



zappletech

100% focused on
Test Automation

TEST AUTOMATION TRENDS IN 2022

Test automation trends in 2022

The automated testing market is developing by leaps and bounds. [Markets And Markets](#) predicts the market will reach \$49.9 billion by 2026. For comparison: in 2021 it was \$20.7 billion. Based on our experience, we can say that such growth is associated with the active introduction of advanced technologies, new tools, and methods. Zappletech team is constantly monitoring the situation in the QA, and therefore we have something to tell you about. Based on the current situation in the market, we have highlighted some striking trends and briefly described the essence of each of them.



What is Going On in the Test Automation World?

1. DevOps, TestOps, and QAOps

DevOps brings development and operations together. But to simplify and speed up work processes, developers must interact with testers. As a response to this need, a new direction arose – TestOps (QAOps). We call it the next stage in the DevOps culture.

Criterion	DevOps	TestOps (QAOps)
Essence	Integration between the workflows of programmers and IT specialists	Close interaction between QA, Dev, and Ops within the SDLC.
Objectives	Ensure the predictability, efficiency, and safety of development.	Maximize product delivery speed without sacrificing quality.
Stages	Planning, coding, assembly, testing, release, deployment, operation, monitoring.	Planning, control/test ownership, management, insights.



2. ShiftLeft and ShiftRight Approach

These approaches appear regularly on trending lists. The first involves testing at an early stage of the SDLC, emphasizing functional aspects. The second aims to conduct testing at the end of the SDLC. The approaches complement each other perfectly.

Criteria	ShiftLeft	ShiftRight
Essence of approach	Early testing in parallel to development with communication between developers, QA, and the client.	Testing after deployment, when real users interact with the product.
Process	<ol style="list-style-type: none">1. Product analysis, study of customer requirements, and end-user preferences.2. Development of tests (integration, modular, functional).3. Executing tests through end-to-end automation.4. Running other tests.	<ol style="list-style-type: none">1. Tests in production environments.2. Collecting reviews from real users.3. Load and use case testing that is not possible in a test environment.
Pros	<ul style="list-style-type: none">• Save time on finding and fixing bugs• Early detection of design bottlenecks• Optimization of development• Fast delivery of a quality product.	<ul style="list-style-type: none">• More possibilities for automation• Improved user experience• Expanded test coverage• Timely detection of production problems.

3. Quality is owned by the entire team, but only QA and AQA

Many specialists contribute to the quality of a product: from an analyst to an entire development team. Each performs a set of tasks to achieve a common goal – the rapid release of a product that works consistently and meets users' requirements.





Besides QA, the following are responsible for product quality:



Development team:

performs tasks in identifying, establishing, designing, and testing products.



Product manager:

researches the market, finds the client's weak points and forms a vision so that the final product complies with the market.



Project manager:

sets tasks and distributes the workload so that the product is released on time without sacrificing quality.



Program Manager:

defines the strategies and goals of the program, ensures consistency between teams.



UX Designer:

Provides quality product interaction with the user.



Data Scientist:

Gathers and analyzes product data for informed decision making.

4. API testing will continue to grow

API tests are aimed at testing a newly developed product from different points of view: performance, security, reliability, functionality, and the correct interaction with other programs.

Why API testing is popular:

- Early diagnosis and troubleshooting.
- High testing speed: 35 times faster than GUI.
- Reduced overall testing costs.
- No attachment to programming languages.
- Wide test coverage.



5. API and Services Test Automation or contract testing

Contract testing can be called the link between unit and integration testing. This method is actively used by modern QA teams in cases where it is necessary to check each system independently from the other.

Advantages of contract testing:

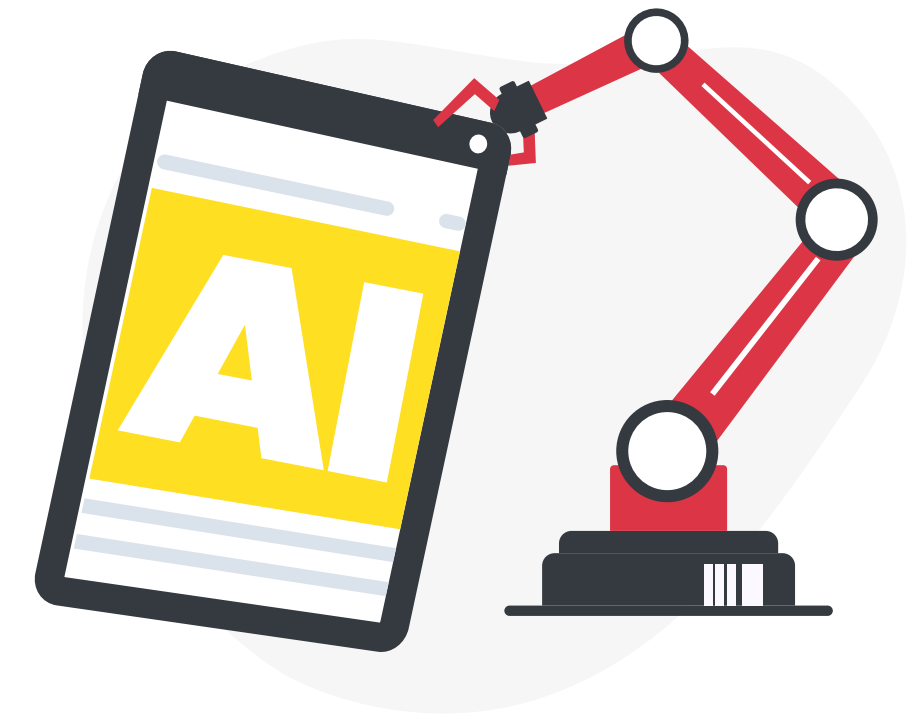
- High speed due to the lack of interaction with several systems.
- Easy to maintain because you don't need to understand the whole ecosystem.
- Fast debugging and fixing because the problem lies in the component that is being tested.
- Run tests and detect defects locally on the developer's machines.

6. Artificial Administrative unit for Testing or AIAssisted Test Automation

QA specialists have appreciated AI benefits. The technology can be integrated into work tools to make it faster and easier to create test cases and perform visual testing. Also, AI can play the role of an assistant in the analysis and reporting of testing.

**AI capabilities for testing:**

- Quickly find visual errors in the interface thanks to pattern and image recognition.
- Minimizing the influence of the human factor in repetitive tasks.
- Better test coverage and maintenance of continuous testing, allowing you to bring products to market faster.
- Early error detection, which increases the reliability of the product.

**7. NLP-based automation tools are under consideration**

Modern companies use such tools to automate scriptless testing. Thus, complex automation is performed in simple natural language and is accessible to users without QA skills.

Benefits of NLP tools:

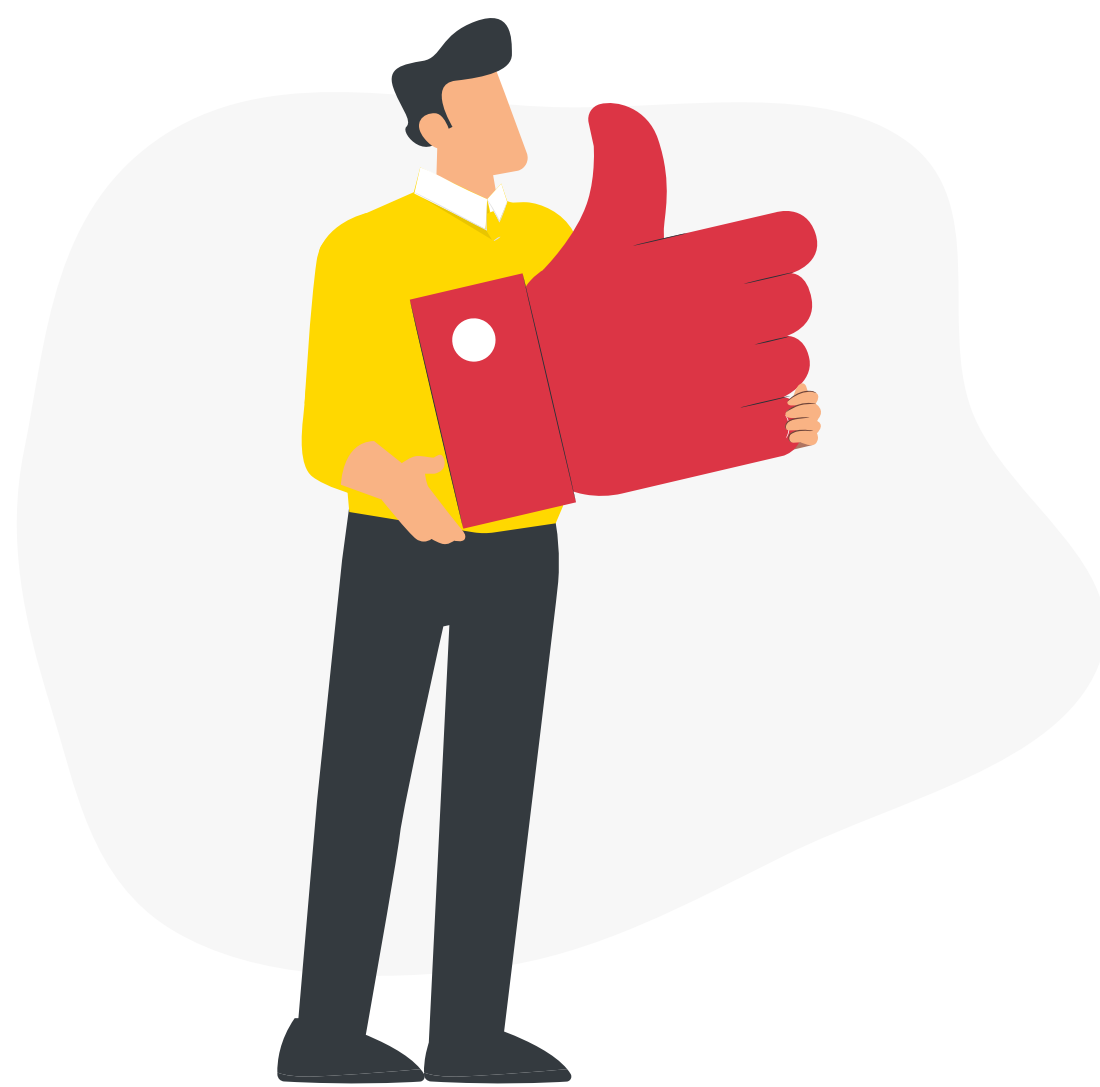
- Short learning curve due to no need to learn languages.
- Clear test cases for users of all levels and stakeholders.
- Ability to involve any team member, including from the client-side, in the creation of a test case.

8. Mobile Test Automation

Based on our experience, we note that mobile tests are more complicated than web tests: in addition to the platform and mobile screen resolution, custom UIs must be considered. Yes, some actions are problematic to automate, for example, scrolling. But in most cases, automation is essential.

When it is important to automate mobile tests:

- A long-term project that needs to be constantly maintained and developed.
- Maintenance checks to monitor sync daily and prevent crashes.
- Smoke testing when it comes to a product that does not use external resources.
- Regression testing, including routine, repetitive tasks.
- Continuous integration that always focuses on autotests.

**9. Codeless automation or Scriptless Test Automation**

These terms essentially mean the same thing. Tools and frameworks that work without code greatly simplify the life of specialists without programming skills: from manual QA to business users and citizen testers.

How codeless automation makes life easier for non-techies:

- No need to learn complex coding techniques and frameworks.
- Intuitive use of drag and drop tools.
- Independence from developers in the testing, which minimizes costs.
- Easy maintenance and scalability with visual UI workflows with up-to-date business standards.

10. Integration of Tools and Activities

In testing, it is common practice to implement additional tools to improve work efficiency and solve specific problems.

It is useful to use tools for:

- Test management: planning, defect logging, tracking, analysis.
- Configuration management: implementation, execution, change tracking.
- Test data preparation: analysis, design, data generation.
- Test runs: implementation and execution.
- Project planning and tracking: managerial decision making.
- Incident management: test management.

11. JavaScriptbased testing tools will proliferate

The JavaScript language is a leader in the programming world. It's no surprise that automation professionals around the globe are choosing JavaScript-based frameworks. And we are confident that this trend will continue in the coming years.

Popular JavaScript-based testing tools:

- WebdriverIO: next-gen browser and mobile automation testing.
- Cypress: fast and reliable End-to-End testing.
- TestCafe: cross-browser testing out-of-the-box.
- Playwright: quality end-to-end testing for modern web apps.



12. Fragmentation in test automation tools will continue

Many companies tend to invest in a range of tools that collectively solve all problems and cover all automation needs. The difficulty is to choose the right tools that are compatible with each other.

Questions when choosing instruments in a fragmented market:

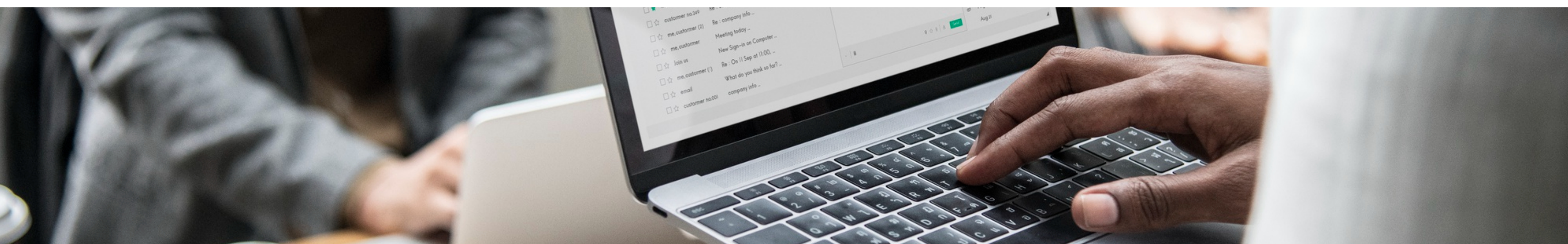
- What types of tests will you work with?
- What technologies will be used?
- Is there a need for low-code or no-code tests?
- What is more profitable: open-sourced or licensed?
- Which features are desirable and which are required?

13. Cloud-based crossbrowser testing or CloudNative Testing

Cloud-based crossbrowser testing is actively used in relation to cloud, web, and installed apps. Unlike traditional testing, this approach provides end-to-end testing across all devices across platforms, operating systems, and browsers.

Questions when choosing instruments in a fragmented market:

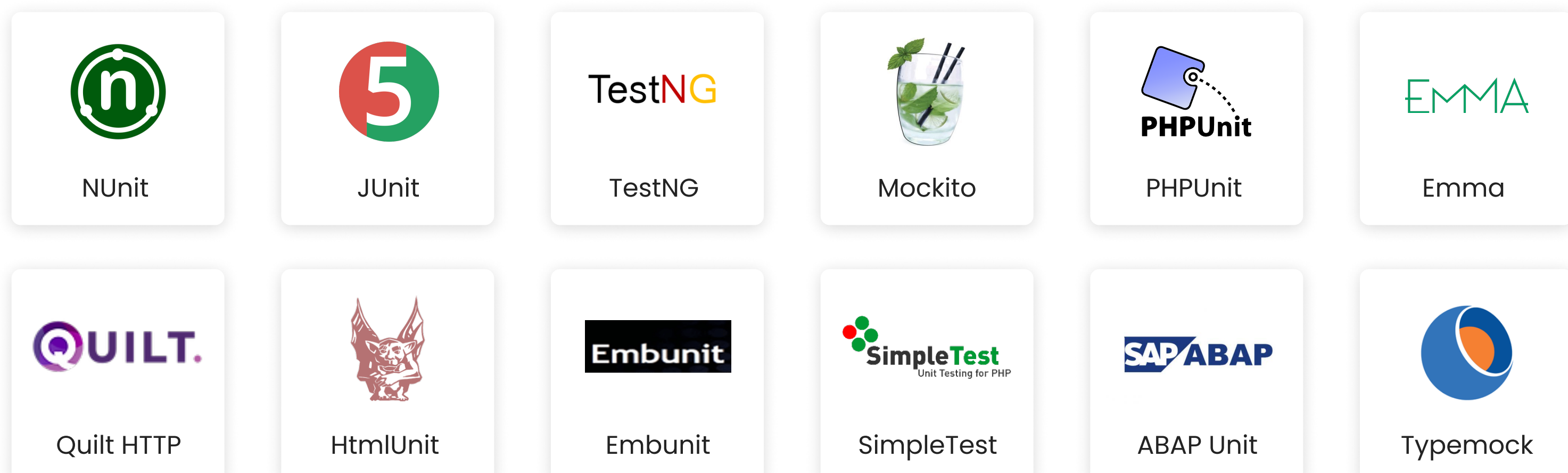
- Cloud-based crossbrowser testing: real-time tests focusing on desktop and mobile browsers, such as LambdaTest.
- CloudNative Testing: test apps hosted in a dynamic environment in a private, public, or hybrid cloud.



14. External Dependency Test Tooling on the Rise

Sometimes, there is a need to test an app along with external dependencies. They come in two types: those you control (e.g., database, file system) or those you don't control (an API developed by another team). This is where unit testing tools come in handy.

Popular unit test tools:





15. Test Environments and Data management

As Agile and DevOps practices have grown in popularity, the complexity of working with data in various test environments has increased. This becomes an incentive to rethink their TDM and TEM strategies.

What to focus on in TDM and TEM:

- **TDM:** reduce data delivery time, meet data accuracy requirements, reduce security risks without sacrificing speed, minimize storage and archiving costs.
- **TEM:** test environment awareness for better decision making, test environment planning, and coordination, optimization through environment automation and insight.



16. The Top Testing Programming Languages for QA

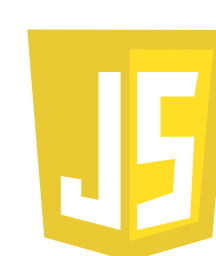
Based on our experience, we can distinguish the top 3 languages in terms of convenience, efficiency and reliability.

Which programming languages to choose for automated testing:

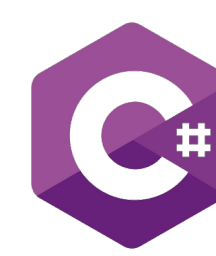
- **Java:** the most common language for automation with many frameworks and plugins, a large community, and many educational resources.
- **JavaScript:** its popularity is due to the spread of left-shift methodologies and the desire of QA to speak the same language as developers.
- **C#:** the language has earned respect among testers due to its efficiency in processing test scripts and compatibility with Selenium WebDriver.



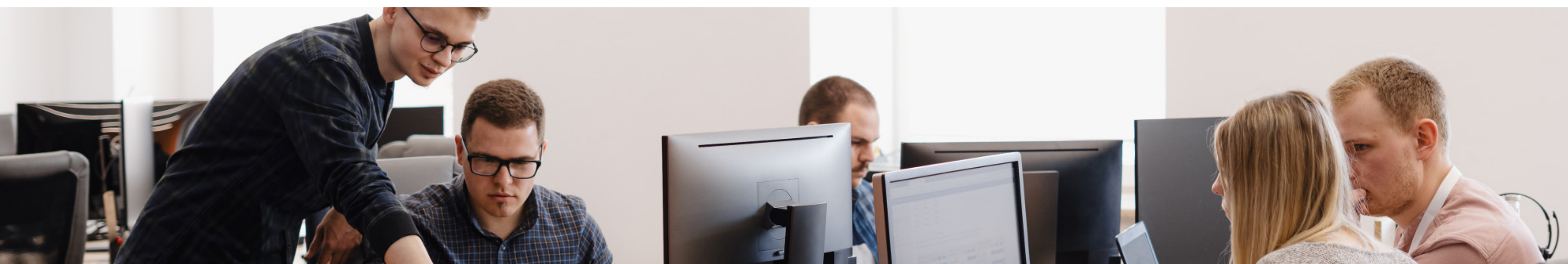
Java



JavaScript



C#



17. Testing Centers of Excellence to give skills more quickly

TCOE is the basis for business process testing, where skills are more important than speed. Many large enterprises use this model to accelerate innovation and improve the quality of processes and systems.

4 steps to create a TCOE:

- **Assess the situation:** people and their skills, processes, technologies, unique challenges, and initiatives.
- **Test a new approach to demonstrate automation's value:** define business processes, train the team on the automation platform, develop functional tests, test processes again and again, and create documentation.
- **Build a TCOE program:** define leadership structure, choose the best automation platform, formulate goals, document experience.
- **Drive enterprise implementation:** define sponsorship, prioritize, create a roadmap, track results, report success.

19. Security testing trendy due to raised risks

As digitization becomes more widespread, security risks increase. Every company wants to know if there are weaknesses in its product that attackers can use for their own purposes and therefore turns to QA.

What tasks does security testing solve:

- Identifying system vulnerabilities and loopholes for hackers.
- Checking the effectiveness of the current protection strategy.
- Ensuring business processes continuity.
- Maintain the trust of customers and partners through strict adherence to data security standards.

As you can see, the world of QA is very diverse and comprehensive. To be successful in any initiative, you need to consider many nuances, have deep knowledge in different areas, and be able to balance speed and quality. It once again emphasizes the importance of cooperation with professionals.

18. Expect more emphasis on skills than tools

Once again, the emphasis on skills often plays a greater role than tools. Since the effectiveness of working with tools depends on the level of skills.

Three reasons to focus on skills:

- Deep immersion in the project and understanding of the specifics of the tasks.
- Professional knowledge of tools and technologies.
- Personal responsibility for the quality of tasks.

