Course Organisation
LINGI 2255

2017–2018

Prof. Kim Mens

moodleucl.uclouvain.be

Course: LINGI2255

LINGI2255
Software Development Project

- 1st semester
- 6 ECTS
- 30h + 30h
- Master: INFO2M, SINF2M,SINF2M1
- Language: English

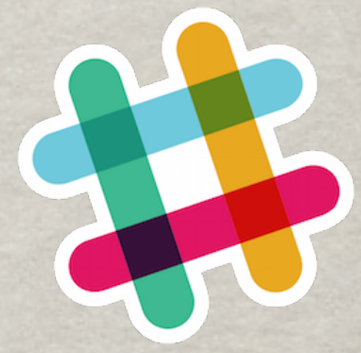
LINGI2255
Software Development Project

- Course website: (will open soon)

  - https://moodleucl.uclouvain.be/course/view.php?id=7599

  - official course announcements, deadlines and deliverables

- Slack:



  - We'll use Slack as our communication medium

  - Slack is a messaging app for development teams

  - https://lingi2255project.slack.com

LINGI2255
Software Development Project

- Teacher:
  - Kim Mens — kim.mens@uclouvain.be
- Assistants:
  - Michaël Saint-Guillain — michael.saint@uclouvain.be
  - Benoît Duhoux — benoit.duhoux@uclouvain.be
  - John Aoga — john.aoga@uclouvain.be
  - Xavier Gillard — xavier.gillard@uclouvain.be
  - Quang Minh Ha — quang.ha@uclouvain.be

LINGI2255
Software Development Project

- This project consists of the **development of a realistic application**, representative of a typical industrial software system, under semi-professional working conditions.

- The topic of the software system to be constructed is **proposed by a non-profit organisation** who participates in the organisation and evaluation of this course.

- The project will be carried out by **teams of 6 to 8 students**, each supervised by an assistant.

- **Communication** among team and with client is **crucial**.

LINGI2255
Software Development Project

Client:

- Non-profit organisation : Eureduka
- Product : https://www.oscar.education
- **Laurent Fourny** laurent.fourny@oscar.education
- Maher Chemseddine maher.chemseddine@masi.henallux.be
- **Robin Descamps** robin.descamps@student.uclouvain.be

Software Development Project 2017-18

Tentative Project Schedule (Courses)

week	course activity
1	Course introduction by teacher
	Presentation of case by client
	Technical presentation by client
2	*no course (legal holiday)*
3	Introduction to requirements phase
	Q&A session with client
4	Planning and architecture
5	Feedback on requirements documents from client
6	*Permanence and feedback with client*
7	*no course (legal holiday)*
8	show 1st prototype to client
9	*no course session planned (yet)*
10	*Permanence and feedback with client*
11	show 2nd prototype to client
12	*Permanence and feedback with client*
13	*Permanence and feedback with client*
14	Demo and defence of final product

Tentative Project Schedule (Sessions with client)

week	course activity
1	Course introduction by teacher
	Presentation of case by client
	Technical presentation by client
2	*no course (legal holiday)*
3	Introduction to requirements phase
	Q&A session with client
4	Planning and architecture
5	Feedback on requirements documents from client
6	*Permanence and feedback with client*
7	*no course (legal holiday)*
8	show 1st prototype to client
9	*no course session planned (yet)*
10	*Permanence and feedback with client*
11	show 2nd prototype to client
12	*Permanence and feedback with client*
13	*Permanence and feedback with client*
14	Demo and defence of final product

Project Phases

Project split in several **phases** of ~2 weeks

- **Phase 0** : **Getting acquainted** with case and technology (2 weeks)
- **Phase 1** : **Requirements** analysis (2 weeks)
- **Phase 2** : **Design** (2 weeks)
- **Phase 3** : **Implementation** of first prototype (2 weeks)
- **Integration** **Merge** and integrate prototypes (1 week)
- **Phase 4** : **Implementation** of second prototype (2 weeks)
- **Integration** **Merge** and integrate prototypes (1 week)
- **Phase 5** : **Reporting** and project defence (2 weeks)

Expected Deliverables

- Project split in several **phases** of ~2 weeks
  - **Phase 0** : Database model
  - **Phase 1** : Requirements documents (10 pages)
  - **Phase 2** : Design documents
  - **Phase 3** : First prototype
  - **Integration** Prototype integrated and merged with others
  - **Phase 4** : Second prototype (2 weeks)
  - **Integration** Prototype integrated and merged with others
  - **Phase 5** : Final reports, manuals, presentation, …

Project Schedule (deadlines)

	W	phase	deliverable		deadline	
0	1-2	getting acquainted	**database model**		end W2	
1	3-4	requirements	**requirements document**		end W4	10%
2	5	design	**design documents**		end W6	5%
3	6-7	first prototype	**1st working prototype**		end W7	20%
I	8	integration	**prototype integrated**		end W8	
2	9	design	**design documents**		end W9	5%
4	9-1	second prototype	**2nd working prototype**		end W10	20%
I	12	integration	**prototype integrated**		end W11	
5	12	reporting	**documentation**		end W12	15%
	13	final product	**user manual**	**code, tests**	end W13	
	14	demo & presentation	**project**		W14	25%

Project Evaluation

Continuous evaluation:

- Intermediate prototypes : **40%** (20% each)
- Intermediate documents : **20%** (requirements and design documents)
- Final reports and documentation : **15%**
- Presentation and demonstration of the final product : **25%**
  - delivered system, documentation, tests, presentation, demonstration, internal product quality, product usability, team organisation, client appreciation, …

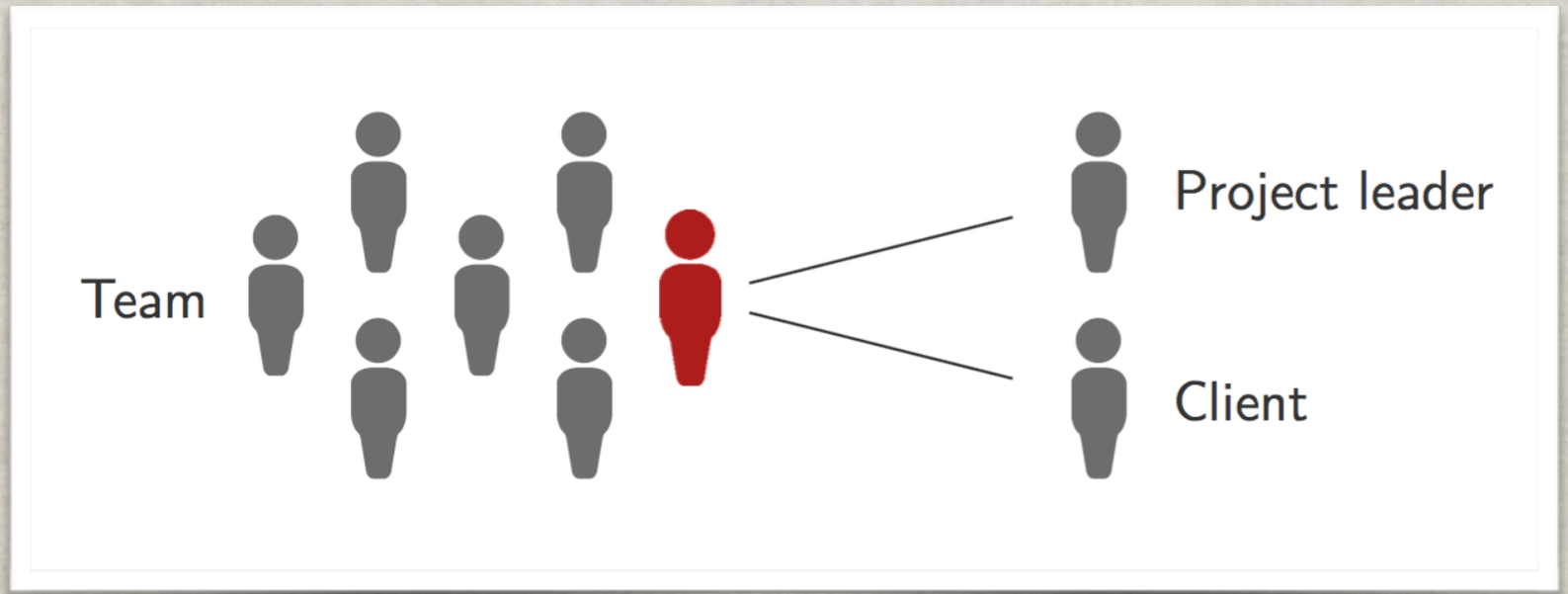
No possibility to do a second session for this course **!**

Points of Attention

- **Timing** 
  - Start early; work regularly; respect planning & deadlines
  - 75% of final grade is based on intermediate evaluation!
- **Team** 
  - Organise your team
  - Assign responsibilities
  - Distribute work among team members

Team Organisation

- **Teams** of 6-8 developers
  - assign one team member as (internal) **team manager**
- Teams overseen by a **project leader**
  - project leader = a (pair of) course assistant(s)



Team Organisation

- **Weekly meetings** with project leader (obligatory)
  - presenting progress and difficulties
  - assessing alternative options
  - distribution of work within team
  - feedback on intermediate and progress reports
  - monitoring of schedule

Team Organisation

- Role of (internal) project manager
  - Spokesperson for the team
  - Delegates responsibilities to team members
- Responsibilities to be mentioned in reports
  - And discussed with project leader during weekly meetings
- Don't exclude newcomers from your team (added in week 2)
  - managing unexpected situations is a skill to be learned, as well as working in heterogeneous teams

Project Defence

- In week 14 (just before the Christmas holidays)
  - dates and hours to be determined (could be an evening)
  - with the client, your project leader and the professor
    - presence of *all* team members is mandatory
- Two parts (40 minutes)
  - 15 minutes of demonstration
  - 25 minutes of presentation (including questions)

Getting Started...

- **Join Slack** (***now***) and start finding team members
- **Register to Moodle** (***Thursday or Friday***) and create teams of 5
  - at least two French-speaking members per team
  - deadline for initial team creation: Fri 22.08
    - up to 2 (exceptionally 3) extra members may be added by professor
    - Project leader (assistant) will be assigned to teams immediately after creation
    - Choose an internal team manager (student)
    - Team managers contact project leader to fix date for weekly team meeting
      - first meeting must be held **before** end of week 2

Next Steps...

- **Learn technology: Python, Django, GitHub, ...**
- **Download and install code base, development tools, versioning system.**
- Get to know your team.
- Discuss and **choose module to implement.**
- Make detailed ORM or ER model of the existing database.
- **Read case specification** and technical description (will be available this Friday)
- Team manager can contact client if things are unclear (email / Slack)
    - for questions or clarifications about case description
    - client will also be present in week 3 for Q&A session

Attribution-ShareAlike 4.0 International (CC BY-SA 4.0)

You are free to:

Share — copy and redistribute the material in any medium or format

Adapt — remix, transform, and build upon the material for any purpose, even commercially.

The licensor cannot revoke these freedoms as long as you follow the license terms.

Under the following terms:

Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

ShareAlike — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

No additional restrictions — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

https://creativecommons.org/licenses/by-sa/4.0/