

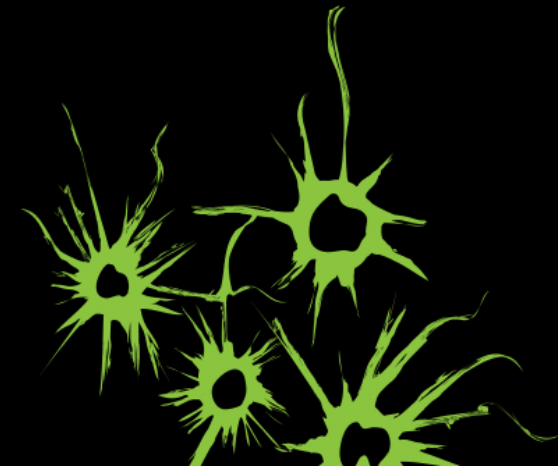
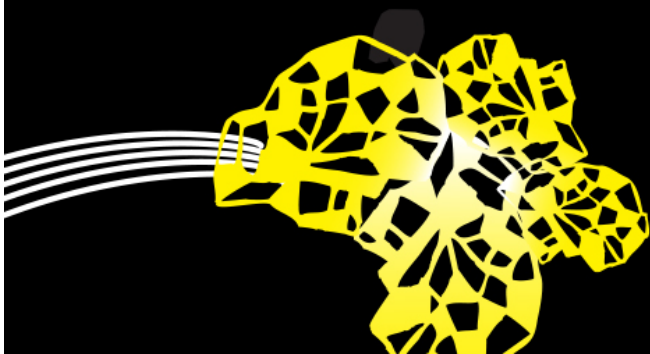
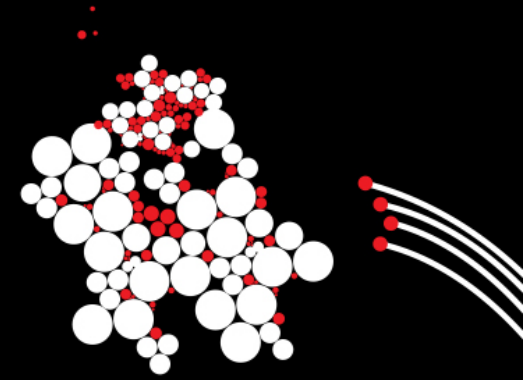
UNIVERSITY OF TWENTE.

P6.2: PROJECT KICK-OFF

LUÍS FERREIRA PIRES

201700117-1B MODULE 2: SOFTWARE SYSTEMS

17 DECEMBER 2019





PROGRAMMING LINE OVERVIEW

Week 1 Values and variables Control flow	Week 2 Classes and objects Testing	Week 3 Interfaces and Inheritance Subtyping Security 1
Week 4 Arrays and Lists List implementations Collections	Week 5 Stream I/O and MVC Exceptions Security 2	Week 6 Concurrency Project kick-off IDE Tips & Tricks
Week 7 Basic Networking Networking and Multithreading GUIs	Week 8/9 Advanced Java facilities Test	Week 10 Project Test resit



CONTENTS

PROJECT KICK-OFF

- What is the project about?
- Project activities
- Project planning
- Tips





ABALONE



- Board game for 2 to 4 players
- Objective is to push 6 marbles of an opponent out of the board
- Game description is available on Canvas
- Play over a network (client/server)
- Computer player

ORGANISATION

- Work **with your lab partner** (and **collaborate!**)
- Problems between partners:
 - Keep the housekeeper or group TA informed
 - Rather earlier than later
- Early-bird deadline (bonus): **Wednesday Week 10**
- Final deadline: **Friday Week 10**
- Minimal requirements in the **manual 'Project descriptions' chapter**

Revised version
(16 December 2019)



PROJECT ACTIVITIES

- **Project discussion meeting with TA**
 - Wednesday week 6, with house
- **Discuss project planning with TA**
 - Wednesday week 7, during lab session
 - Sign-off exercise P-7.1
- **Protocol session**
 - Wednesday week 7, with lab group
 - Feedback session: Wednesday week 8

PROJECT ACTIVITIES

- **Peer feedback sessions** (in pairs of project pairs)
 - Tuesday Week 8
 - Tuesday Week 9
 - Tuesday Week 10
- **Tournament**
 - Wednesday, week 10

IDEAL PLANNING

FOR BEING READY BEFORE THE DEADLINE

- Week 6: Game logic
- Week 7: TUI & Protocol
- Week 8: Testing
- Week 9: Report
- Week 10: Extras



GRADING

CRITERIA DEFINED IN THE MANUAL

- **Packaging**
 - Complete submissions that fulfil the requirements (see manual page 21)
- **Global design**
 - Overall design, class specifications
- **Programming**
 - Code quality, functional requirements, game rules, documentation

GRADING (CONT.)

CRITERIA DEFINED IN THE MANUAL

- **Testing**
 - Unit tests, system tests
- **Reflection on planning**
- **Report**
 - Structure, writing (readability)
- **Extensions**
 - Optional (for bonus)

BONUS RULES

- **Protocol maintenance**
 - 0.5 points (pair or single student, one per tutorial group)
- **Tournament:** computer players play against each other
 - First place: 1.0 point
 - Second place: 0.5 point
- **Extensions**
 - Chatbox, challenge, leaderboard, security, GUI, etc. (see manual)
 - Only graded **if minimal requirements are fulfilled**

HINTS

- Read the **project description** and the **game rules**
- **Steps**
 1. Identify fundamental **sub-systems**
 2. Identify **main objects**
 3. **Pseudo-code**
 4. **Implement**
- Follow the **Model-View-Controller pattern** (Week 5)



HINTS

CONTINUATION

- GUI is allowed, but not necessary
- **Internals** are most important
- **Document and testing are mandatory!**
- Write report for the **intended audience**
→ Someone who is **supposed to maintain your system!**
- Take advantage of **peer feedback**





TAKE HOME MESSAGES



- Apply **what you have learned**
- **Work together** with your project partner
- Start **as soon as possible**
- Keep up with your planning (avoid procrastination!)

Have **fun** with the project!