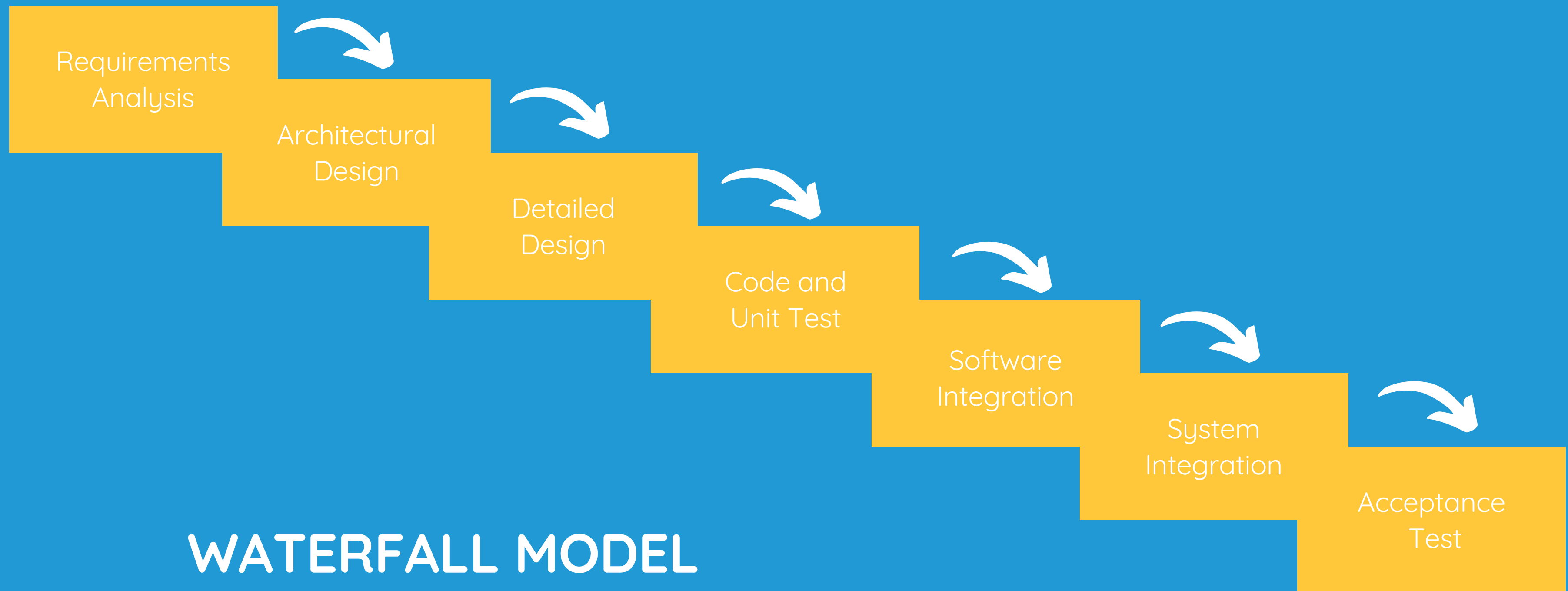


CAROLINA AZEVEDO | NELSON COSTA

ACCEPTANCE TESTING

Software Testing,
Verification and Validation



BUT WAIT ...

WHAT'S ACCEPTANCE TESTING?



Formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system.

Definition by **International Software Testing Qualifications Board**

BUT WAIT ...

WHAT'S ACCEPTANCE TESTING?



Formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system.

Definition by **International Software Testing Qualifications Board**

Purpose



Other Testing :

Intent is principally to reveal errors

Acceptance Testing :

- performed to determine whether or not the software system has met the requirement specifications
- evaluate the system's compliance with the business requirements and verify if it's has met the required criteria for delivery to end users

Types of Acceptance Testing

Alpha

Development Environment. Usually done by internal staff. Based on feedback - collect from the alpha testers - teams then fix certain issues and improve the usability of the product. &

Beta

Customer's environment. Testing is made by a group of customers who use the system in their environment. Beta testers then provide feedback, which in turn leads to improvements of the product.

Contract

Means that a developed software is tested against certain criteria and specifications which are predefined and agreed upon in a contract. The project team defines the relevant criteria and specifications for acceptance at the same time when the team agrees on the contract itself.

Regulation

Also known as Compliance Acceptance Testing, examines whether the software complies with the regulations. This includes governmental and legal regulations.

Operational

Also known as Operational Readiness Testing or Production Acceptance Testing, these test cases ensure there are workflows in place to allow the software or system to be used. This should include workflows for backup plans, user training, and various maintenance processes and security checks.

Black Box

It's a method of software testing which analyzes certain functionalities without letting testers see the internal code structure. During the test, the user isn't aware of any code base, but only about the requirements which the software should meet. The tester conducting Black Box Tests is only aware of what the software is supposed to do. They don't know how it should be done.

FRAMEWORKS AND TOOLS



Cucumber




- **Open Source** software tool used by computer programmers that runs automated acceptance tests written in a **behavior-driven development (BDD)** style.
- Allows expected software behaviors to be specified in a logical language that customers can understand.
- **Gherkin** is the language that Cucumber uses to define test cases. It is designed to be non-technical and human readable, and collectively describes use cases relating to a software system.
- Originally **written in the Ruby** and originally used exclusively for Ruby testing, Cucumber now supports a variety of different languages including **Java** and **JavaScript**.

Concordion



- Is an **open source tool** for writing automated acceptance software tests in Java development environment.
- Directly **integrates with JUnit**, which allows for easy use with IDE of your choice (Netbeans, Eclipse, IntelliJ IDEA).
- Uses **acceptance tests specifications in native language**, which allows use of Concordion for requirements management.

Robot Framework



- Robot Framework is an open source automation framework written in Python for acceptance testing, **acceptance test driven development (ATDD)**, and **robotic process automation (RPA)**.
- Test cases are written using a keyword-testing methodology written in a **tabular format**.
- Tables can be written in plain text, HTML, tab-separated values (TSV), or reStructuredText (reST) formats files in any text editor or using the Robot Integrated Development Environment (RIDE).
- **RIDE simplifies writing test cases** by providing framework-specific code completion, syntax highlighting, etc.



- Is an **open source reporting library** , in a behavior-driven development (**BDD**) style.
- Produces **rich meaningful test reports** (or "living documentation") that not only report on the test results, but also what features have been tested.
- Helps writing better **structured**, more **maintainable automated** acceptance criteria.



- Is a **web server**, a **wiki** and an **automated testing tool** for software.
- Allows users of a developed system to enter specially formatted input (its format is **accessible to non-programmers**).
- The program first supported only **Java**, but versions for several other languages have been added over time (**C++**, **Python**, **Ruby**, **Delphi**, **C#**, etc.).
- Is based around the notation of **black-box testing**, in which a system under test is considered to be a black box and is tested in terms of the outputs generated in response to predetermined inputs.



And FitNesse is the chosen one !

Testing within the FitNesse system involves **four components per test**:

- The **wiki** page which expresses the test as a decision table.
- A testing **engine**, which interprets the wiki page.
- A test **fixture**, which is invoked by the testing engine and in turn invokes the system under test.
- The **system under test**, which is being tested.

Of these components the software development team produces two: the wiki page and the fixture (of course it also produces the system under test, but from the point of view of the black-box test only two).

CAROLINA AZEVEDO | NELSON COSTA

ANY QUESTIONS?

TVVS 2019 -
ACCEPTANCE TESTING