UNIVERSITATEA DIN BUCUREȘTI

VIRTUTE ET SAPIENTIA

WEB TECHNOLOGIES USING JAVA

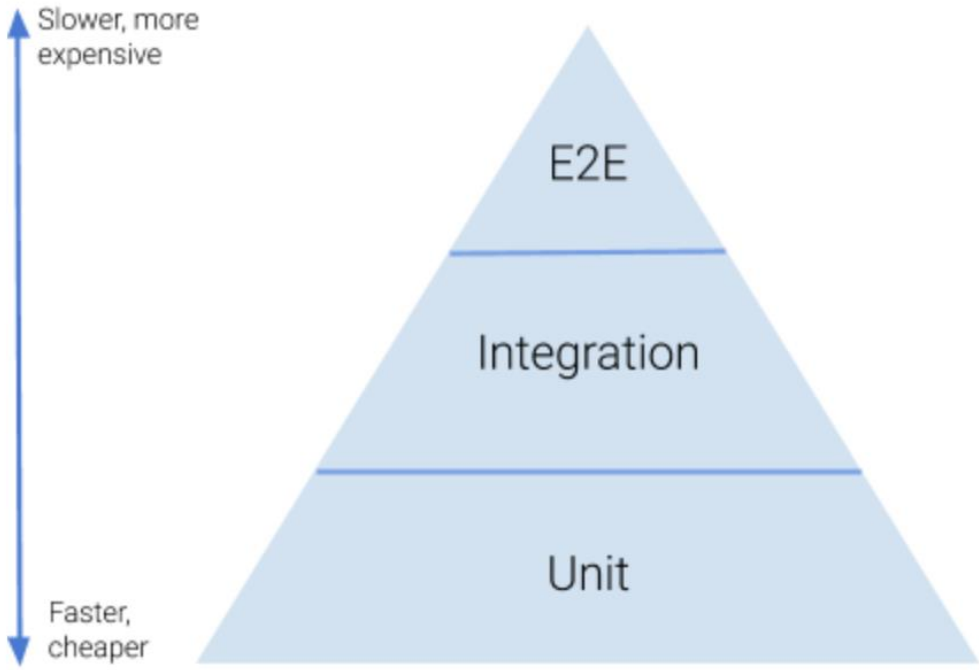
COURSE 12 – INTEGRATION TESTING WITH SPRING

AGENDA

- INTEGRATION TESTING
- SPRING BOOT TESTS
- TESTING SLICES WITH SPRING
- BEST PRACTICES FOR TESTING

INTEGRATION TESTING

- tests that cover multiple layers of an application in combination.
- integration tests launch the entire Spring context, including business and data components
- automated integration tests usually don’t rely on external dependencies (for example, in-memory database are used or stubs for external services)



endava

SPRING BOOT TESTS

- Spring Boot provides a **@SpringBootTest** annotation that can be used to quickly launch an integration test
- integration tests may start from the controller layer or the service layer
- to configure a web environment to be used in the test:
  - MOCK: To unit test controllers in a mock web environment.
  - RANDOM_PORT: Launch the entire web context, including other layers, in an embedded server on an available random port.
  - DEFINED_PORT: Similar to RANDOM_PORT, except that you hardcode a port number. May lead to port conflicts.
  - NONE: Loads up a Spring context without a web context.

TESTING SLICES WITH SPRING

- slice = a particular layer / domain of an application (controller, repository etc)
- testing slices of application allows for focused, fast, light testing and facilitate targeting specific functionality
- Spring Boot test slice annotations load an ApplicationContext and select components that make sense for the specified slice
- **@WebMvcTest**
  - used with tests that focus just on the Spring MVC components
  - will disable all auto configuration that is unrelated to web MVCs (for example, @Service, @Repository)
  - MockMvc bean is used to make requests
  - @MockitoBean is used to mock service beans or other beans, if needed (@MockBean was used with Spring Boot < 3.4.0)

BEST PRACTICES FOR TESTING

- FIRST properties of a good test:
  - [F]ast
  - [I]solated
  - [R]epeatable: produces the same results each time you run it. You must isolate them from anything in the external environment not under your direct control.
  - [S]elf-validating
  - [T]imely

BIBLIOGRAPHY

- Spring in Action, by Craig Walls
- Java Unit Testing with Junit 5, by Shekhar Gulati, RahulSharma
- Pragmatic Unit testing in Java 8 with Junit, by Jeff Langr

Q&A

THANK YOU

DANIELA SPILCĂ