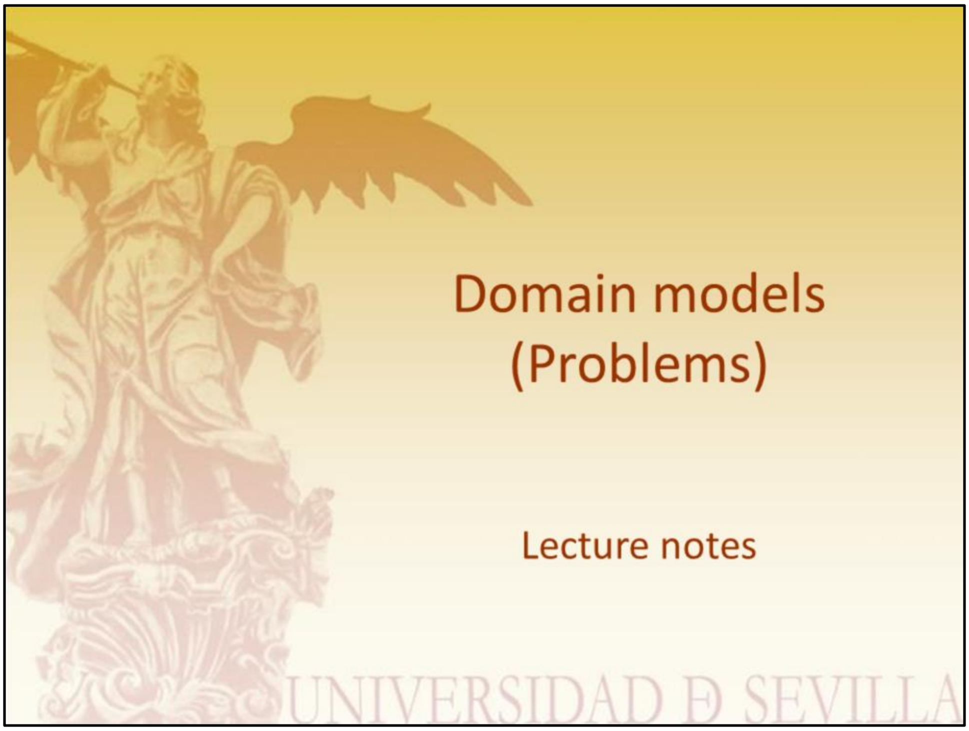


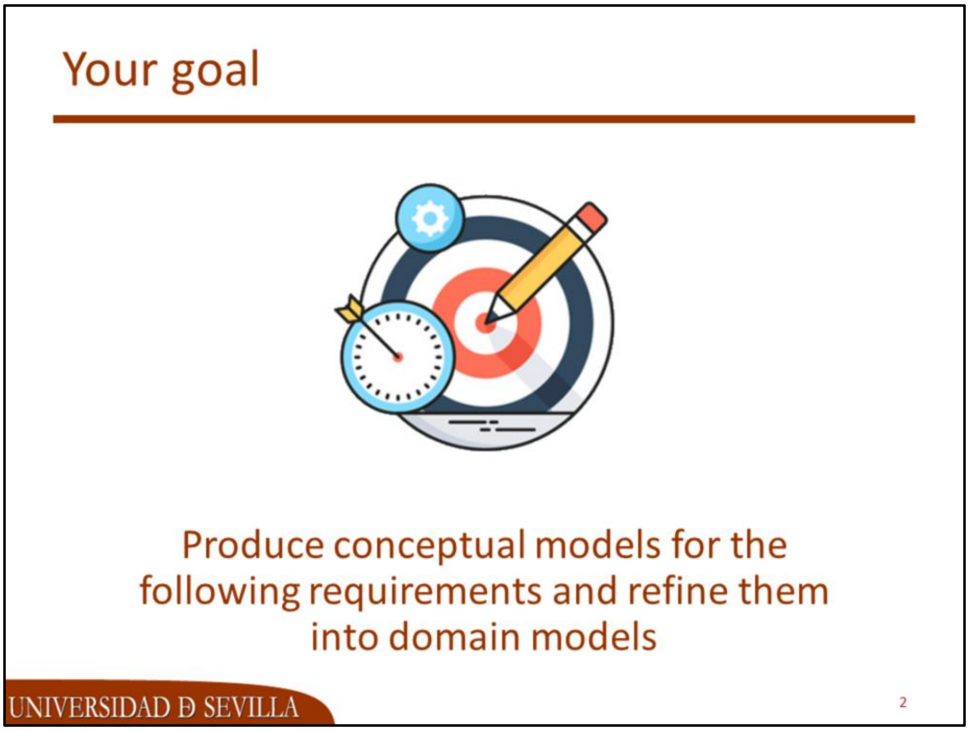
Domain models (Problems)

Lecture notes

UNIVERSIDAD Ð SEVILLA

Welcome to this problem lecture. The volunteers are going to present their solutions to the problems, and the class is expected to discuss and produce feedback.

--



Your goal

Produce conceptual models for the following requirements and refine them into domain models

UNIVERSIDAD DE SEVILLA

2

Your goal is to produce conceptual models for the requirements that are excerpted in the following slides and refine them into domain models.

Your presentation

Allocate 1 min to plug your laptop, 5 min to present your solution, and 10 min for class discussion

Regarding your presentation, please, note that you have 15 minutes. Allocate 1 minute to plug your laptop, 5 minutes to present your solution, and 10 minutes for class discussion.

Check your cables!

Check the overhead projector socket!
Bring an adapter if necessary.

Please, check the connections of our computer beforehand. Make sure that you can plug your computer to the overhead projector. Overhead projectors typically require a VGA connection; if your computer doesn’t have a VGA socket, please, make sure that you have the appropriate adapter.

Come on! First volunteers, please!

Come on! First volunteers, please!

Problem #1: actors and messages

- R1: The actors of the system are administrators, customers, and handy workers
- R2: Actors can exchange messages

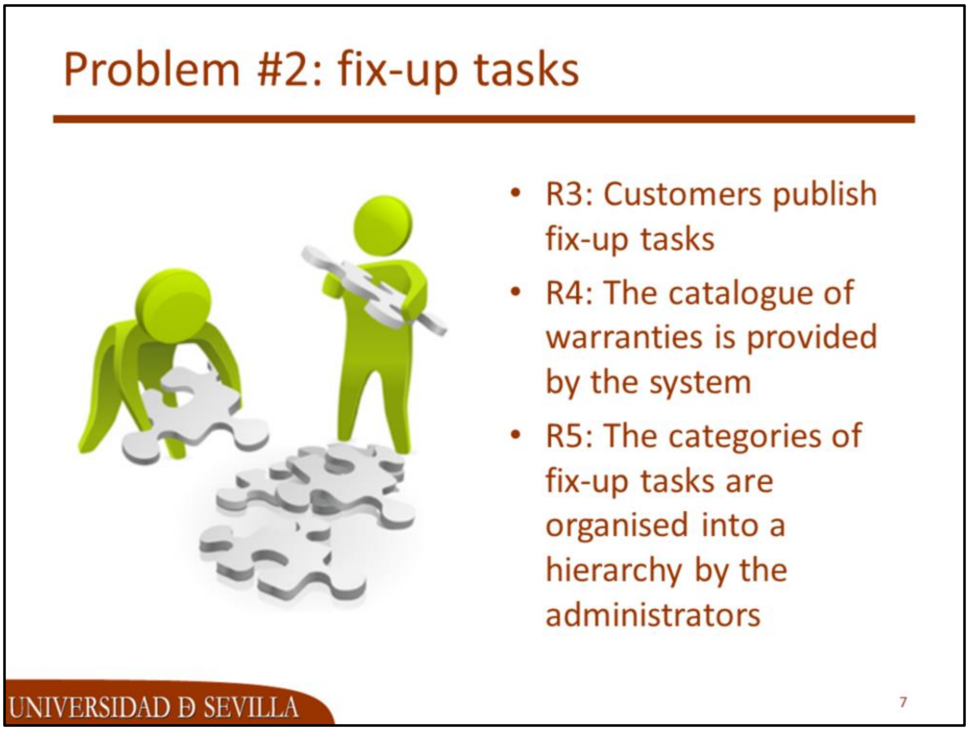
UNIVERSIDAD Ð SEVILLA

6

The first problem's regarding modelling actors and messages. These are the requirements that you have to model, but don't forget to review the others:

R1: The actors of the system are administrators, customers, and handy workers. For every actor, the system must store a name, an optional middle name, a surname, an optional photo, an email, an optional phone number, an optional address, and an arbitrary number of social profiles. The system must store the following data regarding such profiles: a nick, the name of the social network, a link to a profile in that social network. The system also stores the make of every handy worker; by default, the make is his or her full name, but it can be changed at will.

R2: Actors can exchange messages. For every message, the system must keep track of the sender, the recipient, the moment when it was sent, the subject, the body, its priority, and some optional tags. Priorities are HIGH, NEUTRAL, or LOW; no other values are expected. Every actor has the following message boxes: in box, out box, trash box, and spam box. When an actor receives a message, it gets to the in box unless the system flags it as spam, in which case it gets to the spam box. When he or she sends a message to another user, it's saved to the out box. When an actor removes a message from a box other than trash box, it is moved to the trash box; when he or she removes it from the trash box, then it is actually removed from the system. The previous boxes are pre-defined and the actors must not be allowed to delete them, to change their names, or to move them. Actors are allowed to create new boxes that they can manage arbitrarily. Note that a message may be stored in several boxes and, but the system must keep a unique copy; removing a message from the “trash box” removes it from every other box.



Problem #2: fix-up tasks

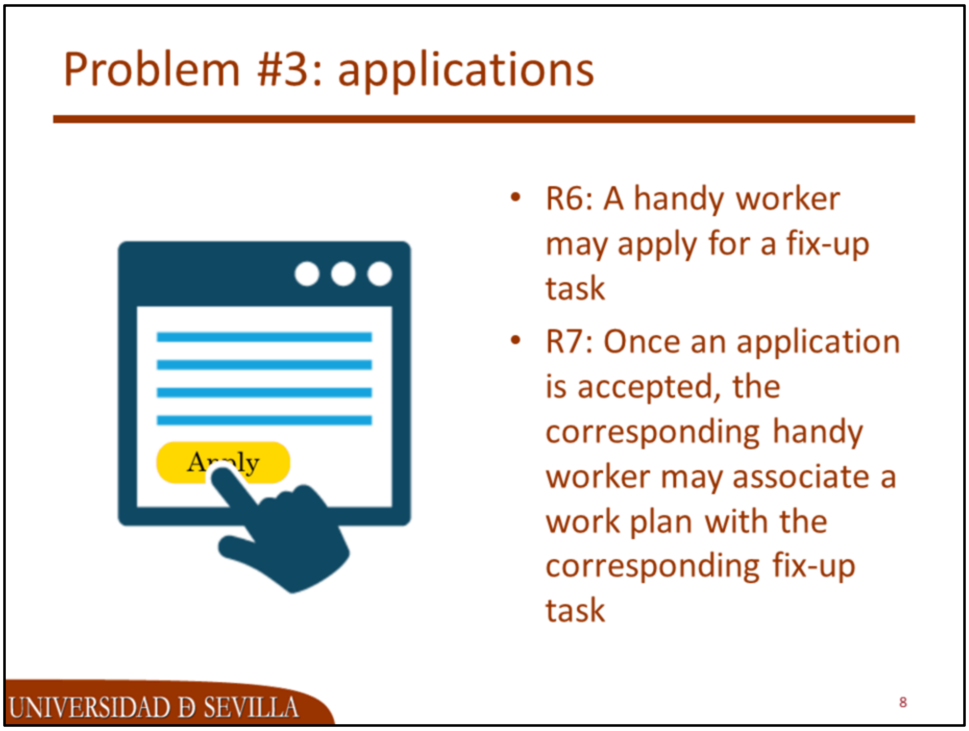
- R3: Customers publish fix-up tasks
- R4: The catalogue of warranties is provided by the system
- R5: The categories of fix-up tasks are organised into a hierarchy by the administrators

UNIVERSIDAD Ð SEVILLA

7

The second problem's regarding modelling fix-up tasks. These are the requirements that you have to model, but don't forget to review the others:

R3: Customers publish fix-up tasks. For every fix-up task, the system must store a ticker, the moment when it's published, a description, an address, a maximum price, a period of time to carry it out, a warranty required, and the category to which it belongs.
R4: The catalogue of warranties is provided by the system. For every warranty, the system must store a title, its terms, and the applicable laws.
R5: The categories of fix-up tasks are organised into a hierarchy by the administrators. Every category belongs to a parent category, but the root category, which is called "CATEGORY".



Problem #3: applications

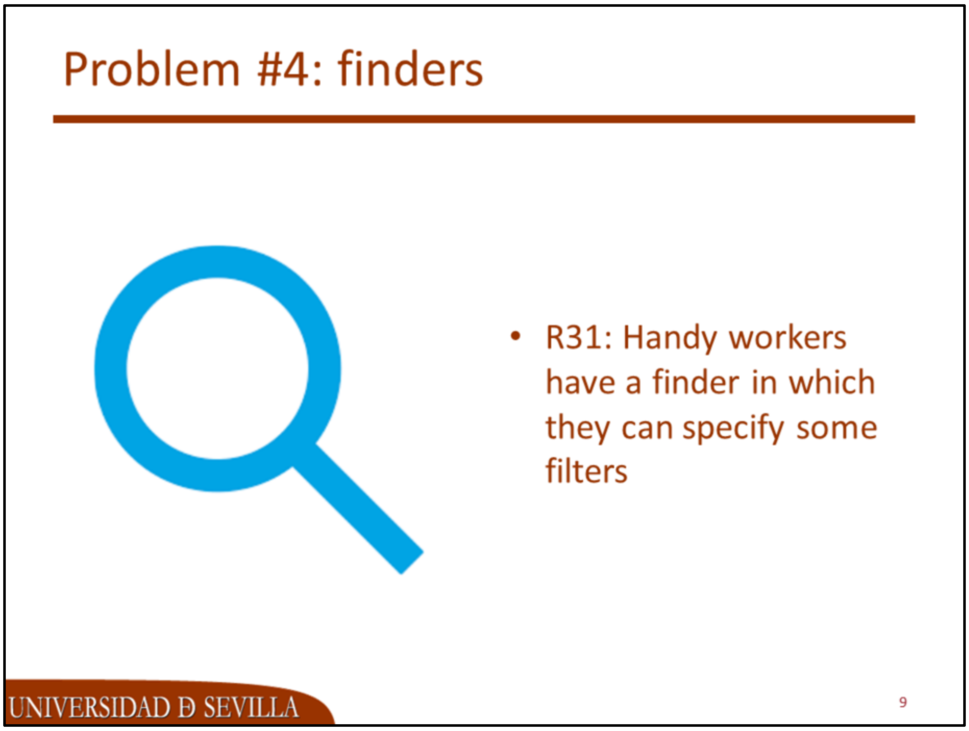
- R6: A handy worker may apply for a fix-up task
- R7: Once an application is accepted, the corresponding handy worker may associate a work plan with the corresponding fix-up task

UNIVERSIDAD Ð SEVILLA

8

The third problem's regarding modelling applications. These are the requirements that you have to model, but don't forget to review the others:

R6: A handy worker may apply for a fix-up task. For every application, the system must store the following data: the moment when the application is registered, its status, an offered price, and some comments. The status can be either pending, accepted, or rejected.
R7: Once an application is accepted, the corresponding handy worker may associate a work plan with the corresponding fix-up task. A work plan is composed of an arbitrary number of phases; for each phase, the system must store a title, a description, a start moment, and an end moment. Multiple phases may overlap, but none of them can be scheduled before or after the period of time during which the corresponding fix-up task must be carried out.

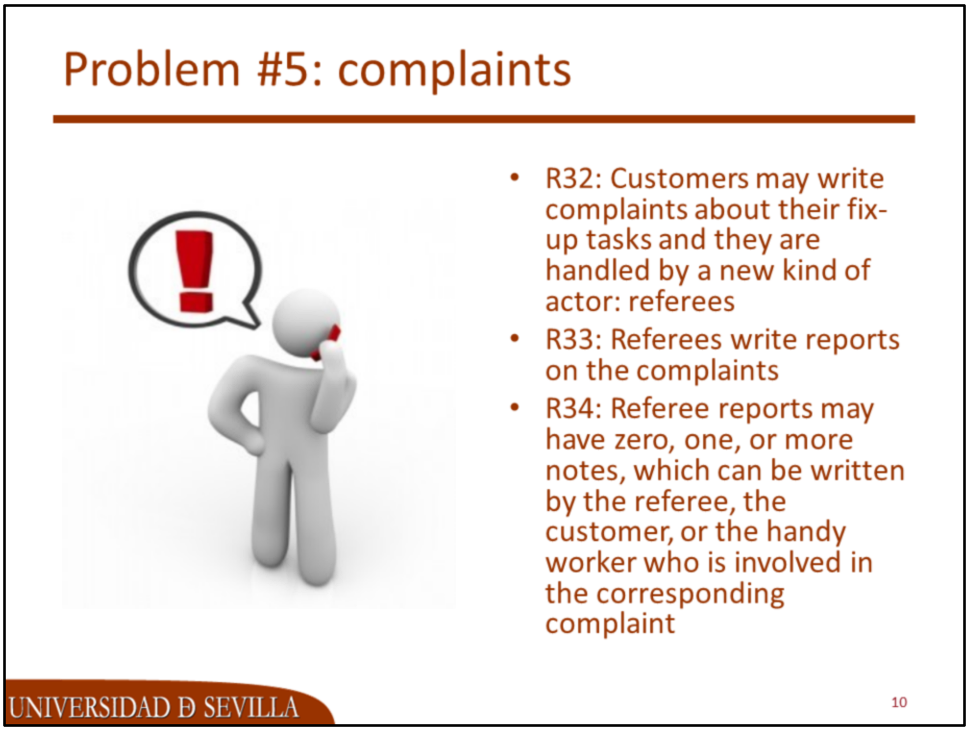


Problem #4: finders

- R31: Handy workers have a finder in which they can specify some filters

The fourth problem’s regarding modelling finders. These are the requirements that you have to model, but don’t forget to review the others:

R31: Handy workers have a finder in which they can specify some filters: a single key word that must appear somewhere in the ticker, description, or address of a fix-up task; a category to which the task must belong; a warranty required by the task; a range of prices; or a range of dates. The finder stores the fix-up tasks that pass the filters for one hour by default.



Problem #5: complaints

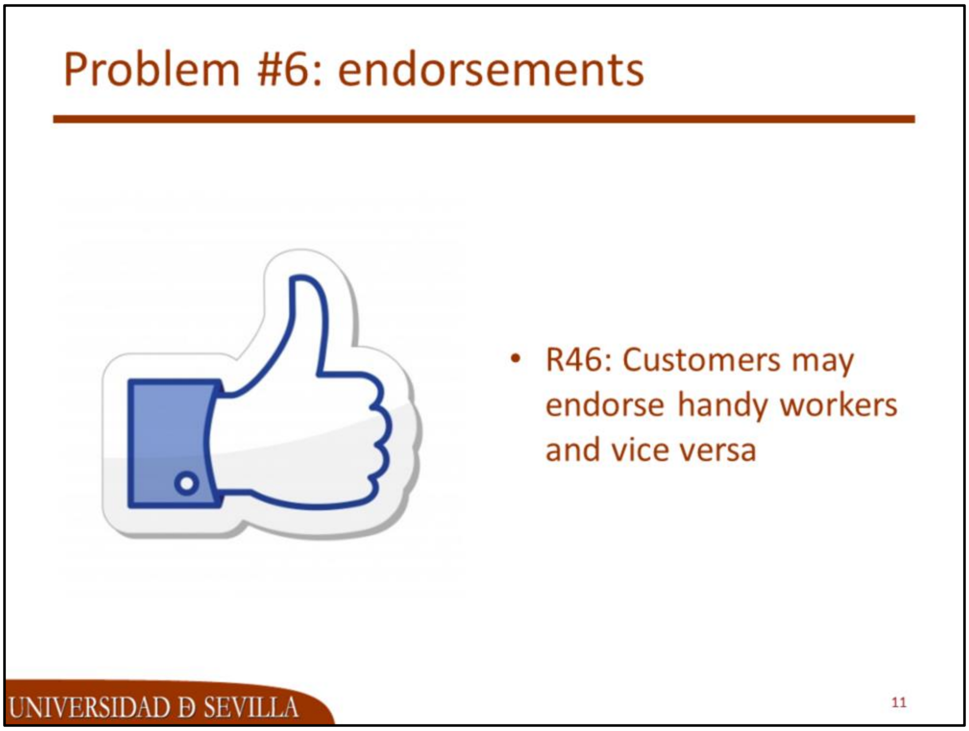
- R32: Customers may write complaints about their fix-up tasks and they are handled by a new kind of actor: referees
- R33: Referees write reports on the complaints
- R34: Referee reports may have zero, one, or more notes, which can be written by the referee, the customer, or the handy worker who is involved in the corresponding complaint

UNIVERSIDAD Ð SEVILLA

10

The fifth problem's regarding modelling complaints. These are the requirements that you have to model, but don't forget to review the others:

R32: Customers may write complaints about their fix-up tasks and they are handled by a new kind of actor: referees. For each complaint, the system must store a ticker, a moment, a description, and a number of attachments.
R33: Referees write reports on the complaints. For every such report, the system must store the moment when it's written, a description, and some attachments.
R34: Referee reports may have zero, one, or more notes, which can be written by the referee, the customer, or the handy worker who is involved in the corresponding complaint. For every note, the system must store the moment when it was written, a mandatory comment by the actor who creates the note, and optional comments by the other actors involved.



Problem #6: endorsements

- R46: Customers may endorse handy workers and vice versa

The sixth problem’s regarding modelling fix-up tasks. These are the requirements that you have to model, but don’t forget to review the others:

R46: Customers may endorse handy workers and vice versa. For every endorsement, the system must store the moment when it’s written and some comments.

Thanks!

Light of the moon
Moves west, flowers' shadows
Creep eastward

Thanks for attending this lecture! See you soon.