Testing

- is a process of making sure a software works as expected.
- Manual testing is the old way where we test it pretending to be a user. which have its own drawbacks.

Automated testing

- are programs that automate the task of testing our software.
- we basically write a code to test our software code.
- it is not time consuming, and we can easily identify and fix features that break tests.
- the more our tests resembles the way our software is used, the more confidence they can give us.

Unit test

- Test individual functions, methods, or components in isolation.
- is a small and targeted test.

Integration Test

- verify that different part of the app work together.
- tests the interaction between modules, components and services | like component and API
- often includes APIs, databases or external services..

End to End Test

- tests a complete workflow from a user side, simulating a real interaction with the app
- covers everything from frontend to database.

Performance Test

- tests how an app perform under specific conditions like Load, Stress
- make sure the app can handle high traffic, large dataset....

Testing Pyramid

- when testing follow the Testing Pyramid concept, where Unit Tests comes first, then Integration and End to End Test.

Tools | Libraries

Vitest and React Testing Library

- to setup Vitest with our Next.js Application: "npm install -D vitest @vitest/ui @testing-library/react @testing-library/jest-dom jsdom" install packages as a dev dependencies.
- create a config file vitest "vitest.config.mts"
- add "test" script to our package.json "vitest"
- create a test and run it: "npm run test"
- we can either store all the test files inside a folder called "__tests__" or just inside the app folder.

Testing Components

- test how they Render and how they respond to user interactions.
- to write a good Tests, we should focus on testing how our component behave(end result) not how it is implemented.