Nils Hagner 4346038 Felix Karg 4342014
Michael Fleig 4340085 Anush Davtyan 4368689

Solutions for Excercise sheet 3

Exercise 1 – Requirements Elicitation

i Explicitly different situations represented in the following questions.

ii

Question 1:
With 'The game' is there ment to be collected information about the game (and or it's mechanics) or about the application? Is the to-be-collected information supposed to reflect in-game status or general information like CPU-usage, RAM-usage, execution time or bandwidth?

Answer 1:
The collected statistics should be game-related. That is, the game should collect statistics about its behavior and mechanics. But we like your idea of displaying the ping and currently used bandwidth. Besides the in-game statistics, this should also be displayed.

Result 1:
Collect from within the game useful information for the player.

Question 2:
Is 'The game' supposed to save the collected information or is it okay to begin anew each start?

Answer 2:
That depends on the kind of statistics.

Result 2:
Save user-created situation data, but forget (on the long run) environmental information (such as ping, CPU load or other physically dependent unnecessary information).

Question 3:
Does 'displaying' mean that the statistics is somehow available for the player or might it only be supposed for developers through a (debug) console or the like? Display in what form, e.g. 'raw' data in a table or does it have to be a nice-looking graphic?

Answer 3:

The statistics should always be available for the player to view. Otherwise, collecting statistics doesn't make much sense. We are not interested in the development process. If the developers need to display some internal data for themselves, they very well may do that, but don't bother the player with that internal gibberish.

Result 3:

An interface for displaying statistics in a good-looking way to the player is required. This does not have to be centralized, but accessible and understandable.

Question 4:

Should the statistics differ in the their topic, eg. should they be closely related or cover broader area? How many of them should relate to the players individual activity, and how many different statistics are we (the developers) allowed to collect?

Answer 4:

You (the developers) must collect at least 5 different statistics. The statistics should be sensible and not too arbitrary. That is, the player should be able to retrieve information useful to him to evaluate his gameplay. Therefore, the "topic" or "broader area" depends on the type of game and the possibly collectible statistics.

Result 4:

The information collected should be of help for the user to evaluate his gameplay, or help him in some other sense. Depending on the game different values migh be of interest.

Question 5:

The statistics could be saved as a chronological list of events which allows us to have a clear understanding of when and what things happend and provide easier support if the player experiences. Or we could save them in a tablerized way which would have allow us to compare the different metrics.
For player related statistics we recommend the second option.
Which option would you like us to implement?

Answer 5:
Yes, the first part is a good idea. What is being collected is dependent on the kind of statistics. This could, e.g. be a replay function where the player can review the entire game and evaluate his gameplay.
The second one could also make sense if the collected statistics allow for displaying them in a tabular fashion. This could be overall time spent, built units, used resources, etc. Depending on what the game provides to collect. We like both ideas. Realize them both if it helps the player to understand the implications of the data for his game better.

Result 5:
Collecting gameplay information is mostly useful for the user and thus should be collected and (displayed in a nice fashion). Different ways of displaying them might be required depending on the type of statistics.

Before, those interpretations seemed all plausible:

Here are three different Interpretations of the sentence:

The game must collect and display at least 5 different statistics.

Interpretation 1:
The game collects statistics about the users progress and behaviour, and will make them available through either showing it in-game or through sharing it on facebook or other social media.

Interpretation 2:
There will be collected a lot of information about the behavior of the application, like memory usage or CPU-utilarisation, effective multicore-usage or GPU-load and settings.

Interpretation for the Originality Challenge:
We consider, that the statistics to be collected is a questionnary with several questions (at least 5), which have to be answered by every player. The questions should ask for how much the player liked the game and (maybe) some (private) information about him if he wants to share. Those statistics would then be send to the developers and might be displayed in a nice fashion for them (within their part of the game).

Exercise 2 – Analysis of Decision Tables

i

ii

iii

iv

Exercise 3 – Creation of Decision Tables

i

ii

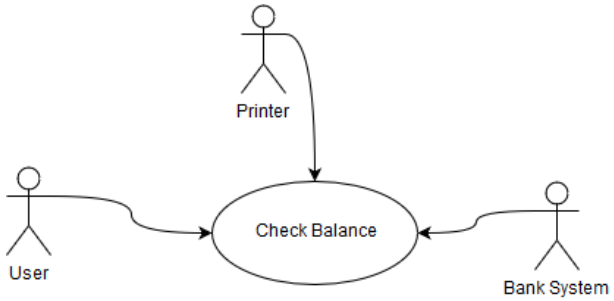
iii

Exercise 4 – Use Cases

name	Chack Balance
goal	Show specific information about the latest transactions
pre-condition	User Authenticated, showing Main Menu, ATM is operational
post-condition	showing Main Menu, Maybe printed Summary
post-condition in exception-case	withholding/returning card and return to authentication screen
actors	client (main actor), bank system, printer
open questions	none
normal case	1. User Presses 'Balance'-Button 2. get balance-information from bank-system 3. ATM shows Balance-Summary screen 4. User presses confirm 5. returns to Main screen
normal case 2	1. User Presses 'Balance'-Button 2. get balance-information from bank-system 3. ATM shows balance-summary screen 4. User presses print-button 5. disable print-button 6. balance summary will be printed 7. Success-Message appears on screen 8. User pushes confirm button 9. returns to Main screen
exc. case 0a	User not responding for 30sec 0.1 withhold card 0.2 return to authentication screen
exc. case 0b	Weird sensory data 0b.1 withhold card 0b.2 return to authentication screen
exc. case 2a	No connection to bank system could be established 2a.1 show error message 2a.2 show Authentication screen 2a.3 return card
exc. case 6a	Printer cannot print 6a.1 Message about not workign printer appears on screen 6a.2 (continue with default point 8)

i

ii



This use-case diagram only contains the use-case names and the actors.