

ADVANCED SOFTWARE DESIGN

LECTURE I

INTRODUCTION

Kiko Fernandez-Reyes

CONTENTS

- Practical Matters
- Course overview – topics, text book
- Project
- Feedback-driven approach to learning
- Forming groups

PRACTICAL MATTERS

ARE YOU REGISTERED?

Admitted students must web-register themselves for this course in the Student Portal.

Students who are unable to web-register (or who were registered before) should contact the Student Office (it-kansli@it.uu.se).

Do this as soon as possible!

EXPECTATIONS

Take the course if you know...

- **Object-Oriented programming**
- **Coding** in an OOP language

Think about switching to another course if you...

- already know UML and design patterns or
- are **not familiar with OOP** or
- have **no experience programming**

POLL

EMAIL ADDRESS

If your email is broken — mailbox full, etc — you will miss crucial information.

Failing to exist is not an excuse for failure.

CONTENTS

- Practical Matters
- **Course overview – topics, text book**
- Feedback-driven approach to learning
- Project
- Forming groups

COURSE OVERVIEW

LECTURE SCHEDULE

Week 1

- 1. Domain modelling**
- 2. Structural modelling**

Week 2

- 3. Dynamic modelling**
- 4. GRASP principles**
- 5. Seminar**

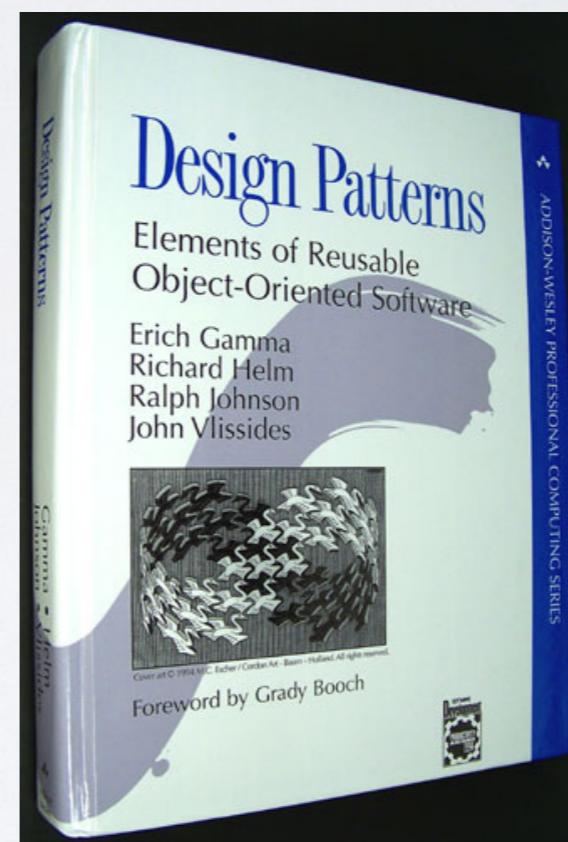
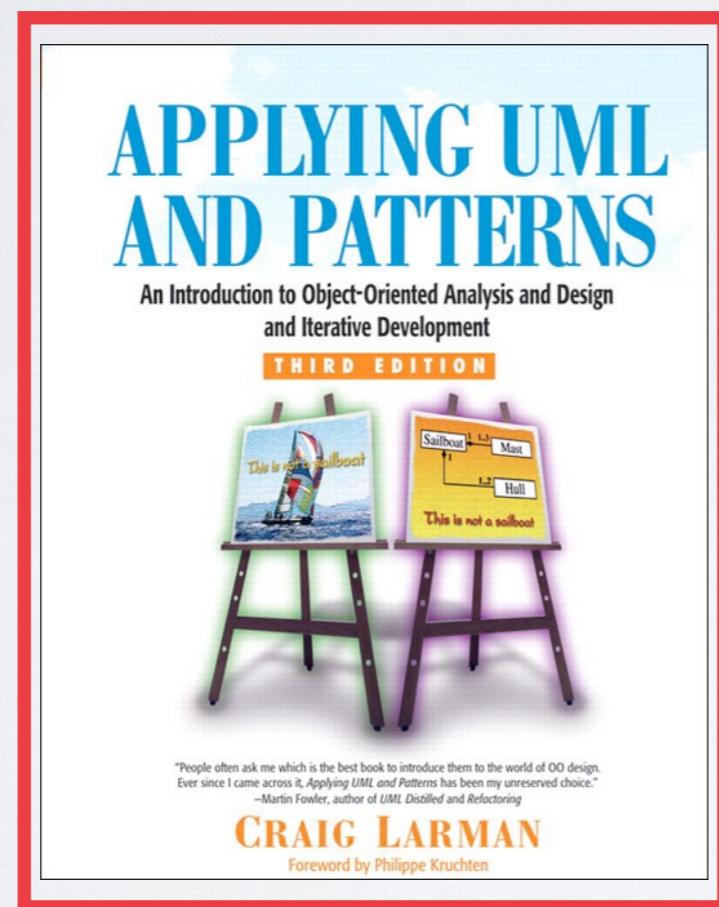
Week 3

- 6. Software Architecture**
- 7. Design Patterns I**
- 8. Design Patterns 2**

Week 4

- 9. Seminar**
- 10. Seminar**

TEXTBOOK AND REFERENCES

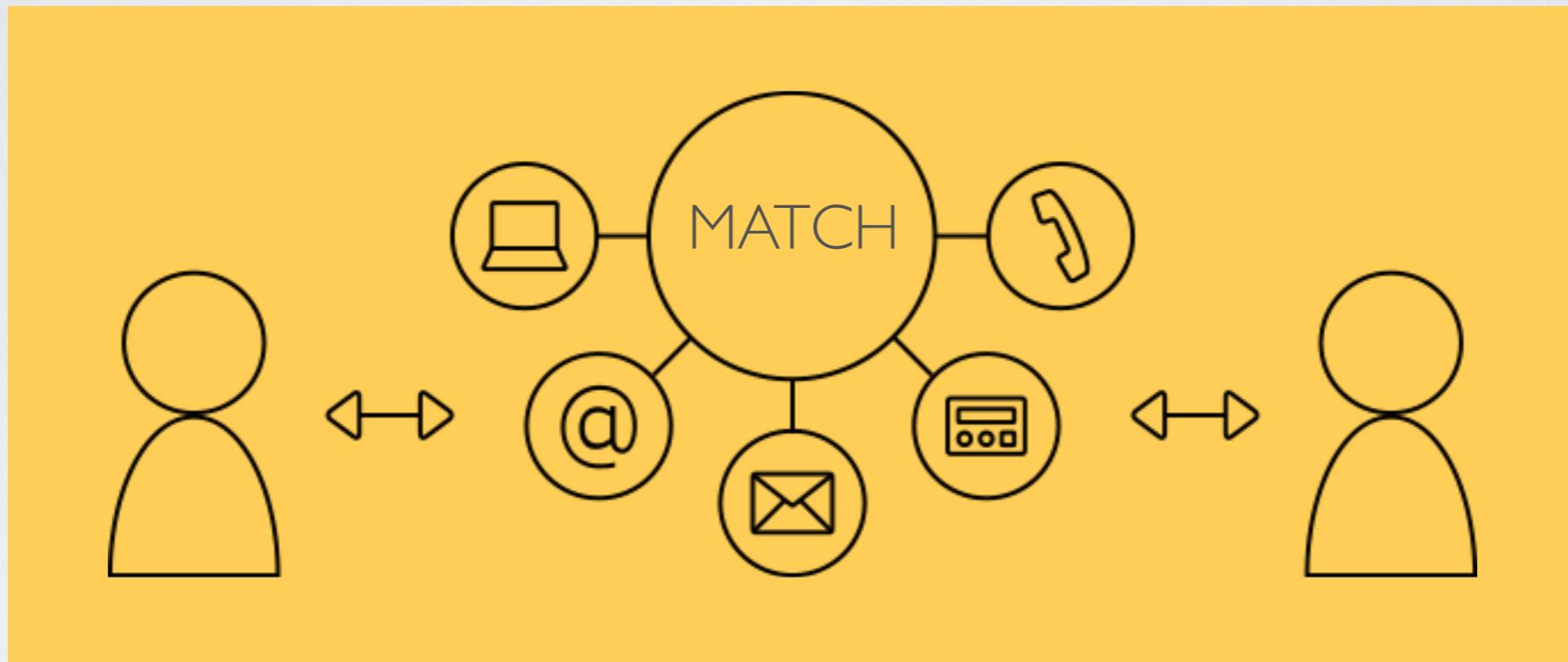


CONTENTS

- Practical Matters
- Course overview – topics, text book
- **Project**
- Feedback-driven approach to learning
- Forming groups

PROJECT

THE MATCH SYSTEM



Connecting people

SENIOR CARE APP



SENIOR CARE APP



SENIOR CARE APP



SENIOR CARE APP



MatchCare
(service implemented on
top of Match System)



SENIOR CARE APP



MatchCare
(service implemented on
top of Match System)



SENIOR CARE APP



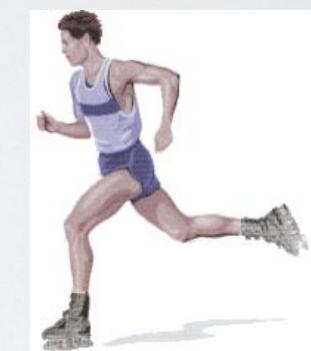
MatchCare
(service implemented on
top of Match System)



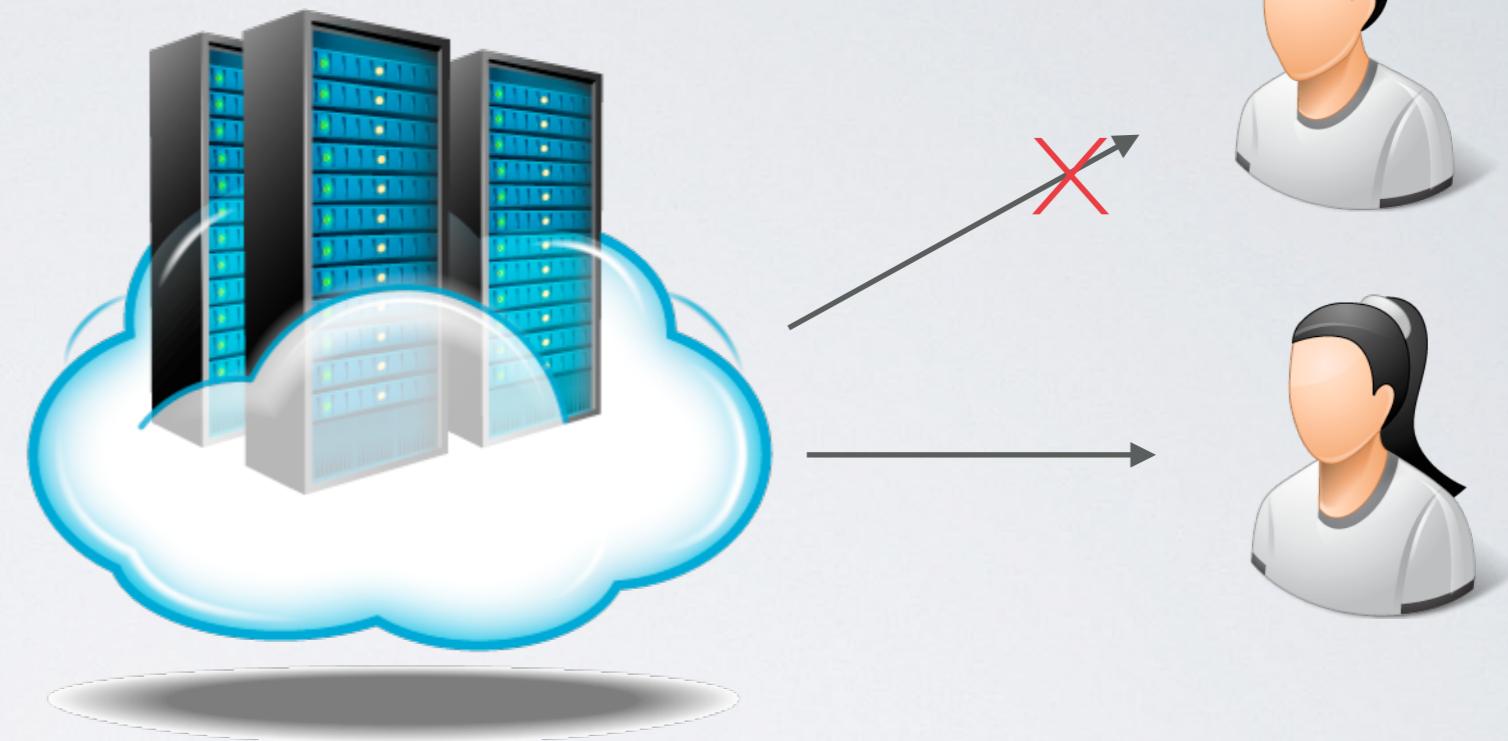
SENIOR CARE APP



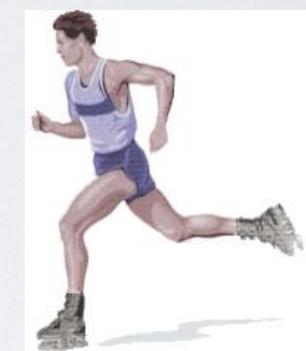
MatchCare
(service implemented on
top of Match System)



SENIOR CARE APP



MatchCare
(service implemented on
top of Match System)



SENIOR CARE APP



(no reply in 30

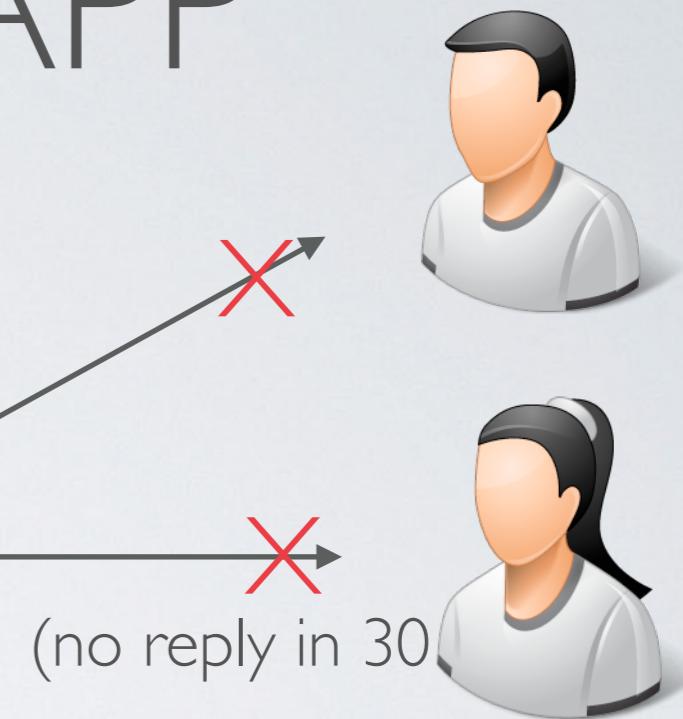
MatchCare
(service implemented on
top of Match System)



SENIOR CARE APP



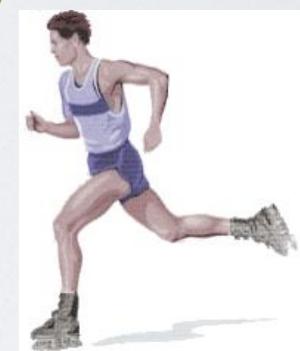
MatchCare
(service implemented on
top of Match System)



SENIOR CARE APP



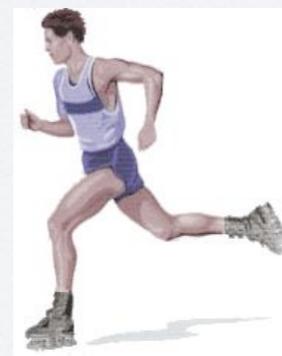
MatchCare
(service implemented on
top of Match System)



SENIOR CARE APP



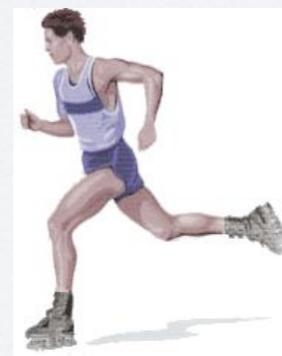
MatchCare
(service implemented on
top of Match System)



SENIOR CARE APP



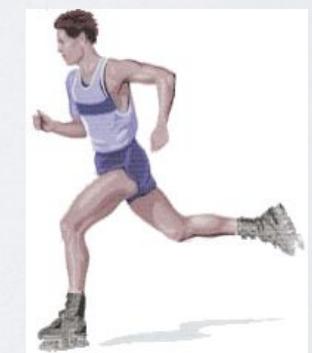
MatchCare
(service implemented on
top of Match System)



SENIOR CARE APP



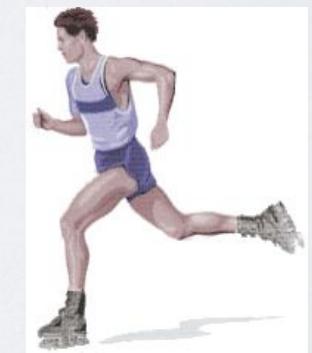
MatchCare
(service implemented on
top of Match System)



SENIOR CARE APP



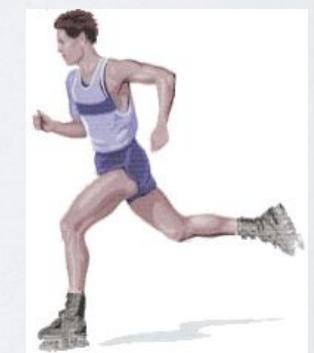
MatchCare
(service implemented on top of Match System)



SENIOR CARE APP



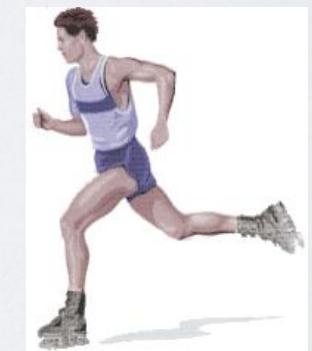
MatchCare
(service implemented on top of Match System)



SENIOR CARE APP



MatchCare
(service implemented on
top of Match System)



OTHER IMPLEMENTATIONS

- Odd Jobs
- Job Hunting
- Workforce planning
- Sports

Descriptions for these examples
are at the end of the
Match System configuration
(requirements.pdf)

PROJECT

Design and **partial implementation** of MATCH system

Groups of 4

Design core program plus some **extensions**.

Implement part of it

The design is more important than the implementation.

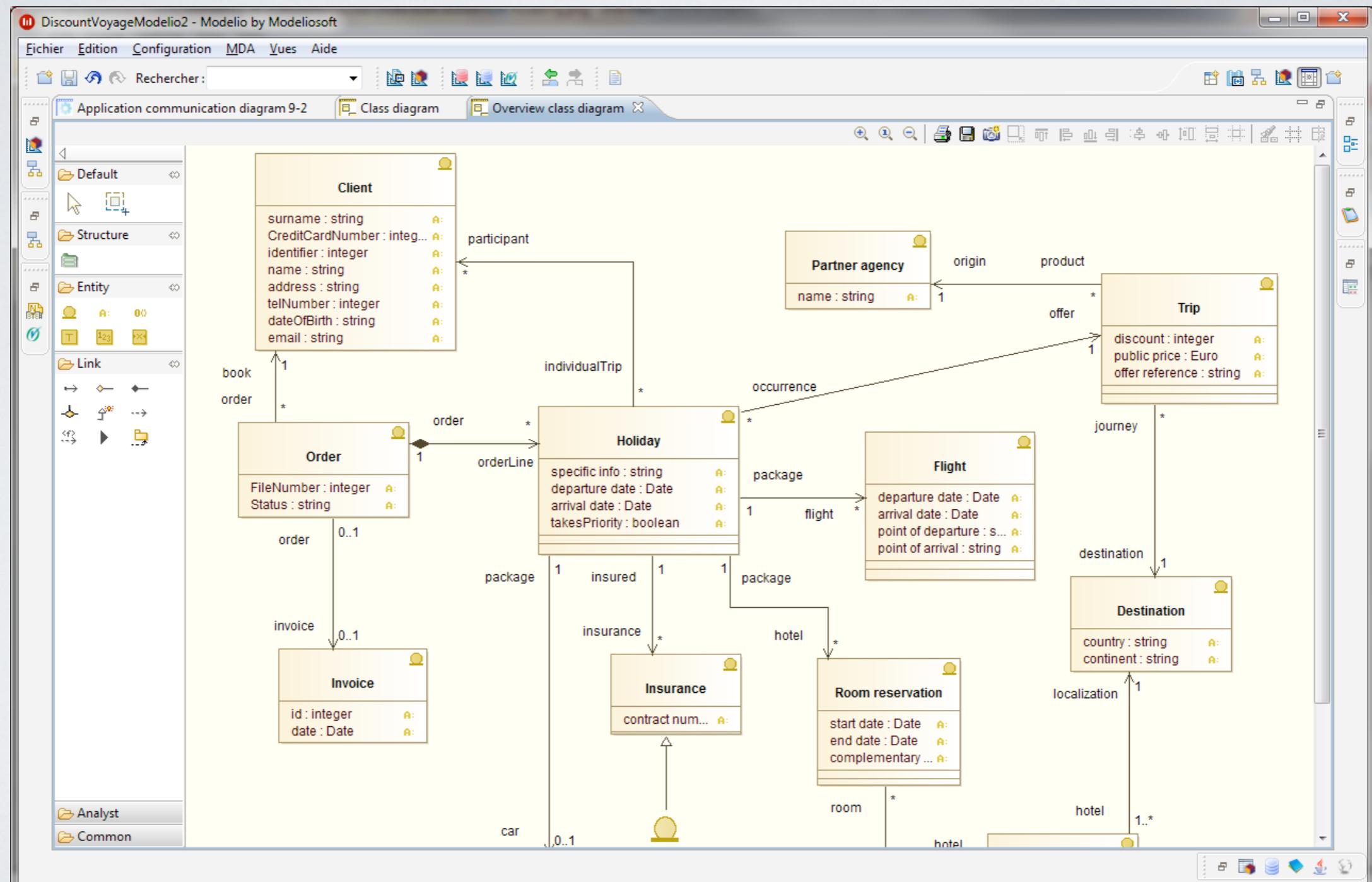
If you are stuck, try doing part of the implementation

PROJECT

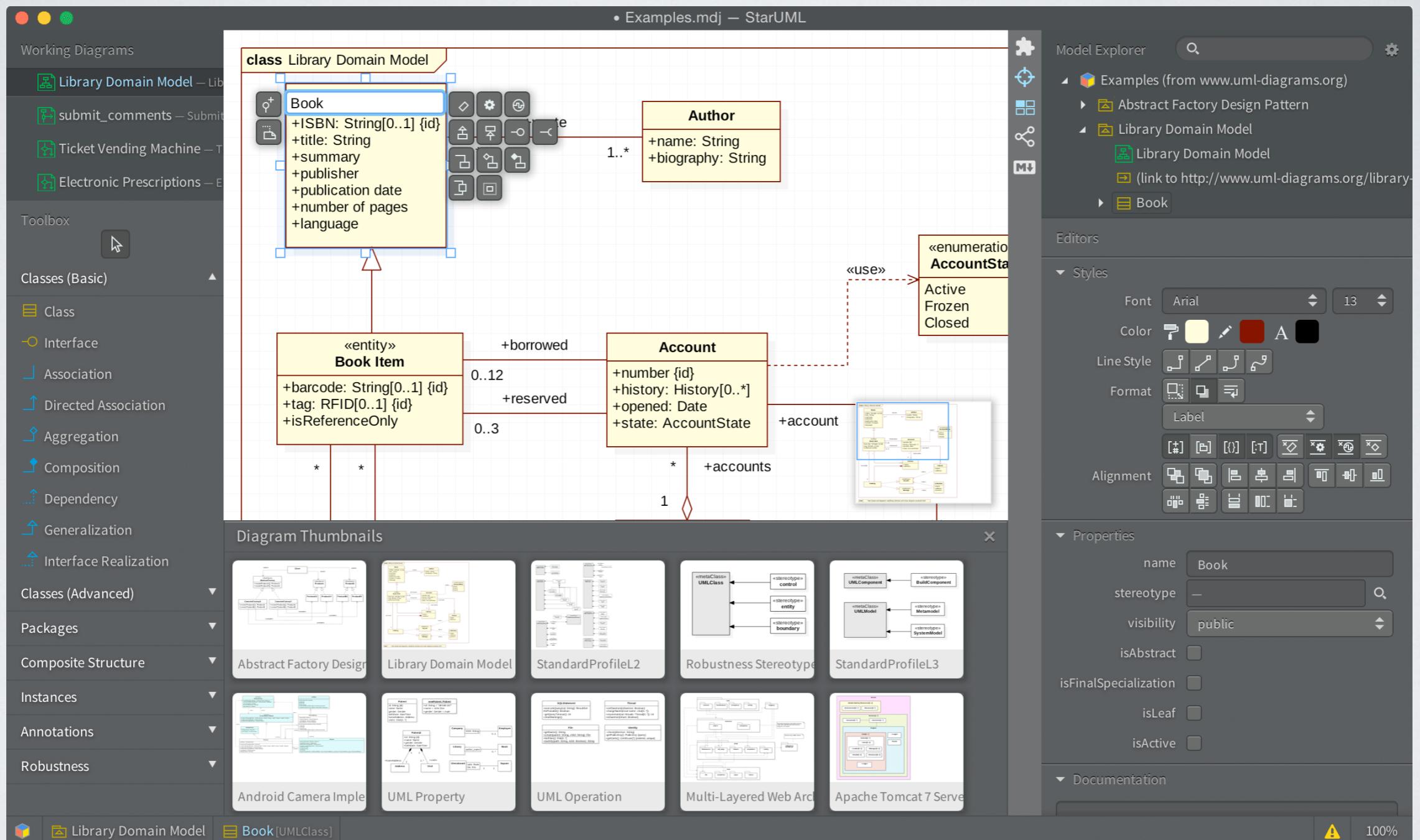
Volume produced is not what will be measured.

Rather, understanding and application of design processes and design principles, response to feedback, improvement of design, evaluation of others' design.

TOOLS: MODELIO



TOOLS: STARUML



CONTENTS

- Practical Matters
- Course overview – topics, text book
- Project
- **Feedback-driven approach to learning**
- Forming groups

FEEDBACK-DRIVEN LEARNING

FEEDBACK-DRIVEN LEARNING

A new assessment approach based on a fine-grained breakdown of learning outcomes — called *achievements*.

Detailed list of learning achievements known in advance.

Students need to **work towards receiving feedback for the achievements**.

Different achievements required for 3, 4, and 5 – higher grades require deeper levels of mastery, not more stuff.

C Domain Modelling

	Goal	Level	ILO
C1	Construct a valid domain model of the core elements of the software under design.	3	ILO-C

Criteria. Answer includes the most important domain elements and most important relationships between them, with few spurious elements and relationships.

Documents. Domain model.

THE PROCESS

You need to **have a plan**

TA will give you **feedback** during the weekly meeting

Achievements are *individual*

- when discussing with a TA need **all members of team** to participate
- **if you miss many meetings it will impact your grade**

THE PROCESS

At the end of the course, produce a report that clearly:

1. Links achievement(s) to sections/artefacts
2. Explains how it has been applied, and
3. Do not leave the reader to guess how something works or you may lose that achievement.

Final grade:

Individual participation + (Peer-review + Report)

THE PROCESS

Examination results from previous year

Number of students examined: 76

Fail: 15 (19.7%)

Pass (3): 28 (36.8%)

Pass with Distinction (4): 20 (26.3%)

Pass with Honours (5): 13 (17.1%)

MEETINGS WITH TA

One meeting every other week with TAs:

- **discuss issues** that arise in lectures
- discuss project, **obtain feedback**

Raphaela



Kiko



Albert



EXPECTATIONS

- Don't leave everything until the end
- Do research to **find out how to do things** – teach your team!
- Keep your group **alive** and **active**
- **Structure your work** towards getting feedback on as many achievements as possible
- **Show multiple achievements** per meeting.

PANIC?

Achievements are essentially the things you need to learn – hence *nothing to be afraid of*.

Your attempts will originally get plenty of feedback – this is reasonable, as you don't yet know about design. You will learn through the feedback you receive.

Better to try and fail than to delay and present a broken report.

Every week, you need to produce an artefact

GUIDELINES

DOCUMENTS

- Learning Goals and achievements (Goals.pdf)
- Guidelines for ASD (Overview.pdf)
- Match System (requirements.pdf)

CONTENTS

- Practical Matters
- Course overview – topics, text book
- Project
- Feedback-driven approach to learning
- **Forming groups**

FORMING GROUPS

FORM GROUPS

Groups of 4

Ensure that you have 30 minutes time slot every other week with TA

You can leave when you have signed up.

Do not leave room without joining a group.

(email to kiko.fernandez@it.uu.se)

BEFORE THE LECTURE

COMMON QUESTIONS

- *Can we start working on the project?*
 - Yes, you are encourage to
- *Do we have to write an implementation in code?*
 - No, you have to create a good design (using UML)
- **Advice**
 - If I don't reply your question, ask again
 - To get the most of this course, we **encourage you to write code** that matches your design

COMMON QUESTIONS

- *How are the achievements related to the project?*
 - Achievements are the things that you need to do.
 - You will design the MatchSystem little by little, completing achievement after achievement

COMMON QUESTIONS

- How can I pass the course?
 - Plan the achievements you want to work on and get feedback on as many as possible
 - In order to pass, you need to complete all achievements marked as 3
 - To get grade 4, you need to complete all achievements marked as 3 and 4
 - To get grade 5, talk to me and we setup something interesting

COMMON QUESTIONS

- I don't have a group, what can I do?
 - Actively talk to people during the breaks
 - Contact us and we will possibly find a group for you
- Why covering so much content in 3 - 4 weeks?
- If I don't have a group... start creating a draft of some achievement

GET FEEDBACK ON ACHIEVEMENTS

- I. Your TA will send a Doodle
2. One member is the leader and:
 - I. Marks one choice from the Doodle with team number
 2. Puts artefacts in the folder 24 h before meeting