Information on final project

Final Project

- Non-trivial project with a programming aspect
- Topic (in context of course) freely selectable
- Scope: large enough to justify clear, distincts roles for team members
  - Every team member should have a main area of responsibility
  - Every team member should have an overview of what other team members are doing, and know details of their own area
- But not expected to be a market-ready software product
- Project report: 4-6 pages (as orientation, not strict)

- **Submission of report and code:** December 16

Timeline

- **Today:** project information session
  - **Assignment:** proposal for your project
- **November 18:** pitch session for proposal
  - We will discuss your proposals, "greenlighting"
- **December 2**: submission of halfway progress report
- **December 16**: project submission (code + report) and project presentations

Topic suggestion I

- Develop your own product line, including a domain analysis and creation of various artifacts
  - Either top-down: analysis, design, implementation
  - Or refactoring of existing application towards a software product line

- Inspiration: examples from previous years

Example 1: Chess

Chess comes in many variants. Variations include different board types, different information about the entire status of the game, different capturing rules, different starting positions, etc[1]. For our software product line, we implemented a chess game where players can play against each other on one computer, and can select different features at compile time using FeatureIDE.

The wide range of possible variants and features lead to an interesting project with several possible feature interactions. This, in combination with creating a fun game, led to our decision to use chess as our software product line.

Figure 2: Gliński's hexagonal chess board and starting positions.

Example 2: Steering behavior simulator

For this project we decided to build a 2D Simulator of steering behaviours. Steering behaviours are a technique in game development to create realistic movement. Take for example the flocking steering behaviour: this behaviour tries to mimic the way a flock of birds or a school of fish move in unison. This behaviour uses three other steering behaviours: separation, cohesion, and alignment. Separation causes entities to move away from other entities when within a certain radius, while cohesion does the opposite and pulls entities together. Lastly, alignment causes entities to move in the same direction. The simulator is able to combine a multitude of these kinds of steering behaviours to create dynamic and interesting interactions between entities.
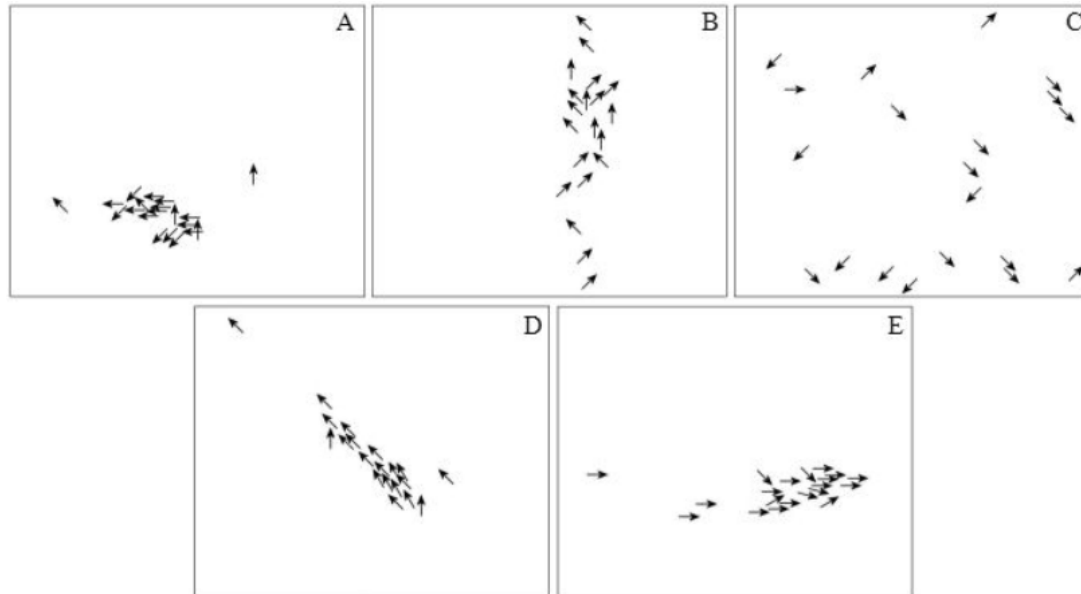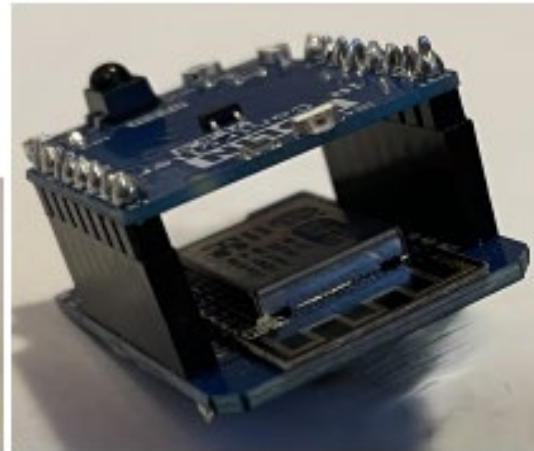
Example 3: Microcontroller service

The product we created is a piece of software that runs on a Wemos D1 mini micro controller that is equipped with an infrared shield. The Wemos D1 mini is an ESP8266 with an added UART chip for USB connection and some GPIO pins broken out. The device has WiFi and infrared capabilities, which allows it to be used for many purposes. One example is a REST API running on the device which allows replaying infrared codes. Another one would be monitoring incoming infrared signals and calling web hooks for special signals. The versatility of the device allows for many different features, but also for combinations of multiple features. This makes it a perfect application of product line technology. The idea is that the device can be integrated into existing systems and sold by different companies to fulfil various customer needs. It is not meant to be sold to end customers directly as depending on the selected features, various services, like a mobile app or some external server need to be provided to allow for full functionality. In order to make integration into existing systems easy we provide a collection of interfaces for the network, but also for the infrared aspects of the device.

(a) Hardware components   (b) Complete device

Figure 1: The Wemos D1 Mini

Topic suggestion II

- Comparison of two approaches or languages for product line development in a specific example
  - Aspects vs. Features
  - Preprocessors vs. Features
  - AHEAD vs. FeatureHouse

Topic suggestion III

- Repository mining study
  - Extract a large number of relevant projects from e.g. GitHub
  - Perform measurements to answer some research question in the scope of the lecture, e.g.:
    - **Feature interactions**: How frequently occuring, how many features involved?

Grading criteria

- Process quality (33.3%):
  - Was a systematic methodology followed (either from software engineering or research perspective)?

- Product quality (33.3%):
  - Are the resulting artifacts of high quality? External and internal quality: e.g., generated products are useful, non-trivial to build; code quality

- Report quality (33.3%):
  - Is the final report well-written, appropriately structured, makes justified conclusions based on concrete evidence?

Grading criteria

- Grades are generally group grades
- Individual adjustments possible
  - If clearly visible that contribution differs from average
  - In controversial cases: follow-up meeting

Final Report

- 4-6 pages
- Typical structure: context/background, motivation, project idea, methodology, results
- Explain rationale (e.g., *Why is an SPL methodology useful in this project? Which value does it add?*)
- Explain efforts for quality assurance (e.g., *How did you establish maintainability? To what extent did you test?*)
- Make area of responsibility of each group member explicit
- Submit by: December 16

Halfway progress report

- 2-3 pages
- Follow same structure as final report
- Text reuse allowed
  - You can reuse content from proposal in halfway report
  - You can reuse content from halfway report in final report
- Submit by: December 2

Final presentation

- Per group:
  - 10 minutes presentation + 5 minutes questions
- Should explain context/background, motivation, methodology, results
- Should include demo

Feedback and other discussion

- Assignment 9: prepare a proposal and pitch
  - to present and discuss in next week's session
- After that, contact me (d.strueber@cs.ru.nl) if
  - you are interested in preliminary feedback,
  - you are stuck, or
  - there are some other issues that need discussion.
- We can plan a follow-up meeting as required
  - typically on the regular Monday slots, but other times possible as well