn about

- Test Driven Development
- Test Planning
- Writing effective Test Scenario and Test Cases
- Bug Tracing and Reporting to stack holder
- Proper knowledge about system Compatibility

Completing a software testing manually using different project management tools is very beneficial and increase experience. And keep practicing to be an expert.

Automation Testing:

Now you are already well known about testing, to automate your manual test cases you need to know about programming language like python or java. Also, you have to remember that automation testing codes cannot require a testing. So, automate tester need very good knowledge on coding and the application under test.

We can divide automation testing as

1. API Automation Testing
2. Mobile Automation Testing
3. Web Automation Testing.

When learning to become a smart automation testing engineer, automation tools are mandatory. There are lot of automation tools like:

- Ranorex Studio
- Selenium
- 21
- Testim
- QTP
- Appium
- Kobiton
- PostMan
- SOAP UI
- RunScope

Experimenting with those tools will help you in gaining much knowledge about their architecture, environment, feasibility for your applications under tests, and much more.

And Few topics should know to be a Good SQA:

- Git - Version Controller
- CI & CD basics (Jenkins, GoCD)
- Repository Hosting (Github)

- Terminal and Command Line
- UI Testing
- Profound knowledge on Database for DB Testing.

This is the way to start learning about SQA but this is not the end, you can call another beginning. So, keep learning and stay updated with SQA trends. The trending best practices, tools, techniques, tips, and tricks will help you to achieve success in optimizing your test strategies and methodology. To become a smart SQA engineer and to be successful in this industry, you need to really work hard with passion and dedication.