

ITP2200

Introduction to Software Testing

Bogdan Marculescu

Lecture 9

Testing for Different Technologies

So, more programming this week?



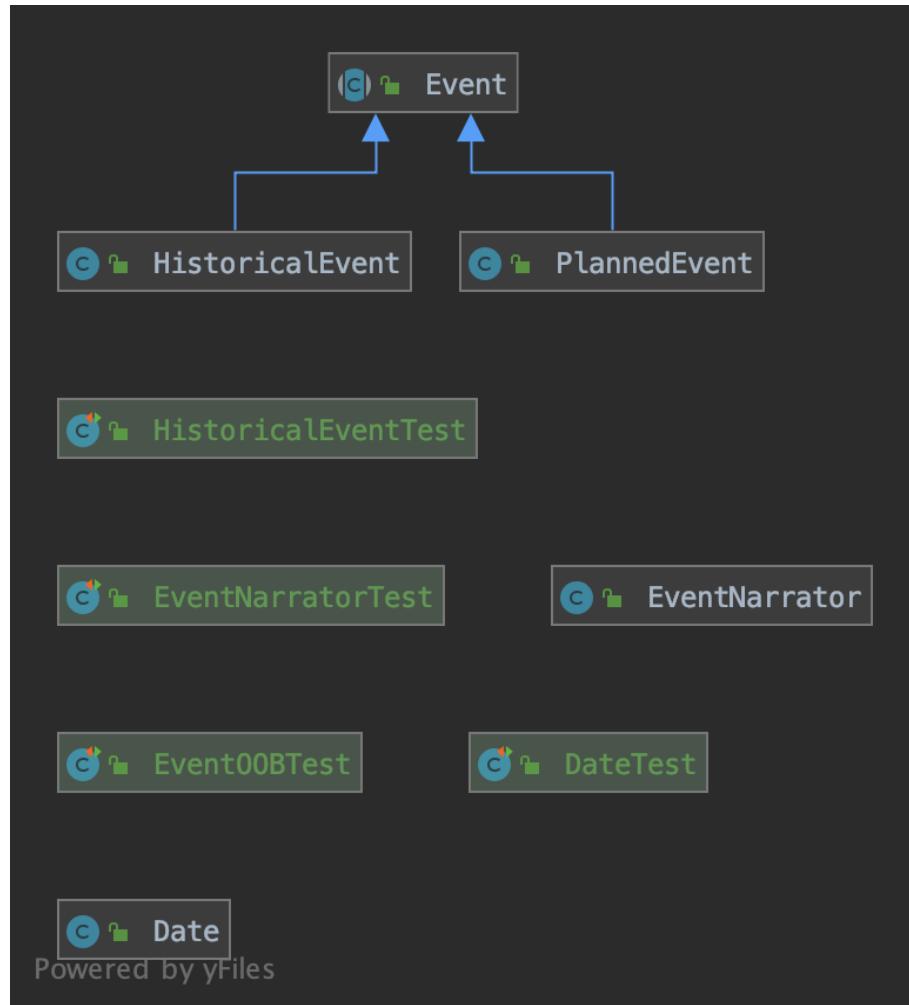
Let talk about a practice exam!



Document exists with exam tasks.

- The actual exam: same tasks, different system
- Today is about explaining what my expectations are for those tasks.
- There will be programming in the exam!

Let talk about a practice exam!



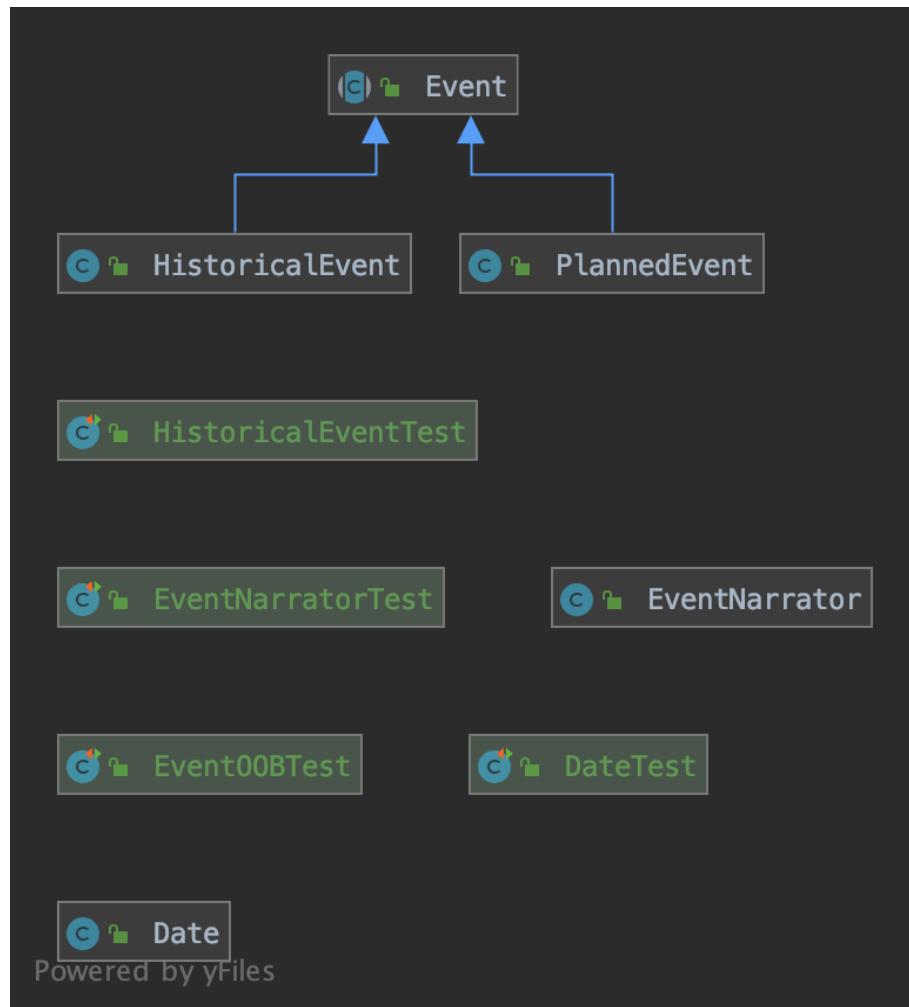
Topic:

- A system (System under Test or SUT)
- Some requirements it should implement
- An overall design

Example (note, the exam system will differ)

Write an event narrator that...

System under Test



A system (System under Test or SUT) - some assembly required

- Won't be too complex.
- So, you will create a new project, containing sources, tests, readme.md, and a pdf of the reflections.

Let talk about a practice exam!

Teams! (of max 6)

- A google doc should be up.
- Discuss, form groups. If you don't know anyone, add yourself to the "No Group" track.
- Partial groups are okay (i.e. less than 6, but you may get an extra colleague or two).
- I will randomly assign people from the "No Group" set to complete teams.



Wait? Teams?

Yes! (Online, obviously)

Some recommended tools for collaborations:

- Version Control System: git (github or bitbucket)
- Planning: Trello
- Chat: Discord, Slack (or any other number of them)
- IDE: IntelliJ (many other alternatives exist, of course)
- Voice chat: skype, whatsapp, etc.
- Documents: google docs, dropbox, etc.



First Steps



- Discuss the team, skills, what you want to learn
- Discuss procedures for work (who is responsible for each component, how do we set deadlines, how often we report in, stand-up meetings, what do we do if we have questions/problems)
- Read the assignment thoroughly and see if you all agree on an interpretation

Key things

- Focus on the team and the process
- **Analyse and discuss** before starting to work
- Think about testing criteria **before** you implement
- Write tests **before** or **while** you implement
- Reflect on the procedures, criteria, tests, requirements, etc.
- Change anything you think needs changing, then **record** that change and **motivate** it
- Reflect on the quality of the process and the product





Høyskolen
Kristiania



Questions so far?
