Welcome

GiPHouse Spring 2022 Employees

Welcome

GiPHouse Spring 2022 Employees

GiPHouse Student Project “Company” (founded in 1992)

Real projects, real-life customers.

Welcome

GiPHouse Spring 2022 Employees

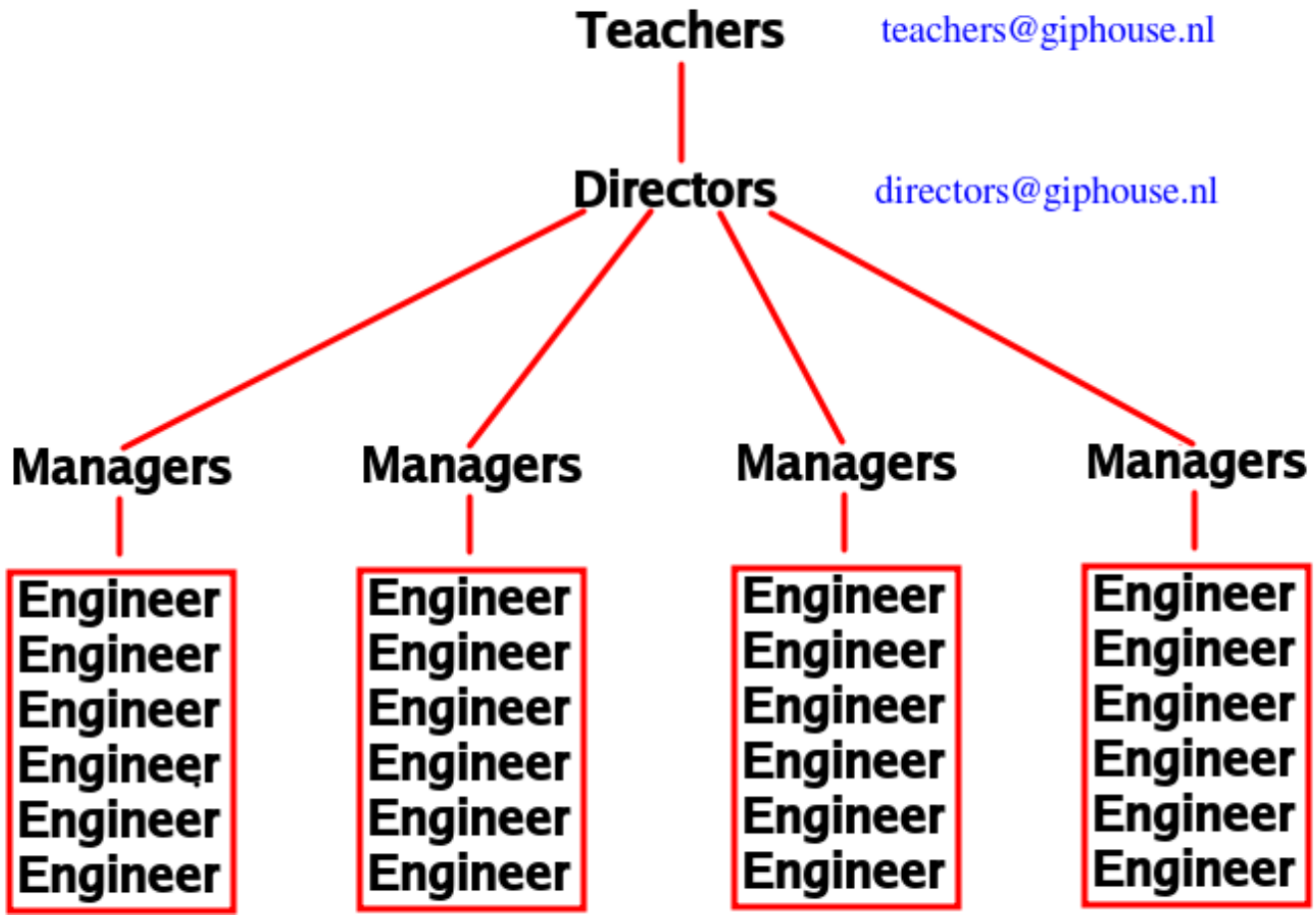
GiPHouse Student Project "Company" (founded in 1992)

Real projects, real-life customers.

Courses:

- Bachelor
  - Software Engineering
    (97 students as of 28-01)
- Master
  - System Development Management
    (24 students as of 28-01)
  - Software Development Entrepreneurship (fall)

GiPHouse Organisation



Teachers

Software Engineering (IBI001)

Cynthia Kop, Diego Nascimento Damasceno, Marc Schoolderman




System Development Management (IMC021)

Cynthia Kop, Edwin Hendriks, Marc Schoolderman

Directors

System Development Management and Software Engineering

Stijn Derks (CEO), Lars van Rhijn (COO)




Software Engineering

Jen Dusseljee (CTO), Job Doesburg (CIO)

Goals of SE and SDM

Goals of SE and SDM

Software Engineering

- develop a realistic software product in a SCRUM team;

Goals of SE and SDM

Software Engineering

- develop a realistic software product in a SCRUM team;
- apply agile practices such as the SCRUM stand-up and pair programming for effective collaboration;

Goals of SE and SDM

Software Engineering

- develop a realistic software product in a SCRUM team;
- apply agile practices such as the SCRUM stand-up and pair programming for effective collaboration;
- work as a team by planning around skill differences, helping each other and addressing individual problems;

Goals of SE and SDM

Software Engineering

- develop a realistic software product in a SCRUM team;
- apply agile practices such as the SCRUM stand-up and pair programming for effective collaboration;
- work as a team by planning around skill differences, helping each other and addressing individual problems;
- create high-quality code using design principles and software patterns;

Goals of SE and SDM

Software Engineering

- develop a realistic software product in a SCRUM team;
- apply agile practices such as the SCRUM stand-up and pair programming for effective collaboration;
- work as a team by planning around skill differences, helping each other and addressing individual problems;
- create high-quality code using design principles and software patterns;
- apply systematic testing techniques to deliver demonstrably correct code.

Goals of SE and SDM

Software Engineering

- develop a realistic software product in a SCRUM team;
- apply agile practices such as the SCRUM stand-up and pair programming for effective collaboration;
- **work as a team by planning around skill differences, helping each other and addressing individual problems;**
- create high-quality code using design principles and software patterns;
- apply systematic testing techniques to deliver demonstrably correct code.

Goals of SE and SDM

System Development Management

Goals of SE and SDM

System Development Management

- manage your team by identifying and removing obstacles and keeping meetings on track;

Goals of SE and SDM

System Development Management

- manage your team by identifying and removing obstacles and keeping meetings on track;
- manage your client by setting good expectations, adapting to changing requirements as appropriate and communicating clearly;

Goals of SE and SDM

System Development Management

- manage your team by identifying and removing obstacles and keeping meetings on track;
- manage your client by setting good expectations, adapting to changing requirements as appropriate and communicating clearly;
- manage your superiors by providing transparency in your team's progress and communicating your planning and potential problems in a timely manner;

Goals of SE and SDM

System Development Management

- manage your team by identifying and removing obstacles and keeping meetings on track;
- manage your client by setting good expectations, adapting to changing requirements as appropriate and communicating clearly;
- manage your superiors by providing transparency in your team's progress and communicating your planning and potential problems in a timely manner;
- build a smooth working atmosphere for your team.

Goals of SE and SDM

System Development Management

- manage your team by identifying and removing obstacles and keeping meetings on track;
- manage your client by setting good expectations, adapting to changing requirements as appropriate and communicating clearly;
- manage your superiors by providing transparency in your team's progress and communicating your planning and potential problems in a timely manner;
- build a smooth working atmosphere for your team.

Overall: at the end of the course you will have the skills of an IT project leader.

Goals of SE and SDM

The goals of both courses are achieved via lectures and a **real software project for a real customer** in the context of the student-run GiPHouse "Company" with students in various roles:

- team members (SE);
- team managers (SDM).

Course structure

Theory: Software Engineering
Practical Lab: GiPHouse Team Members

Theory: System Development Management
Practical Lab: GiPHouse Management

Course structure

Theory: Software Engineering
Practical Lab: GiPHouse Team Members

Theory: System Development Management
Practical Lab: GiPHouse Management

- Managers work just as hard as team members
  - Besides their main role as group managers, SDM students also have a mini-project.

Course structure

Theory: Software Engineering
Practical Lab: GiPHouse Team Members

Theory: System Development Management
Practical Lab: GiPHouse Management

- Managers work just as hard as team members
  - Besides their main role as group managers, SDM students also have a mini-project.
- Theory prepares for the practical work and also provides context, general theory, methods, techniques, and guidance for future projects.
  - SE lectures only during the first quarter.
  - SDM lectures concentrated in the first quarter, but some later.
  - You apply the theory in your project so that you can motivate its applicability and its advantages and disadvantages.

Way of working

Way of working

- Iterative, agile development in three-week cycles (roughly).

Way of working

- Iterative, agile development in three-week cycles (roughly).
- Working code / infrastructure after each sprint.

Way of working

- Iterative, agile development in three-week cycles (roughly).
- Working code / infrastructure after each sprint.
- Working closely with clients.
  - Talk to the client as much as possible.
  - Minimum: once every sprint.

Timeline

Q3 W1	SHARED	01 feb	02 feb	SDM	04 feb	11
Q3 W2	SHARED	08 feb	SHARED	SDM	11 feb	11
Q3 W3	SE	15 feb	16 feb	SDM	18 feb	11
Q3 W4	SE	22 feb	23 feb	SDM	25 feb	11
Q3 W5	HOLIDAY	HOLIDAY	02 mar	03 mar	PLAN	4
Q3 W6	SE	08 mar	09 mar	SDM	11 mar	11
Q3 W7	PRESENT	15 mar	16 mar	17 mar	18 mar	11
ELP	21 mar	22 mar	23 mar	24 mar	25 mar	2
EXAMS	28 mar	29 mar	30 mar	31 mar	01 apr	2
EXAMS	04 apr	05 apr	06 apr	07 apr	08 apr	2
Q4 W1	11 apr	12 apr	13 apr	14 apr	HOLIDAY	11
Q4 W2	HOLIDAY	19 apr	20 apr	21 apr	22 apr	4
Q4 W3	25 apr	26 apr	27 apr	28 apr	29 apr	11
HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	2
Q4 W4	9 may	10 may	11 may	12 may	13 may	11
Q4 W5	16 may	17 may	18 may	19 may	20 may	11
Q4 W6	23 may	24 may	25 may	HOLIDAY	HOLIDAY	11
Q4 W7	30 may	31 may	01 jun	02 jun	03 jun	11
Q4 W8	HOLIDAY	07 jun	08 jun	09 jun	DELIVER	4

Timeline

Q3 W1	SHARED	01 feb	02 feb	SDM	04 feb	11
Q3 W2	SHARED	08 feb	SHARED	SDM	11 feb	11
Q3 W3	SE	15 feb	16 feb	SDM	18 feb	11
Q3 W4	SE	22 feb	23 feb	SDM	25 feb	11
Q3 W5	HOLIDAY	HOLIDAY	02 mar	03 mar	PLAN	4
Q3 W6	SE	08 mar	09 mar	SDM	11 mar	11
Q3 W7	PRESENT	15 mar	16 mar	17 mar	18 mar	11
ELP	21 mar	22 mar	23 mar	24 mar	25 mar	2
EXAMS	28 mar	29 mar	30 mar	31 mar	01 apr	2
EXAMS	04 apr	05 apr	06 apr	07 apr	08 apr	2
Q4 W1	11 apr	12 apr	13 apr	14 apr	HOLIDAY	11
Q4 W2	HOLIDAY	19 apr	20 apr	21 apr	22 apr	4
Q4 W3	25 apr	26 apr	27 apr	28 apr	29 apr	11
HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	2
Q4 W4	9 may	10 may	11 may	12 may	13 may	11
Q4 W5	16 may	17 may	18 may	19 may	20 may	11
Q4 W6	23 may	24 may	25 may	HOLIDAY	HOLIDAY	11
Q4 W7	30 may	31 may	01 jun	02 jun	03 jun	11
Q4 W8	HOLIDAY	07 jun	08 jun	09 jun	DELIVER	4

Timeline

- First half of February
  - Get to know your group.
    - meet and discuss in person
    - know each other's skills
    - discuss limitations, preferences and solutions
  - Understand / update / adapt project definition.
    - meet and discuss with your client
    - prioritised requirements list, basic risk assessment
    - key design decisions such as architecture
    - perhaps: make wireframes, drawings
  - Explore implementation.
    - Github
    - check and understand relevant libraries
    - create basic utilities
    - get familiar with programming language / framework

Timeline

Q3 W1	SHARED	01 feb	02 feb	SDM	04 feb	11
Q3 W2	SHARED	08 feb	SHARED	SDM	11 feb	11
Q3 W3	SE	15 feb	16 feb	SDM	18 feb	11
Q3 W4	SE	22 feb	23 feb	SDM	25 feb	11
Q3 W5	HOLIDAY	HOLIDAY	02 mar	03 mar	PLAN	4
Q3 W6	SE	08 mar	09 mar	SDM	11 mar	11
Q3 W7	PRESENT	15 mar	16 mar	17 mar	18 mar	11
ELP	21 mar	22 mar	23 mar	24 mar	25 mar	2
EXAMS	28 mar	29 mar	30 mar	31 mar	01 apr	2
EXAMS	04 apr	05 apr	06 apr	07 apr	08 apr	2
Q4 W1	11 apr	12 apr	13 apr	14 apr	HOLIDAY	11
Q4 W2	HOLIDAY	19 apr	20 apr	21 apr	22 apr	4
Q4 W3	25 apr	26 apr	27 apr	28 apr	29 apr	11
HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	2
Q4 W4	9 may	10 may	11 may	12 may	13 may	11
Q4 W5	16 may	17 may	18 may	19 may	20 may	11
Q4 W6	23 may	24 may	25 may	HOLIDAY	HOLIDAY	11
Q4 W7	30 may	31 may	01 jun	02 jun	03 jun	11
Q4 W8	HOLIDAY	07 jun	08 jun	09 jun	DELIVER	4

Timeline

- February–March: sprint 1
- Become a team.
  - continue meeting regularly (face-to-face!)
  - divide work
  - report, give feedback
- Start coding.
  - develop top priorities / code framework
  - design code to be maintainable
  - set up basic unit testing for non-trivial code
  - aim to have working code at the end of the sprint
- **14 March: initial presentations**

Timeline

Q3 W1	SHARED	01 feb	02 feb	SDM	04 feb	11
Q3 W2	SHARED	08 feb	SHARED	SDM	11 feb	11
Q3 W3	SE	15 feb	16 feb	SDM	18 feb	11
Q3 W4	SE	22 feb	23 feb	SDM	25 feb	11
Q3 W5	HOLIDAY	HOLIDAY	02 mar	03 mar	PLAN	4
Q3 W6	SE	08 mar	09 mar	SDM	11 mar	11
Q3 W7	PRESENT	15 mar	16 mar	17 mar	18 mar	11
ELP	21 mar	22 mar	23 mar	24 mar	25 mar	2
EXAMS	28 mar	29 mar	30 mar	31 mar	01 apr	2
EXAMS	04 apr	05 apr	06 apr	07 apr	08 apr	2
Q4 W1	11 apr	12 apr	13 apr	14 apr	HOLIDAY	11
Q4 W2	HOLIDAY	19 apr	20 apr	21 apr	22 apr	4
Q4 W3	25 apr	26 apr	27 apr	28 apr	29 apr	11
HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	2
Q4 W4	9 may	10 may	11 may	12 may	13 may	11
Q4 W5	16 may	17 may	18 may	19 may	20 may	11
Q4 W6	23 may	24 may	25 may	HOLIDAY	HOLIDAY	11
Q4 W7	30 may	31 may	01 jun	02 jun	03 jun	11
Q4 W8	HOLIDAY	07 jun	08 jun	09 jun	DELIVER	4

Timeline

- March / April / May: sprints 2 and 3
  - Continue coding.
    - add more features
    - refactor existing code where appropriate
    - deliver high-quality code
    - deliver working code at the end of each sprint (and if possible in between!)
  - Set up more extensive testing.
    - automated unit testing
    - continuous integration
    - checklists for manual testing if relevant
  - Adapt plans.
    - changing requirements
    - emerging problems
    - maintain good communication with your client

Timeline

Q3 W1	SHARED	01 feb	02 feb	SDM	04 feb	11
Q3 W2	SHARED	08 feb	SHARED	SDM	11 feb	11
Q3 W3	SE	15 feb	16 feb	SDM	18 feb	11
Q3 W4	SE	22 feb	23 feb	SDM	25 feb	11
Q3 W5	HOLIDAY	HOLIDAY	02 mar	03 mar	PLAN	4
Q3 W6	SE	08 mar	09 mar	SDM	11 mar	11
Q3 W7	PRESENT	15 mar	16 mar	17 mar	18 mar	11
ELP	21 mar	22 mar	23 mar	24 mar	25 mar	2
EXAMS	28 mar	29 mar	30 mar	31 mar	01 apr	2
EXAMS	04 apr	05 apr	06 apr	07 apr	08 apr	2
Q4 W1	11 apr	12 apr	13 apr	14 apr	HOLIDAY	11
Q4 W2	HOLIDAY	19 apr	20 apr	21 apr	22 apr	4
Q4 W3	25 apr	26 apr	27 apr	28 apr	29 apr	11
HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	2
Q4 W4	9 may	10 may	11 may	12 may	13 may	11
Q4 W5	16 may	17 may	18 may	19 may	20 may	11
Q4 W6	23 may	24 may	25 may	HOLIDAY	HOLIDAY	11
Q4 W7	30 may	31 may	01 jun	02 jun	03 jun	11
Q4 W8	HOLIDAY	07 jun	08 jun	09 jun	DELIVER	4

Timeline

- May / June: sprint 4 and completion
  - continue coding with a focus on maintainability
  - continue testing (unit tests, user acceptance tests)
  - finalise features in discussion with the client
  - deliver the result
  - **10 June: deadline for submitting final presentation**
  - **13–24 June: final project discussions**
    (possibly some groups in 27 Jun–01 Jul)

Timeline

- May / June: sprint 4 and completion
  - continue coding with a focus on maintainability
  - continue testing (unit tests, user acceptance tests)
  - finalise features in discussion with the client
  - deliver the result
  - **10 June: deadline for submitting final presentation**
  - **13–24 June: final project discussions**
    (possibly some groups in 27 Jun–01 Jul)

Requirements for the presentations will be posted on Brightspace.

Plans and schedules may change; updates will be posted on Brightspace.

Check the calendar feature on Brightspace.

Intermediate deliverables

For Software Engineering:

For System Development Management:

Intermediate deliverables

For Software Engineering:

- A Github repository of code.
  - source code
  - tests
  - documentation (requirements, code decisions, testing checklists, accompanying documents)
- Peer review

For System Development Management:

Intermediate deliverables

For Software Engineering:

- A Github repository of code.
  - source code
  - tests
  - documentation (requirements, code decisions, testing checklists, accompanying documents)
- Peer review

For System Development Management:

- Project plan
- Documentation for sprint planning, review and retrospective
- Peer review and cross-group reviews

Grading: software engineering

Grading: software engineering

Recall: learning objectives

- develop a realistic software product in a SCRUM team;
- apply agile practices such as the SCRUM stand-up and pair programming for effective collaboration;
- work as a team by planning around skill differences, helping each other and addressing individual problems;
- create high-quality code using design principles and software patterns;
- apply systematic testing techniques to deliver demonstrably correct code.

Grading: software engineering

1.5 quality of and individual contribution to the product (including presentation quality)
2.5 implementation of the Scrum process
3.0 teamwork: participation in meetings, working together, helping/leadership and contribution to the team synergy
1.5 code quality assignment
1.5 quality of automatic and (systematic!) manual testing

Grading: system development management

Recall: learning objectives

- manage your team by identifying and removing obstacles and keeping meetings on track;
- manage your client by setting good expectations, adapting to changing requirements as appropriate and communicating clearly;
- manage your superiors by providing transparency in your team's progress and communicating your planning and potential problems in a timely manner;
- build a smooth working atmosphere for your team.

Overall: at the end of the course you will have the skills of an IT project leader.

Grading: system development management

2.0	implementation Scrum process
2.0	client interaction and management
2.0	reports to superiors
2.0	individual leadership
0.5	presentation quality
1.5	assignment Edwin

Grading: where does it come from?

Teachers determine grades, based on:

- the final presentation
- customer satisfaction
- managers', directors' and teachers' impression
- code and documentation
- inter-group evaluations
- peer review
- final project discussion
- teacher assignments
- project reports by the managers

Final Project discussions

Clarify open questions regarding the process.

- The whole team (SE and SDM members) will discuss the project.
- Questions both to the group and to individuals.
- Some individuals may also be invited for a personal discussion.

Mandatory attendance?

- Key idea: skipping important theory is unfair to your teammates.

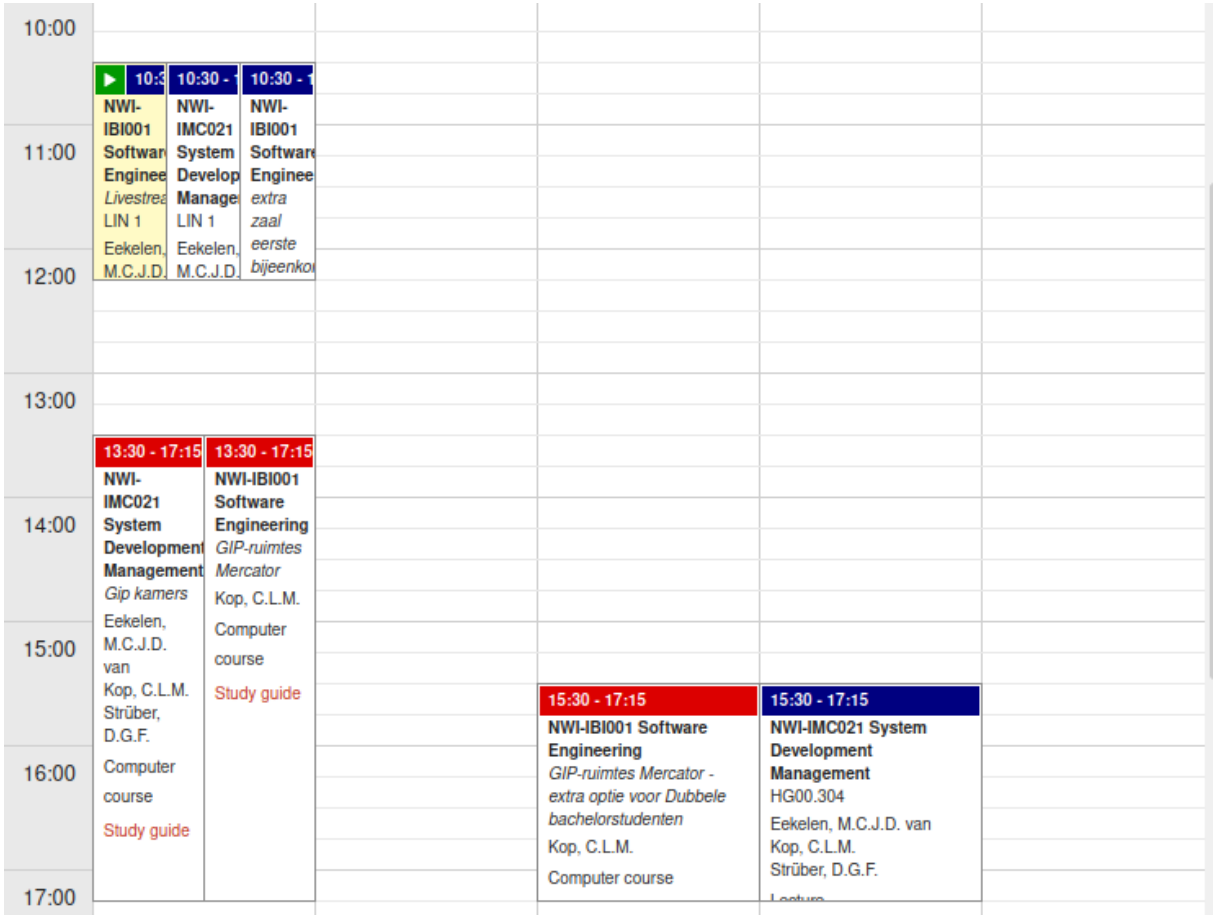
Mandatory attendance?

- Key idea: skipping important theory is unfair to your teammates.
- We expect everyone to view all presentations!

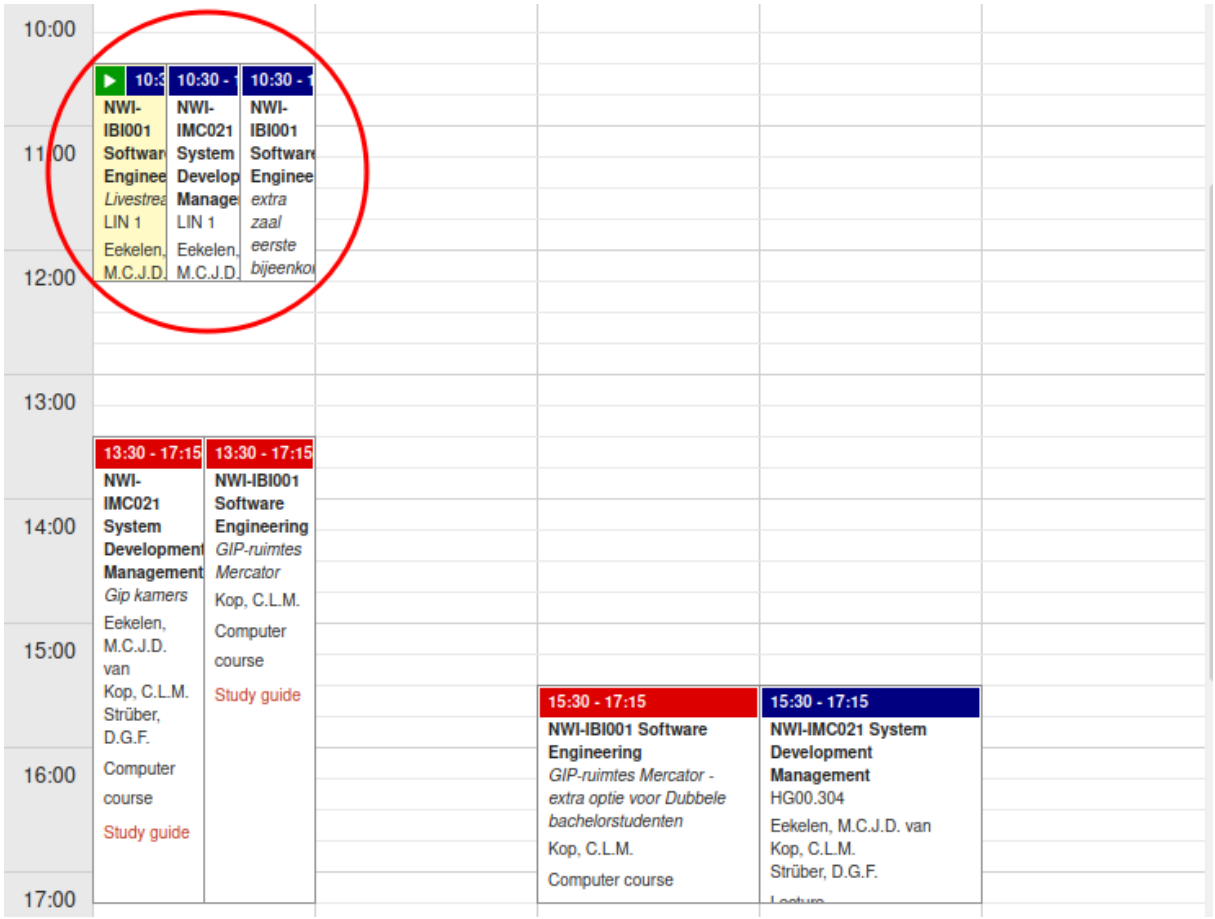
Mandatory attendance?

- Key idea: skipping important theory is unfair to your teammates.
- We expect everyone to view all presentations!
- For SE: no lectures in the fourth quarter!

Schedule

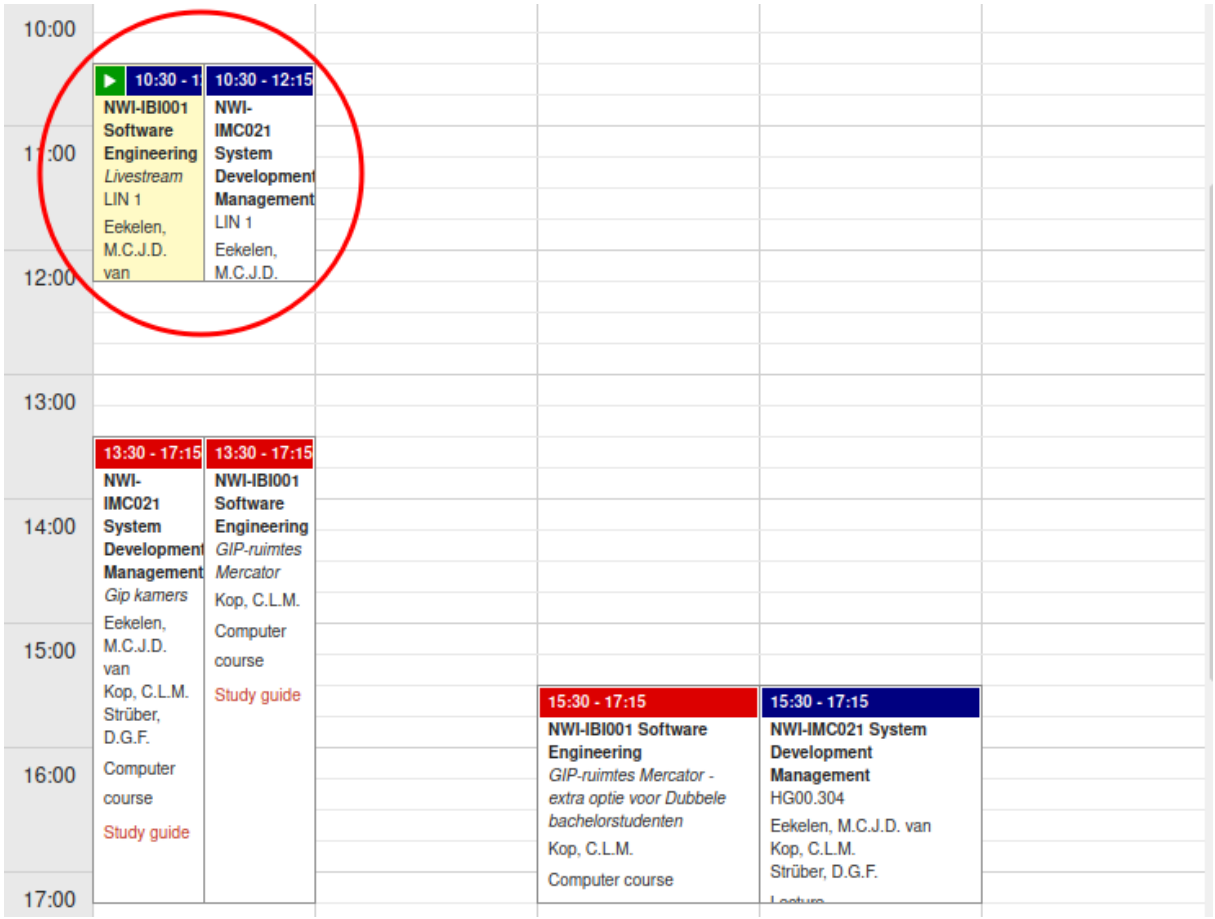

10:00
11:00
12:00
13:00
14:00
15:00
16:00
17:00
10:30 - 1
NWI-IBI001 Softwar Enginee
Livestrea
LIN 1
Eekelen, M.C.J.D.
10:30 - 1
NWI-IMC021 System Develop Manage
LIN 1
Eekelen, M.C.J.D.
10:30 - 1
NWI-IBI001 Softwar Enginee
extra zaal eerste bijeenko
13:30 - 17:15
NWI-IMC021 System Development Management
Gip kamers
Eekelen, M.C.J.D. van
Kop, C.L.M.
Strüber, D.G.F.
Computer course
Study guide
13:30 - 17:15
NWI-IBI001 Software Engineering
GIP-ruimtes Mercator
Kop, C.L.M.
Computer course
Study guide
15:30 - 17:15
NWI-IBI001 Software Engineering
GIP-ruimtes Mercator - extra optie voor Dubbele bachelorstudenten
Kop, C.L.M.
Computer course
15:30 - 17:15
NWI-IMC021 System Development Management
HG00.304
Eekelen, M.C.J.D. van
Kop, C.L.M.
Strüber, D.G.F.

Schedule



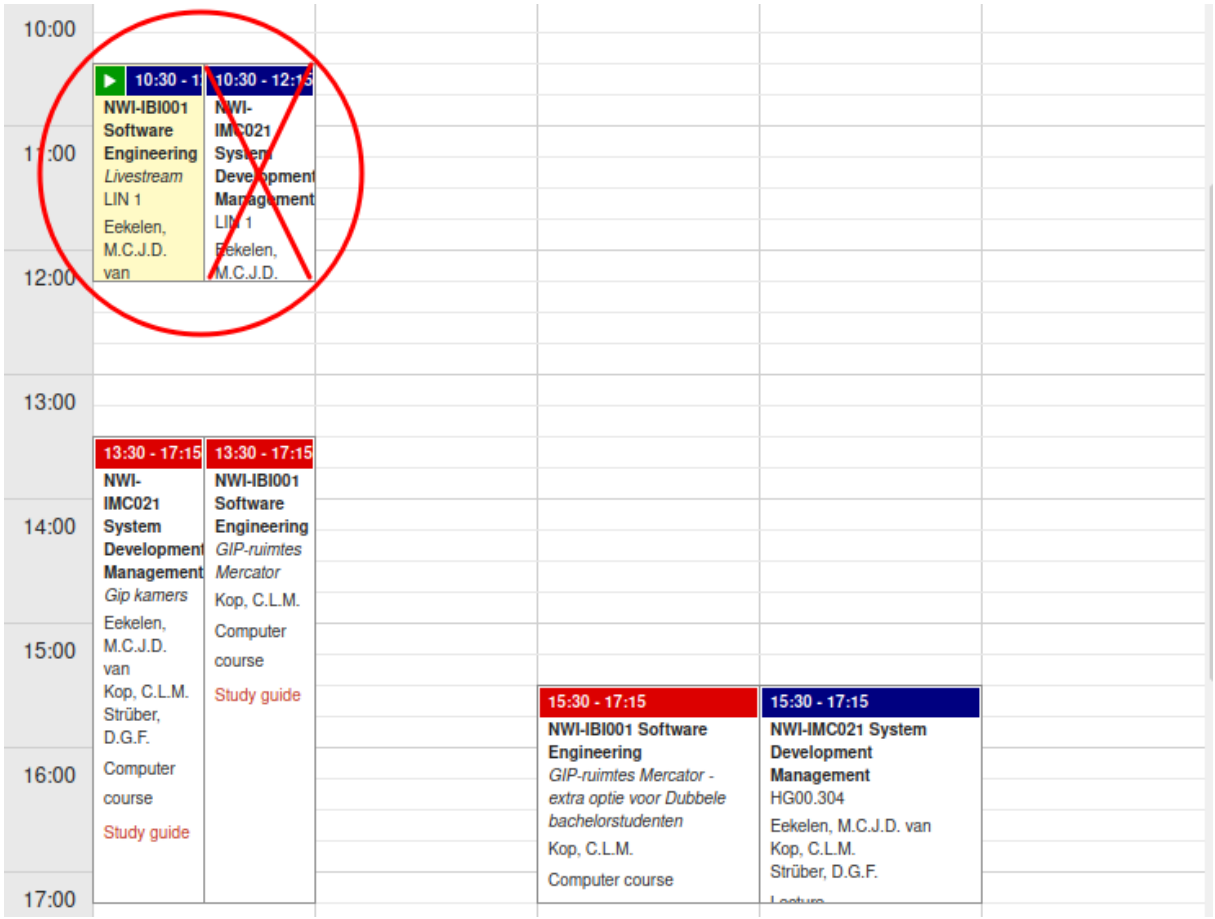
Monday morning: shared lectures – today and next week

Schedule



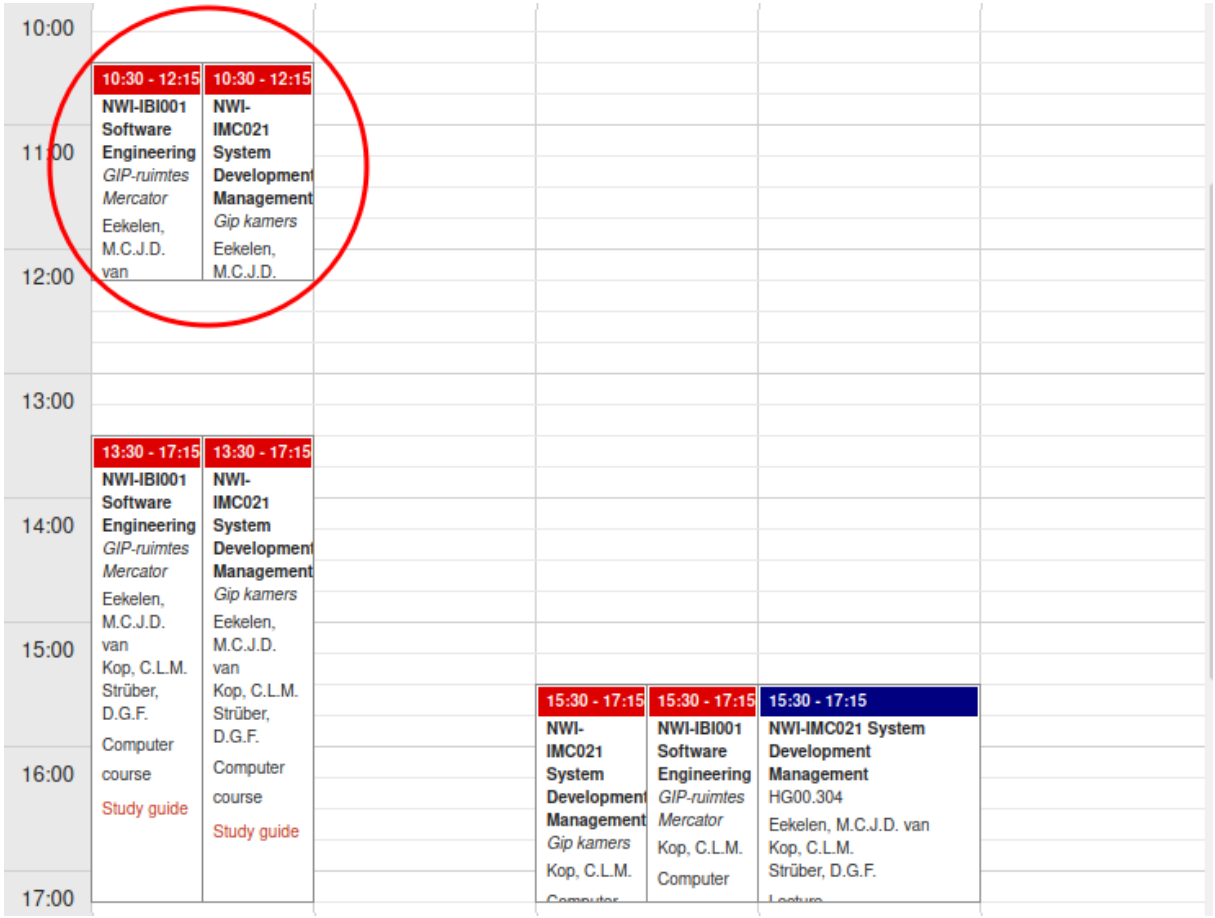
Monday morning: shared lectures – today and next week

Schedule



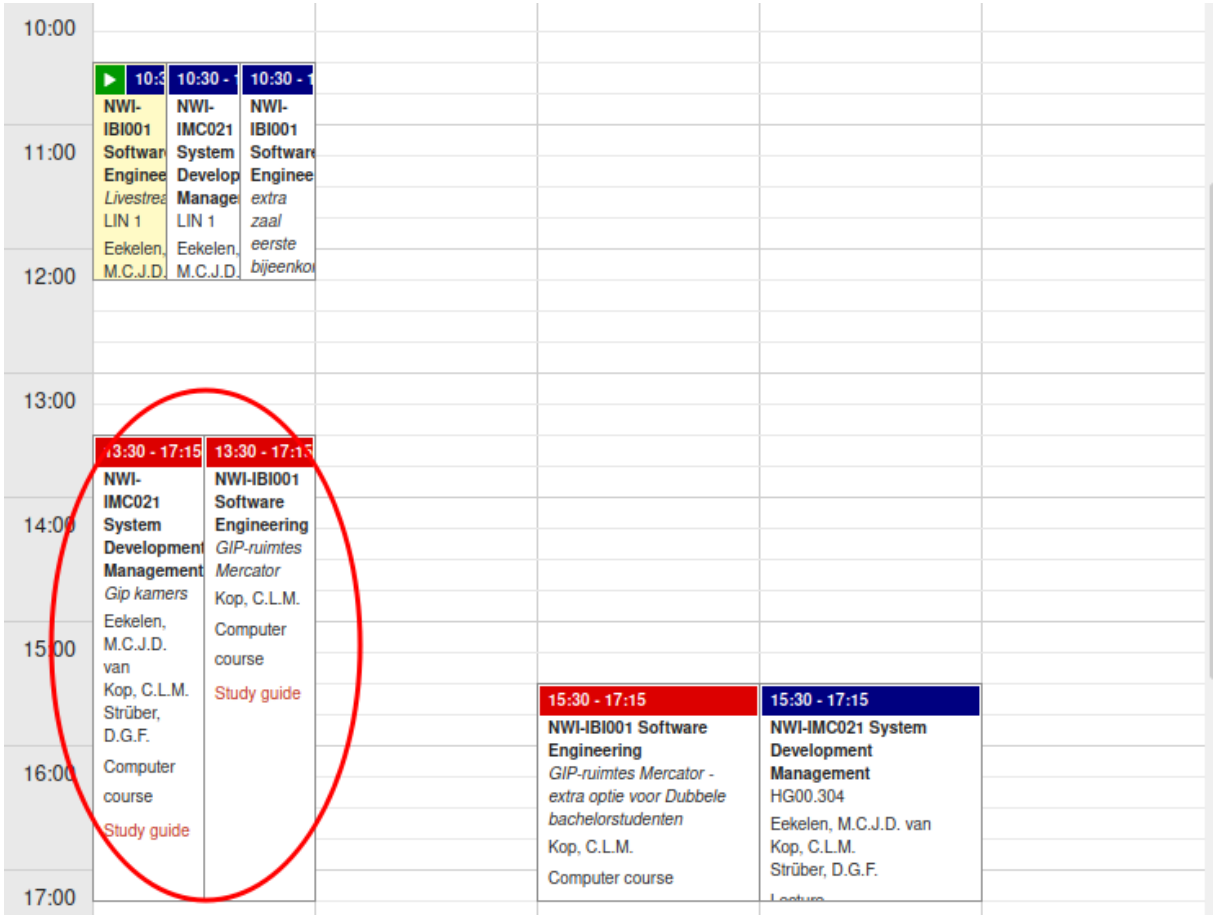
Monday morning: SE lectures in other weeks of quarter 3

Schedule



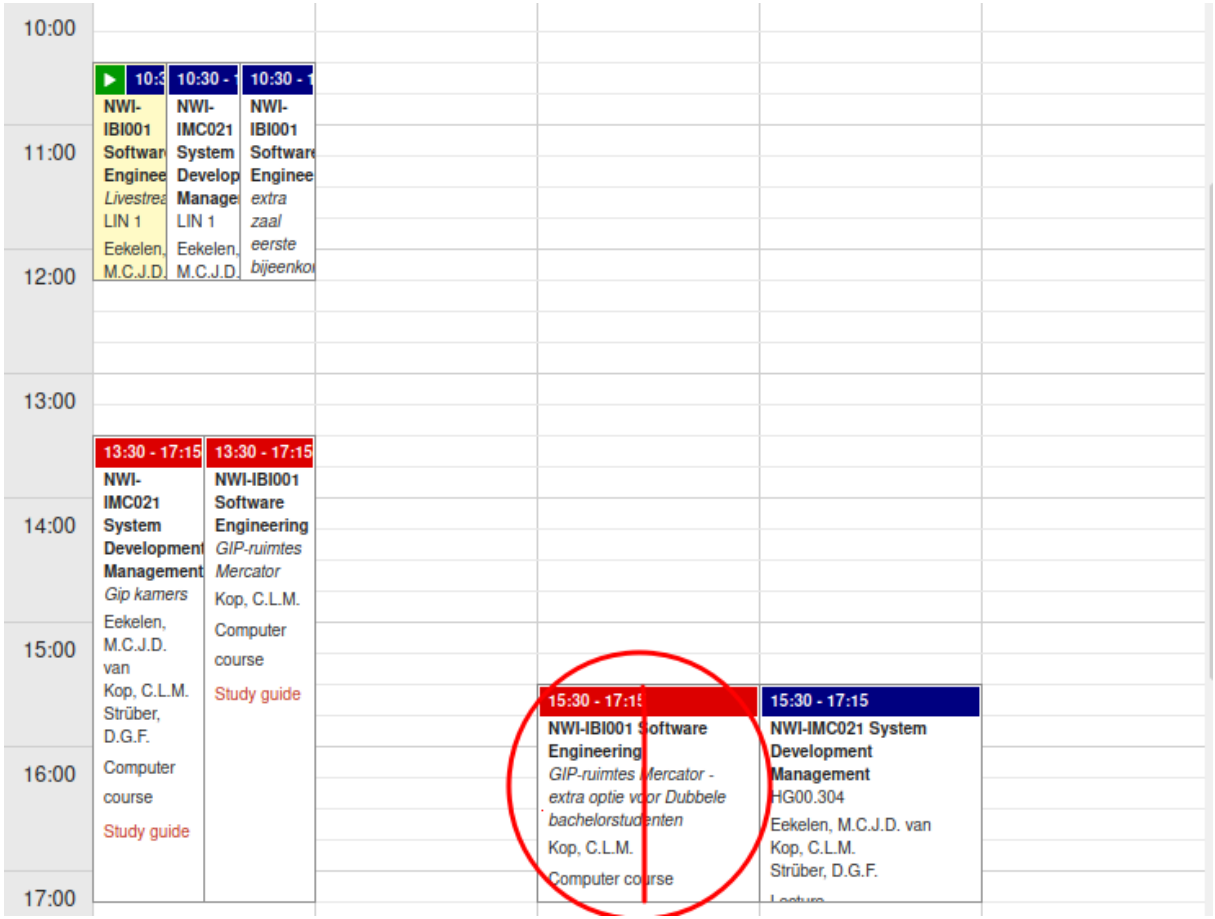
Monday morning: shared working time in quarter 4

Schedule



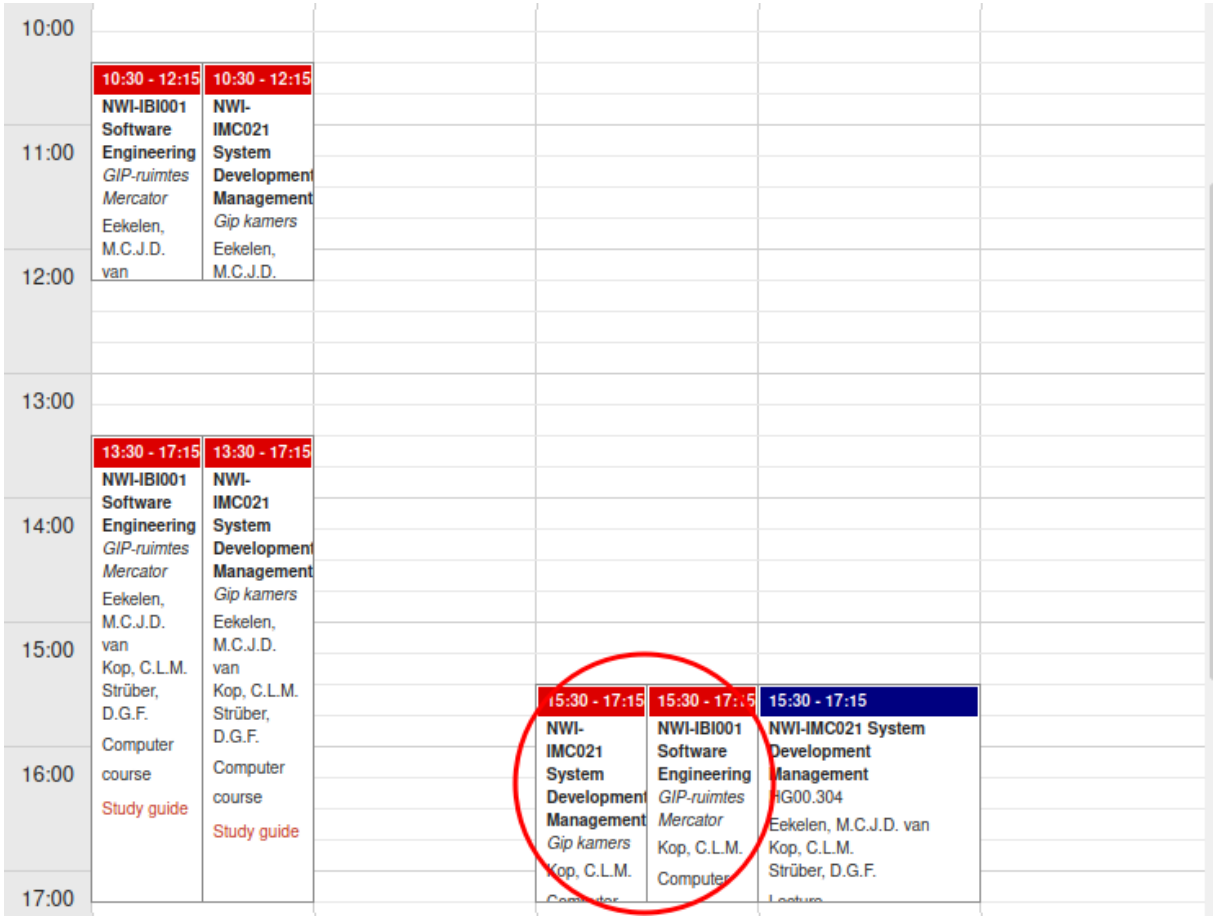
Monday afternoon: shared working time (other times are allowed)

Schedule



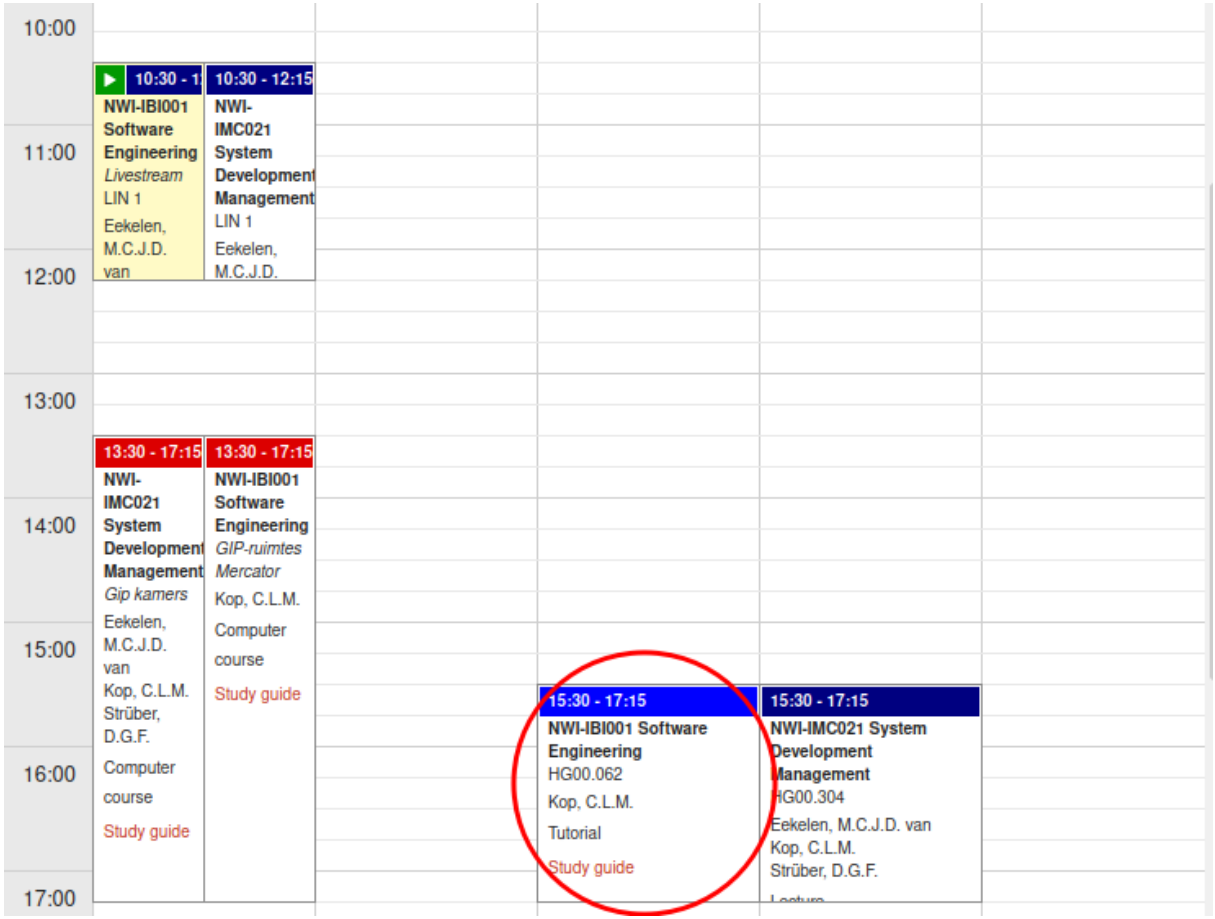
Wednesday afternoon: shared working time

Schedule



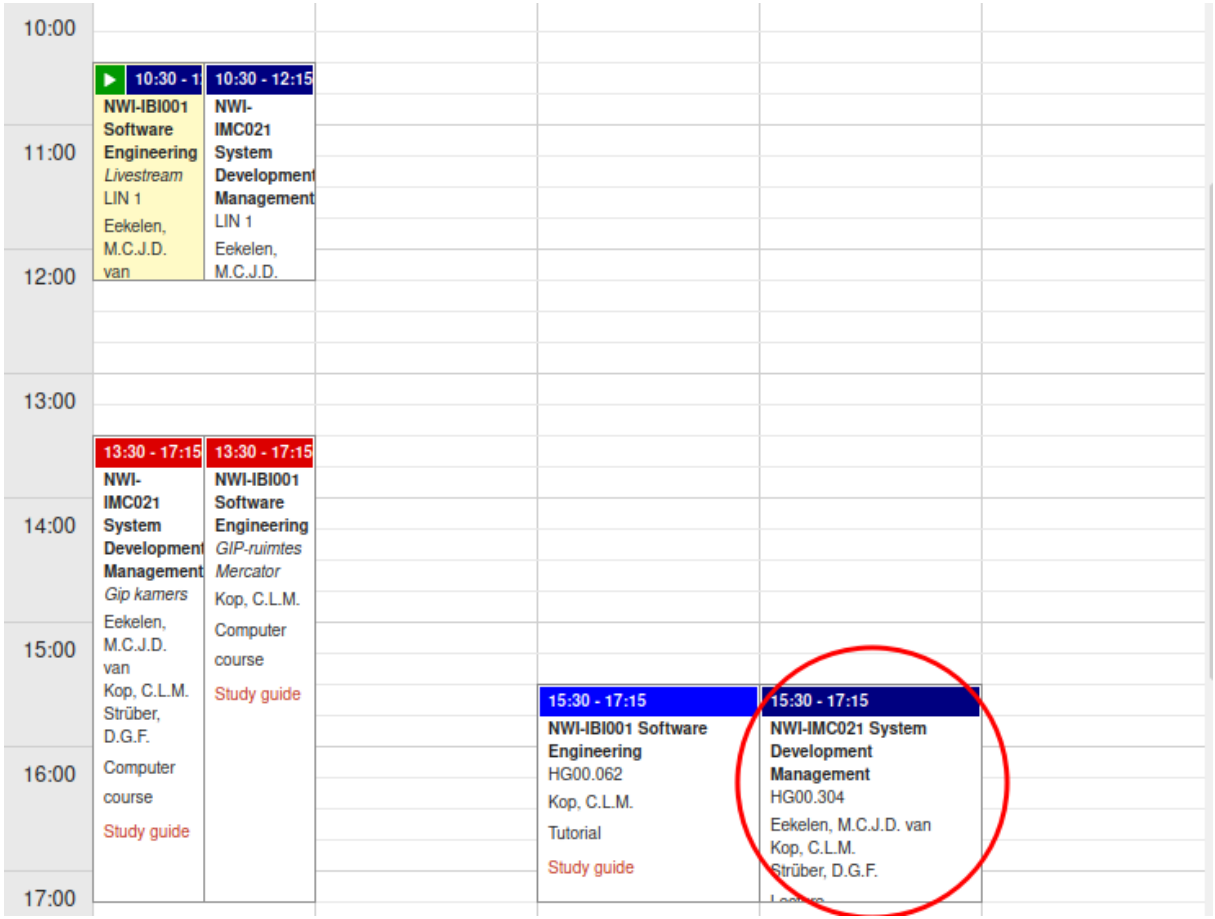
Wednesday afternoon: shared working time

Schedule



Wednesday afternoon: next week: optional Git lecture

Schedule



Thursday afternoon: SDM lecture

In your first two weeks

- 13:30–14:15 today: getting to know each other

In your first two weeks

- 13:30–14:15 today: getting to know each other
  - What are your skills, and how experienced are you?
  - What is your preferred role in a team?
  - Do you like to take the lead, or be told what to do?
  - How do you like to collaborate?
  - What would your preferred times for working together be?
  - ...

In your first two weeks

- 13:30–14:15 today: getting to know each other
  - What are your skills, and how experienced are you?
  - What is your preferred role in a team?
  - Do you like to take the lead, or be told what to do?
  - How do you like to collaborate?
  - What would your preferred times for working together be?
  - ...
- 14:30–15:15 today: getting to know your client

In your first two weeks

- 13:30–14:15 today: getting to know each other
  - What are your skills, and how experienced are you?
  - What is your preferred role in a team?
  - Do you like to take the lead, or be told what to do?
  - How do you like to collaborate?
  - What would your preferred times for working together be?
  - ...
- 14:30–15:15 today: getting to know your client
  - What is their vision for the project?
  - What are their key priorities? Are any parts "nice to have, but okay to omit"?
  - What could you get started on right now?
  - How do they like to be contacted?
  - When will your next appointment be to learn more details?

In your first two weeks

- 13:30–14:15 today: getting to know each other
  - What are your skills, and how experienced are you?
  - What is your preferred role in a team?
  - Do you like to take the lead, or be told what to do?
  - How do you like to collaborate?
  - What would your preferred times for working together be?
  - ...
- 14:30–15:15 today: getting to know your client
  - What is their vision for the project?
  - What are their key priorities? Are any parts "nice to have, but okay to omit"?
  - What could you get started on right now?
  - How do they like to be contacted?
  - When will your next appointment be to learn more details?
- 15:30–17:15 today: start preparations

In your first two weeks

- 13:30–14:15 today: getting to know each other
  - What are your skills, and how experienced are you?
  - What is your preferred role in a team?
  - Do you like to take the lead, or be told what to do?
  - How do you like to collaborate?
  - What would your preferred times for working together be?
  - ...
- 14:30–15:15 today: getting to know your client
  - What is their vision for the project?
  - What are their key priorities? Are any parts "nice to have, but okay to omit"?
  - What could you get started on right now?
  - How do they like to be contacted?
  - When will your next appointment be to learn more details?
- 15:30–17:15 today: start preparations
  - explore existing code, look into necessary frameworks
  - have discussions about task division, architecture
  - ...

In your first two weeks

- 15:30–17:15 Thursday (SDM): **online** lecture
  - how to manage your client?
  - what does a project manager do?
  - assignment

In your first two weeks

- 15:30–17:15 Thursday (SDM): **online** lecture
  - how to manage your client?
  - what does a project manager do?
  - assignment
- Rest of the week:
  - start preparations (reading existing code, learning the language, making a mockup design, making team agreements, SDM assignments, etc.)
  - familiarise yourself with Git (note: use the introductory lectures on Brightspace)

In your first two weeks

- 15:30–17:15 Thursday (SDM): **online** lecture
  - how to manage your client?
  - what does a project manager do?
  - assignment
- Rest of the week:
  - start preparations (reading existing code, learning the language, making a mockup design, making team agreements, SDM assignments, etc.)
  - familiarise yourself with Git (note: use the introductory lectures on Brightspace)
- Monday 10:30–12:15 next week: shared lecture
  - agile development process
  - Scrum

In your first two weeks

- 15:30–17:15 Thursday (SDM): **online** lecture
  - how to manage your client?
  - what does a project manager do?
  - assignment
- Rest of the week:
  - start preparations (reading existing code, learning the language, making a mockup design, making team agreements, SDM assignments, etc.)
  - familiarise yourself with Git (note: use the introductory lectures on Brightspace)
- Monday 10:30–12:15 next week: shared lecture
  - agile development process
  - Scrum
- Wednesday 15:30–17:15 next week: optional Git workshop
  - instruction in Git basics
  - covers the same things as the Brightspace lecture, but with help :)

Time spent on this course

Time spent on this course

- 6 ec = 168 hours

Time spent on this course

- 6 ec = 168 hours
- In theory: 9.3 hours per week (18 non-holiday weeks)

Time spent on this course

- 6 ec = 168 hours
- In theory: 9.3 hours per week (18 non-holiday weeks)
- However: generally **very little** progress in exam weeks

Time spent on this course

- 6 ec = 168 hours
- In theory: 9.3 hours per week (18 non-holiday weeks)
- However: generally **very little** progress in exam weeks (leading to a frantic pace in the last weeks. . .)

Time spent on this course

- 6 ec = 168 hours
- In theory: 9.3 hours per week (18 non-holiday weeks)
- However: generally **very little** progress in exam weeks (leading to a frantic pace in the last weeks. . .)
- Also: no shared working day after Carnaval, Easter and Pentecost

Time spent on this course

Q3 W1	SHARED	01 feb	02 feb	SDM	04 feb	11
Q3 W2	SHARED	08 feb	SHARED	SDM	11 feb	11
Q3 W3	SE	15 feb	16 feb	SDM	18 feb	11
Q3 W4	SE	22 feb	23 feb	SDM	25 feb	11
Q3 W5	HOLIDAY	HOLIDAY	02 mar	03 mar	PLAN	4
Q3 W6	SE	08 mar	09 mar	SDM	11 mar	11
Q3 W7	PRESENT	15 mar	16 mar	17 mar	18 mar	11
ELP	21 mar	22 mar	23 mar	24 mar	25 mar	2
EXAMS	28 mar	29 mar	30 mar	31 mar	01 apr	2
EXAMS	04 apr	05 apr	06 apr	07 apr	08 apr	2
Q4 W1	11 apr	12 apr	13 apr	14 apr	HOLIDAY	11
Q4 W2	HOLIDAY	19 apr	20 apr	21 apr	22 apr	4
Q4 W3	25 apr	26 apr	27 apr	28 apr	29 apr	11
HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	HOLIDAY	2
Q4 W4	9 may	10 may	11 may	12 may	13 may	11
Q4 W5	16 may	17 may	18 may	19 may	20 may	11
Q4 W6	23 may	24 may	25 may	HOLIDAY	HOLIDAY	11
Q4 W7	30 may	31 may	01 jun	02 jun	03 jun	11
Q4 W8	HOLIDAY	07 jun	08 jun	09 jun	DELIVER	4

Managing a team

Managing a team

- Each team has one or two managers.

Managing a team

- Each team has one or two managers.
- Managers may be reassigned.

Managing a team

- Each team has one or two managers.
- Managers may be reassigned.
- Managers should do all core tasks.
  (Don't just let one person manage the team and the other manage the client!)

Managing a team

- Each team has one or two managers.
- Managers may be reassigned.
- Managers should do all core tasks.
  (Don't just let one person manage the team and the other manage the client!)
- Note: Scrum teams are in principle self-managed. The manager is not the boss.

COVID situation

- While working online is possible, it is not recommended.

COVID situation

- While working online is possible, it is not recommended.
- **Recommended:** work in a group common room; focus on face-to-face communication

COVID situation

- While working online is possible, it is not recommended.
- **Recommended:** work in a group common room; focus on face-to-face communication
- **Also allowed:** work in a digital group common room; focus on digital face-to-face and voice communication

COVID situation

- While working online is possible, it is not recommended.
- **Recommended:** work in a group common room; focus on face-to-face communication
- **Also allowed:** work in a digital group common room; focus on digital face-to-face and voice communication
- **If you have to work online:** previous groups have indicated positive experience with Discord

Final notes

Final notes

- **Schedule:**
  - The schedule shows a common reserved time, but you are free to use any slot in the week.
  - Planning is available on Brightspace.

Final notes

- **Schedule:**
  - The schedule shows a common reserved time, but you are free to use any slot in the week.
  - Planning is available on Brightspace.
- **GiP rooms**
  - Three GiPHouse rooms on the fifth floor: M1.05.03, M1.05.04, M1.05.05.
  - One GiPHouse room on the first floor: M1. 00.13 (fits multiple groups)
  - Perhaps in the future: extra room on Mondays (see Brightspace).
  - Rooms can be entered using your keycard (once we set it up).
  - Rooms can be reserved at `https://giphouse.nl/reservations/`
  - Keep these rooms clean and use them for GiPHouse only.

Final notes

- **Communication lines**
  - For issues relating to your project, address your managers first, then the directors.
  - Directors can be reached at `directors@giphouse.nl`
  - For course-related issues, contact either the teaching assistants (the directors), or contact the teachers at: `teachers@giphouse.nl`
  - For technical issues, contact the directors.

Final Notes

- **Systems**
  - All code tracking and project tracking must be done through GitHub. We will provide repositories for everyone.
  - We use the tools from GitHub as a continuous integration testing environment.
  - Servers through Amazon.

Final Notes

- **Systems**
  - All code tracking and project tracking must be done through GitHub. We will provide repositories for everyone.
  - We use the tools from GitHub as a continuous integration testing environment.
  - Servers through Amazon.
- **Best practices**
  - Meet weekly to work on project together in scheduled hours.
  - If you are online, make sure everyone has a working camera and microphone.
  - Communicate openly and early with managers and directors.
  - Start working on quality code and tests early.

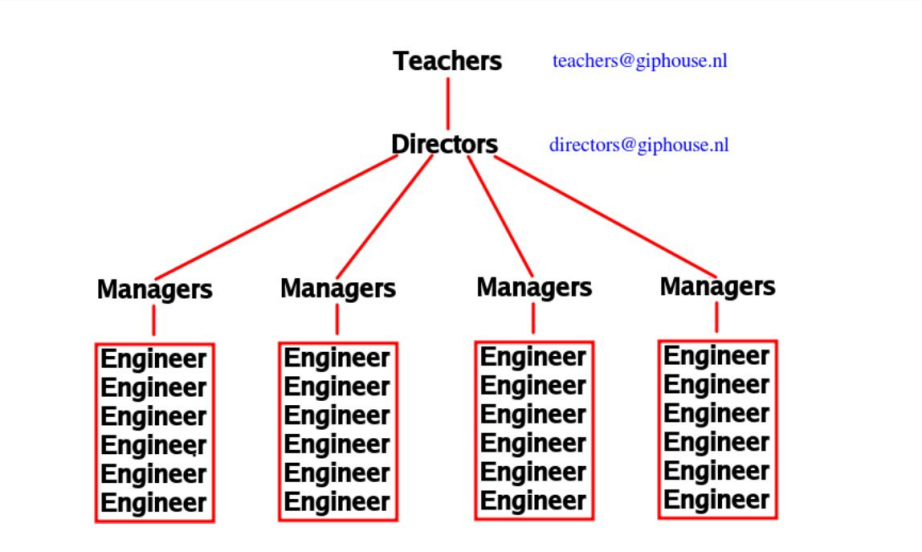
Fall term GiPHouse (master)

- Software Development Entrepreneurship
  - Work in a team to create your own start-up.
  - Create a product and make it successful!
    - business model canvas
    - getting out there
    - elevator pitch
    - working code
    - minimal viable product

Welcome to GiPHouse

Onboarding

Structure of GiPHouse



Directors:

- Anything about the projects itself
- Technical requests / problems

Systems

GitHub:

- You will be invited for a team and repository later today
- GitHub Actions for CI/CD
- GitHub Projects for planning (obligatory)

Amazon Web Services:

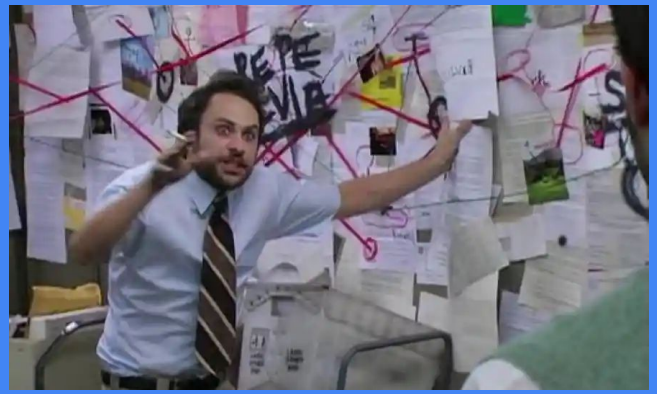
- EC2 instances for test-servers
- Request via CTO when you're ready for it (cto@giphouse.nl)

Best practices

- Meet weekly to work on project, together (preferably in the same room)
  - Additional individual development, depending on preferences
- Communicate openly and early with managers and directors.
- Be aware of cultural differences if applicable
  - More info during lecture next week
- Start working on documentation and tests early
- Shared responsibility towards end-product, self-organizing teams

The project teams

The moment everyone is waiting for…



SAT-solved based on your project and partner preferences and programming experience.

GEARS Voice Assistant

Meeting room: M1.5.05

Client meeting at: 14:30

Email:
2022spring-gears-va@giphouse.nl

Client: Edwin Hendriks
<edwin@xlrit.com>

- **Managers:**
  - Suzan Erven
- **Engineers:**
  - Andrei Octavian Dan
  - Dante van Gemert
  - Ege Sari
  - Elianne Heuer
  - Hans Lous
  - Lavika Singhal

Slides available at giphouse.nl after this lecture.

GoldenEye

Meeting room: M1.5.04

Client meeting at: 14:30

Email:
2022spring-goldeneye@giphouse.nl

Client: Rowan Goemans
<rowan.goemans@bdtnl.com>

- **Managers:**
  - Renato Feroce
  - Wesley Sip
- **Engineers:**
  - Faycel Harrathi
  - Lisanne Weidmann
  - Marten Straatsma
  - Pepijn van Grunsven
  - Quinn Ketelaars
  - Willem Medendorp

Slides available at giphouse.nl after this lecture.

HAN Kletsbot

Meeting room: HG02.028

Client meeting at: 14:30

Email:
2022spring-kletsbot@giphouse.nl

Client: Rene Bakker
<rene.bakker@han.nl>

- **Managers:**
  - Arthur Ramaekers
- **Engineers:**
  - Bayu Fransz
  - Hamzah Al Zubi
  - Jorn van Eldik
  - Sam Haeck
  - Stefan Vlastuin
  - Todd Bellavia

Slides available at giphouse.nl after this lecture.

RU Interactive Campus Map

Meeting room: EOS01.150

Client meeting at: 16:00

Email: 2022spring-campusmap@giphouse.nl

Client: Margot Jansen <margot.jansen@ru.nl>

Special remarks:

- Client meeting together with RU Meeting Walks.
- Client today is Ferry Soetekauw.
- Until client meeting, get to know each other somewhere around campus. (Room available from 15:30)

- **Managers:**
  - Evert van 't Oor
  - Isolde Hoogendonk
- **Engineers:**
  - Marie-Sophie Simon
  - Marijn van Wezel
  - Niels Boegman
  - Quinten Kock
  - Stefan Weijers
  - Twan Bolwerk

Slides available at giphouse.nl after this lecture.

Landolfio

Meeting room: M1.0.02 (“*Fish bowl*”)

Client meeting at: ~16:00

Email:
2022spring-landolfio@giphouse.nl

Client: Job Doesburg
<job.doesburg@gmail.com>

Special remarks:

- Client’s additional documents provided via email later today
- *Yes, this client is also manager and director*

- **Managers:**
  - Ylja van Son
- **Engineers:**
  - Jasper Hage
  - Joey van den Eijnden
  - Luuk Maas
  - Sander van Houtert
  - Stef Gijsberts
  - Tom Rust

Slides available at giphouse.nl after this lecture.

RU Meeting Walks App

Meeting room: EOS01.150

Client meeting at: 16:00

Email:
2022spring-meeting-walks@giphouse.nl

Client: Wilmien Heys
<wilmien.heys@radboudumc.nl>

Special remarks:

- Client meeting together with RU Interactive Campus App.
- Until client meeting, get to know each other somewhere around campus. (Room available from 15:30)

- **Managers:**
  - Jaimy Göertz
- **Engineers:**
  - Alperen Ozkurt
  - Jesper Somers
  - Niccolò Carrivale
  - Nick van Oers
  - Spyridon Alexandros Banos

Slides available at giphouse.nl after this lecture.

Mosadex E-health CoffeeChat

Meeting room: HG00.218a (LoS)

Client meeting at: 14:30

Email:
2022spring-coffeechat@giphouse.nl

Client: David Griffioen
<david.griffioen@mosadex-ehealth.nl>

- **Managers:**
  - Martijn Vogelaar
- **Engineers:**
  - Alex van der Hulst
  - Axel van Abkoude
  - Madelief Slaats
  - Nick Wiebe
  - Rick ten Tije
  - Tomas Woldu Semere

Slides available at giphouse.nl after this lecture.

NS Engineer app

Meeting room: HG00.506

Client meeting at: 14:30

Email: 2022spring-ns-engineer-app@giphouse.nl

Client: Peter Blom <peter.blom@ns.nl>

- **Managers:**
  - Ege Saraydar
  - Roy van der Steen
- **Engineers:**
  - Esther Shi
  - Jonathan Karels
  - Jorrit de Boer
  - Martan van der Straaten
  - Nadezhda Dobreva
  - Sijmen van Bommel

Slides available at giphouse.nl after this lecture.

Optimal Scans 1

Meeting room: M1.0.03 (left)

Client meeting at: 14:30

Email: 2022spring-optimal-scans-1@giphouse.nl

Client: R. de Nooij <r.denooij@optimalplanet.nl>

Special remarks:

- Client meeting together with Optimal Scans 2

- **Managers:**
  - Yorick Last
- **Engineers:**
  - Artur Wiadrowski
  - Luc Schrauwen
  - Thomas Berghuis
  - Vamsi Yerramsetti
  - Wout van den Berg

Slides available at giphouse.nl after this lecture.

Optimal Scans 2

Meeting room: M1.0.03 (right)

Client meeting at: 14:30

Email: 2022spring-optimal-scans-2@giphouse.nl

Client: R. de Nooij <r.denooij@optimalplanet.nl>

Special remarks:

- Client meeting together with Optimal Scans 1

- **Managers:**
  - Yentl van Dop
- **Engineers:**
  - Bart Veldman
  - Cas Visser
  - Dirk Doesburg
  - Leon Zhang
  - Simcha van Collem

Slides available at giphouse.nl after this lecture.

RU HEN Dashboard

Meeting room: HG00.075

Client meeting at: 14:30

Email: 2022spring-ru-hen-dashboard@giphouse.nl

Client: Tom van Onna <tom.vanonna@ru.nl>

- **Managers:**
  - Stef Thijssen
  - Thomas Klein Breteler
- **Engineers:**
  - Elina Antonova
  - Ilse Slootweg
  - Johan Urban
  - Julius Benjamins
  - Siem Kleuskens
  - Tim Jansen

Slides available at giphouse.nl after this lecture.

Tapando

Meeting room: M1.5.03

Client meeting at: 14:30

Email:
2022spring-tapando@giphouse.nl

Client: Stef van Hulst
<stefvhulst@hotmail.com>

- **Managers:**
  - Benjamin Gerritsen
- **Engineers:**
  - Daan de Grauw
  - Harm Roukema
  - Leah Heil
  - Thomas Luijkman
  - Thomas Broekman

Slides available at giphouse.nl after this lecture.

Thirona Performance Dashboard

Meeting room: HG00.217a (LoS)

Client meeting at: 14:30

Email: 2022spring-thirona-performance-dashboard@giphouse.nl

Client: Annet Meijers <annetmeijers@thirona.eu>, Sander Hendrix <sanderhendrix@thirona.eu>

Special remarks:

- Zoom conference with client. Please send them a meeting link.
- The room has a video setup.
- Several in this team can only work online.

- **Managers:**
  - Emma Schipper
- **Engineers:**
  - Casper Fabritius
  - Gaby Dingena
  - George Dan
  - Jiaben Zhao
  - Ties Speel

Slides available at giphouse.nl after this lecture.

RU Weblectures

Meeting room: M1.0.12 (NDL)

Client meeting at: 14:30

Email:
2022spring-weblectures@giphouse.nl

Client: Peter van der Wolk <peter.vanderwolk@ru.nl>, Ton Toonen <ton.toonen@ru.nl>

- **Managers:**
  - Job Doesburg
- **Engineers:**
  - Ana Cullen
  - David van der Woude
  - Harm Jacobs
  - Lars Abbink
  - Lucas van Kasteren
  - Thomas Rhemrev

Slides available at giphouse.nl after this lecture.

Yoast Site Assessment

Meeting room: HG00.218 (LoS)

Client meeting at: 14:30

Email: 2022spring-yoast-site-assessment@giphouse.nl

Client: Irene Strikkers <irene@yoast.com>

- **Managers:**
  - Tuan-Anh Nguyen
- **Engineers:**
  - Niek Terol
  - Niels Feij
  - Seraph Jin
  - Sven van der Post
  - Thomas De Haan

Slides available at giphouse.nl after this lecture.

Zorgdoc SOS service

Meeting room: HG00.217 (LoS)

Client meeting at: 14:30

Email:
2022spring-zorgdoc-sos@giphouse.nl

Client: Sven Berkvens-Matthijsse <sven@zorgdoc.nl>

- **Managers:**
  - Gianni Monteban
- **Engineers:**
  - Carlota Julià
  - Jorn Heibrink
  - Luke van Leijenhorst
  - Stefan van Ieperen
  - Tim Gerritsen
  - Tim van Alten

Slides available at giphouse.nl after this lecture.

What’s next (1)

- Get in contact with your teammates
  - Also those not physically present!
  - Radboud University Zoom license (register with your @ru.nl adres and you’re automatically set up with the correct license)
  - Some rooms have a teleconferencing system (USB plug-and-play)
- Get to know each other
- Decide on a weekly schedule (physical / online / hybrid, individual / group)

What’s next (2)

- Get to know your client (first meeting)
- Get to know the project details
- **Read documentation, get your prerequisites up-to-date**
- Learn Git (attend the workshop if this is new to you)
- Schedule on BrightSpace (“Course Schedule”)

COVID-wise

- This course is considered a ‘practical course’ by the university
- If you are uncomfortable with meeting physically, this should not be a problem! (even though it could make effective collaboration more difficult)
- If a fellow team member is uncomfortable with meeting physically, this should not be a problem!
- Make the best out of it, find practical solutions, just as in ‘real’ companies

- Of course, adhere to the university COVID guidelines (masks, self-tests, etc…)

Reserving rooms

- Via giphouse.nl/reservations:
  - 3 rooms on Mercator 1, 5th floor (M1.5.0{3,4,5} “*Copenhagen*”, “*Vienna*”, “*Madrid*”)
  - 1 room for 2 teams on Mercator 1, ground floor (M1.0.13 “*New York*”)
  - Be collegial to other teams
  - Access with your university cards (soon™)
- Other rooms: zaalreservering@science.ru.nl
- https://face.ru.nl
- Mercator 1 (*M1.0.02 “fish bowl”*) via whiteboard
  - Be collegial to other people that need to use that room

Questions

- First discuss with manager
- Email to directors@giphouse.nl
- Especially today, we try to answer quickly.

- Today: Come by at M1.1.13 (only if email does not work, let's not make it too crowded)
  - If you do not have a team, definitely come by

Switching groups

- Concrete proposal of 2 or 3 people switching groups
- Needs to be approved by directors
- Come by before 12:30

- Our proposal: stay around here if you want to try switching teams (and/or check the Brightspace chat)