Radboud University

Dr. Daniel Strüber

Radboud University
Institute for Computing
and Information Sciences

Autumn term 2020/21
Published: 07.09.2020
To be submitted:
14.09.2020 13:30

Software Product Lines

Assignment 2

Tasks 1-4 are designed as live exercises for the tutorial session – for this, we will split up in breakup groups. It's not required to submit anything for these tasks.

Tasks 5 and 6 are to be completed offline and to be submitted until the deadline.

Task 1: FeatureIDE tutorial

Familiarize yourself with feature modeling in FeatureIDE: Follow the tutorial from https://github.com/FeatureIDE/FeatureIDE/wiki/Tutorial

Follow the instructions from sections "Introduction" and "Create Feature Model" -- that is, stop before the beginning of the section "Implementing features".

Task 2: Chat product line

Imagine that you're leading a company that develops chat systems for company intranets and marketing websites (and imagine that there actually exists a market for such a company 😉).

a) Perform a domain analysis: What kind of features could the customer companies desire? For which of these features might a market exist? Which features might set your company apart from other companies that develop chats?

b) What benefits would your company gain from using product line technology? What alternative solutions may exist to using product line technology?

c) Create a feature model for this domain using FeatureIDE. First, create the feature hierarchy. Are there any further dependencies? If yes, model them.

d) Represent the feature model as propositional formula.

e) Specify a few of valid configurations and a few invalid configurations of the feature model.

f) How many valid configurations exist?

Task 3: Webmail product line

Follow the same steps a)-c), but for a software product line of webmail applications.

Task 4: Other product lines

Which other existing software product lines do you know? In which other software domains would product line technology be useful?

Discuss: Is each of the following software systems a software product line?
Linux, Windows, Word, Eclipse

Task 5: Feature models (Discussion)

a) Discuss: Why are feature models usually represented as tree, and not as a list/graph/expression/prolog program?

b) The same product line can be represented with different feature models, which, however, lead to exactly the same feature selections and products. Give an example and discuss possibilities for normalization.

Please submit your solution to this task as a PDF document.

Task 6: Chat implementation

Implement a *simple* chat application with Java. The program consists of multiple clients that can connect to the server and exchange messages over the server.

The chat application must have the following features.

1. **Color**: Each message can have a text color. In the client, it's possible to configure the text color for all outgoing messages.
2. **Authentication**: The client has to send a special message to the server, including a password, before it can send and receive messages.
3. **Two encryption algorithms**: All message exchanges are encrypted. To each message, two encryption algorithms are applied (the second one to the result of the first one).
4. **Log**: All clients and the server have a log, recording all received messages.

The features can be implemented in a minimal way (for example, encryption by ROT13 or reverting the message; color only shown as additional text; authentication with a fixed password; log writes to a file).

Tutorials and basic implementations of chats using sockets are available on the internet, for example, at https://www.ashishmyles.com/tutorials/tcpchat/index.html

Hint: We will use this example as a basis for upcoming assignments, in which we consider certain language extensions of Java. Since some of these extensions only support the core Java features, please avoid advanced Java concepts from newer Java versions; specifically, generics and lambdas (I know, might be awkward).

Please submit your solution to this task as a Zip file containing the Java code. If you're using Eclipse (which I recommend), use Eclipse's export functionality *(File -> Export -> Archive File).*