I am a backend developer with a solid foundation in C# and ASP.NET Core, which has helped me develop strong problem-solving skills. Although my primary background is in backend development, I have a keen interest in quality assurance and am eager to transition into this field. I am particularly excited about working with mobile and web applications and gaining hands-on experience in testing and ensuring the quality of software products. I am currently a student at Khazar University, pursuing a degree in Computer Engineering, which has given me the technical knowledge and analytical skills to succeed in QA testing. I am enthusiastic about learning more about manual testing, creating test plans, and contributing to high-quality product development.

What is Functional Testing?
Functional testing is a type of testing where we check if the software works as expected. It tests different parts of the software, like the user interface, APIs, databases, and security. The goal is to make sure everything in the software works according to the requirements.

What Kind of Testing Covers Functional Testing?
Functional testing includes different types of testing, such as:

Unit Testing: Testing small parts of the code.

Integration Testing: Testing how different parts of the software work together.

System Testing: Testing the whole software system together.

Acceptance Testing: Checking if the software meets user needs and is ready to be released.

What Do You Test in Functional Testing? In
functional testing, you test:

User interfaces to make sure they work properly.

APIs to check if they give the right responses.

Databases to make sure the data is correct and saved properly.

Security features to ensure the software has the right access controls.

Overall workflow to check if all the functions meet the requirements.

SDLC (Software Development Life Cycle) refers to the stages followed in the development of software. These stages include planning, designing, developing, testing, deploying, and maintaining the software. SDLC ensures that the software is delivered on time, meets quality standards, and satisfies user requirements.

A **Bug** is a mistake or issue in the software where it doesn't function as expected. It refers to problems or defects that users might encounter, causing the software to behave incorrectly or unexpectedly. Not all bugs are critical; some might not affect the overall system much.

An **Error** is a mistake or issue that occurs during the execution of the software. It can happen due to incorrect logic, incorrect input, or faults in the code that prevent the system from working as expected. Errors can be caused by users or internal programming mistakes.

White Box Testing is a type of software testing where the tester has access to the internal structure, design, and implementation of the application. The goal is to test the internal logic, code flow, and structure of the software. This includes testing individual functions, paths, and branches in the code. The tester needs to have knowledge of the programming language and the source code.

Black Box Testing is a type of software testing where the tester does not have knowledge of the internal workings or code of the application. The tester focuses only on testing the functionality of the software, verifying that it behaves as expected based on the requirements. The goal is to identify any issues related to the software's external behavior, such as incorrect outputs, unexpected results, or failure to meet user expectations, without considering the internal code structure.

What is the life cycle of QA testing?

Here is a standard version of the testing process:

Check requirements Plan Analyze Design Implement E xecute Conclude Close