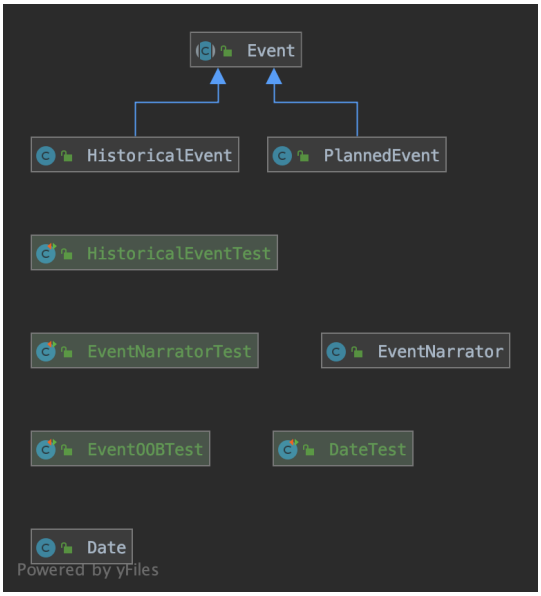
Topic:

- You will get a system description, some requirements, and an overall system design.
- **NOTE: The Exam SUT will be different, but the Tasks will be similar**

Write an event Narrator



The purpose of this system is to provide a comment (as a String) of events that happened in the past or are planned for the future, based on their date.

Requirement 1: Based on the date of a past event, have a method that returns a String, containing the amount of time that has passed since that event.

a. If the event is less than 30 days ago, the method will return the string containing the number of days (For example “The last lecture was 7 days ago”).
b. If the event is more than 30 days, but less than a year ago, the method will return the string containing the number of months (for example, “The new year’s party was 3 months ago”).
c. If more than a year, refer to the number of years, the String will contain the number of years (for example “The last Olympic games were 4 years ago”).

Requirement 2: Print the duration of an event, if the event took more than a day
For example, “The climbing competition lasted for 3 days”.
If the events last for month or years, use the appropriate timescales and values.

Requirement 3:
Print the time that passed between two events, provided they are in chronological order.
For example, “There were 14 weeks between the start of the course, and the exam”.

Tasks:

1. Write a set of test criteria that evaluate if the requirements are met or not. If you feel that the requirements are not clear or specific enough, motivate why and make a justified assumption for your plan.
2. Write a test plan for the system, and update it as needed.
3. Implement the System Under Test (SUT) and the tests. Make sure to include unit tests, integration tests, acceptance tests. You can use TDD or any other techniques, but try to write the tests (especially unit tests) either before or as you are writing the code (rather than long after).
4. Pick a method (one that has interesting behavior) and discuss how the different tests you have written follow different paths through that method.

The final deliverables will be:

- (Group) The test plan (updated during development, of course).
- (Group) The SUT and test code (archived in a single .zip file)
  (The system should compile and run. You should have a Readme file (readme.md) that describes how to compile, run, and what to expect from your system. All the tests should pass, coverage should be above 50%)
- (Group) A document (no more than 10 pages), discussing your reflections on the project. An overview of the project progress, the development work (how were roles divided?, how did the team manage?, how much did each member contribute and how?), and the testing you have conducted. A discussion on what tests you could include further. A discussion on any obstacles you have encountered and how you overcame them.

Deadline: April 28, 2020.