

Principles of Software Design

Introduction

Robert Lukotka

`lukotka@dcs.fmph.uniba.sk`

`www.dcs.fmph.uniba.sk/~lukotka`

M-255

What is the difference between real world project and your typical class assignment?

Class assignment.

- The assignment is clearly stated and complete.
- You work on it for short amount of time, then you forget about.
- You work on your own.
- The resulting product is small, easy to comprehend by a single person.
- The underlying reason on why to do the assignment is not your concern.
- There are no changes about your assignment during your work.
- ...

Areas of knowledge [1]

- Software requirements
- Software design
- Software construction
- Software testing
- Software maintenance
- Software configuration management
- Software engineering management
- Software engineering process
- Software engineering models and methods
- Software quality
- Software engineering professional practice
- Software engineering economics

Areas of knowledge [1]

- Computing foundations
- Mathematical foundations
- Engineering foundations

Supporting:

- Computer engineering
- Systems engineering
- Project management
- Quality management
- General management
- Computer science
- Mathematics

Areas of knowledge

- The previous courses only touched some of these areas.
- We give a very basic overview of most of these areas.

That is why you have these subjects:

- **Principles of software design (1)**
- Principles of software design (2)

In the first course we will (mostly) be more focused on things more close to actual programming. So you can use things you learn in Project (1) and Project (2).

What to expect and what should you focus on

- No theorem or proofs.
- Good practices - things that are right in typical setting.
 - amplify the strong sides of the approach
 - manage the negative aspects
- But you also need to know why to follow these practices so you know when to make exceptions.
- How given practice affects other aspects of software development.

Course overview

- Software development process and software contracts
Lifecycle of software. Importance of feedback in software development. How typical software contract look like. How these choices affect the resulting software.
- git
Most widely used version control system. Makes it easier to handle many versions in the software system. Facilitates work in groups.
- Python and programming paradigms
These will be the languages used here (with more focus on Python). The goal is to show how the choice of the language (and how it will be used) affects software development.

Course overview

- Software design
How to split our problem in separate, easily tractable pieces prepared for implementation.
- Software quality
How to manage software quality. How to keep code good enough for others to work on. How to keep code good enough to be used.

Grading

- One assignment, 60 points, you need at least 30 points to pass. Deadline is late in the semester so we can go through the topics necessary to finish it.
- Oral examination 60 points (two open questions from a published set of questions).

We will have almost all lectures in October so you can focus on the project in November.

Grading

- You have to use Git and Java/Python to complete your assignments (And if you use Python you have to follow style that enforces strong type checking similar to Java).
- The project has team part and individual part.
- The project will be a board game of medium complexity that requires approximately 10-20 mandays to finish.
- You will work collaboratively to do the bulk of the work leaving only a few classes and integration tests for you.
- The individual part will be where the most points come from.

Grading

- Grading:

$A \quad \langle 120, 110 \rangle$

$B \quad \langle 110, 100 \rangle$

$C \quad \langle 100, 90 \rangle$

$D \quad \langle 90, 80 \rangle$

$E \quad \langle 80, 72 \rangle$

Resources

On this frame you usually find materials to learn from. You should read this stuff to prepare for the examination (especially if you missed the course or were sleeping during my presentation :)).

References I

Sometimes I feel it is necessary to give you the source of some claim (I should do that more often then I do). This will appear here. You don't have to read the content of it.



[SWEBOK V3 -TOC](#)