

# Software Engineering Project

David Baelde    Samy Jaziri

`{baelde,jaziri}@lsv.ens-cachan.fr`

MPRI, ENS Cachan, 2015/2016

# Goals

## Project

- ▶ Realistic software development experience.
- ▶ Produce finalized application, usable and maintainable.
- ▶ More programming experience.

## Software Engineering

- ▶ Know what makes a good design.
- ▶ Large code base, large team, long time.
- ▶ Value modularity, use interface specifications.
- ▶ Learn practical rigorous development techniques.

# Organization

## Lectures

- ▶ Principles of software engineering
- ▶ Architecture, design and modeling
- ▶ Rigorous software development, testing and debugging
- ▶ Tutorials on various useful tools

## Project

- ▶ Complete software project, from design to release & demo
- ▶ Mandatory project meetings whenever no lecture
- ▶ Evaluation based on final release and individual activity

(Lecture 1)

# Your project

Next session: **you** present project proposals. After discussions, the class chooses its project and students commit to participate.

## Guidelines

- ▶ **Large**, ideally one project for everybody  $\rightsquigarrow$  be ambitious!
- ▶ **Modular**  $\rightsquigarrow$  propose a decomposition in sub-projects.
- ▶ **Interesting**  $\rightsquigarrow$  exploit various skills, make it fun or useful.

Have an idea? Prepare a short presentation and let us know.

# Your project

Next session: **you** present project proposals. After discussions, the class chooses its project and students commit to participate.

## Guidelines

- ▶ **Large**, ideally one project for everybody  $\rightsquigarrow$  be ambitious!
- ▶ **Modular**  $\rightsquigarrow$  propose a decomposition in sub-projects.
- ▶ **Interesting**  $\rightsquigarrow$  exploit various skills, make it fun or useful.

Have an idea? Prepare a short presentation and let us know.

## Technical choices (not necessarily part of the proposal)

Team chooses its language, tools, development environment, etc.

- ▶ Choice must be motivated, will be discussed and criticized
- ▶ Requirement: tools freely available on linux
- ▶ Requirement: code repository + tracker, e.g., github

## Project ideas (1/2)

### Graphical editor

- ▶ Primitives (vectorial, animation, 3D...)
- ▶ Edition interfaces (desktop, web...)
- ▶ File formats, scripting
- ▶ Output formats

### Video game

- ▶ Engine (e.g., physical)
- ▶ Graphics, music, UI(s)
- ▶ Artificial intelligence
- ▶ Networking
- ▶ Scripting languages (level design, user scripts)

## Project ideas (2/2)

### Mail client

- ▶ Abstraction over protocols
- ▶ UI: filtering, actions, cron, scripting
- ▶ Learning (intelligent filtering)
- ▶ Statistics and visualization

### ENS Cachan 2.0

- ▶ Core (database, web API)
- ▶ Front-ends: web, mobile
- ▶ Agenda
- ▶ Secure cafette “payments”
- ▶ Recommendations