# Assignment 13 - Time for the Test

| 👥 Owner | 🔖 Pankaj Kumar |
| --- | --- |

## 1: What are different types for testing?

- `Unit Testing:` focuses on individual units or components of the software, ensuring they work as intended.

- `Integration Testing:` combines different units and tests their interaction, ensuring they work together as a system.

- `Functional Testing:` tests the functionality of the software, verifying it meets the requirements and specifications.

- `End-to-end Testing:` tests the entire system, from start to finish, simulating real-world scenarios.

- `System Testing:` tests the system as a whole, verifying it meets the required performance, security, and reliability standards.

- `Acceptance Testing:` tests the software from the user's perspective, ensuring it meets the customer's expectations.

- `Performance Testing:` tests the performance of the software, such as response time, scalability, and stability under different load conditions.

- `Security Testing:` tests the security of the software, verifying it is protected against potential threats and vulnerabilities.

- `Regression Testing:` tests the software after changes have been made, ensuring the changes did not introduce new bugs or break existing functionality.

- `Smoke Testing:` a preliminary test to determine if the basic functions of the software work, before proceeding with more thorough testing.

## 2: What is Enzyme?

Enzyme is a JavaScript testing utility for React, developed and maintained by Airbnb. It can be used in both unit and integration testing.

## 3: Enzyme vs React Testing Library

| Features | Enzyme | React Testing Library | | :---- | :------ | : ------------- | | API | Enzyme has a more comprehensive API with methods for `manipulating, traversing, and querying` the React component tree, which can be convenient for `unit testing`. | React Testing Library, on the other hand, has a simpler API that focuses on testing the `behavior` of the components from the user's perspective, making it more suitable for `integration` and `end-to-end` testing. | | Approach | more `implementation-focused` approach to testing, where you test the internal implementation details of the components, such as the `state or props` | React Testing Library, on the other hand, has a more `user-focused` approach, where you test the `behavior` of the components as a user would interact with them, such as `clicking buttons` or `filling out forms`. | | Maintenance | Enzyme requires `more maintenance` as the `internal` implementation of components changes, as the tests are `tightly coupled` to the implementation details. | React Testing Library, on the other hand, is `less` likely to break with changes to the implementation, as it tests the `behavior` of the components rather than the implementation details. |

## 4: What is Jest and why do we use it?

`Jest` is a `JavaScript testing framework` developed and maintained by `Facebook`. It is widely used for testing JavaScript applications, especially for `React` applications. Jest provides a complete and integrated testing solution, with features such as `automatic test discovery, mocking, code coverage, and assertion libraries.`

Jest is a popular and widely used testing framework for JavaScript applications due to the following reasons :

1. **Simplicity:** minimal configuration & low learning curve

2. **Speed:** fast test execution, automatic test caching, parallel test running

3. **Integration:** integrates well with popular JavaScript tools and frameworks, such as React, Babel, and Webpack.

4. **Feature:** mocking, spying, and code coverage reporting

Jest makes it easy for developers to write and run tests, ensuring the quality and reliability of their code.