REPORT ON DESIGN WORKSHOP

Members: Andrei Agape, Cristian – Dan Cojocaru, Cristina – Ioana Simionescu, David Green, Yoana Zaharieva
Class: DM83
University: UCN Technology & Business, Sofiendalsvej 60, Aalborg 9100
Date: November 9, 2012

Address Book with DVD Loan

Group 2 Workshop Project

Goal of the program
Requirments of the system
Evaluation of group dynamics
Main Menu TUI
Usecases
Design Class Diagrams
Observations of coding
Code standards
Description of tests made
Conclusion

Goal Of The Program

- An address book designed to be able to hold information about our friends and a list of DVD's that we are willing to lend out
- The system should be designed in a 3 layer Arcitect

Requirements of the system

- The address book should be able to hold an ID number
- The address book should be able to hold, Name,Address, and Phone Number
- You should be able to register DVD's
- You should have the ability to modify information and get information about your friends and the DVD's
- You should have the ability to lend out DVD's
- You should be able to see what person has borrowed which DVD

Evaluation of group Dynamics

Satisfaction of the group process

Generally the group is satisfied with the process. Members managed to cooperate easily and to communicate freely. Everyone was calm and managed to cope with every problem that occurred.

Self evaluation

Members are satisfied with their performance. However, if more time was available, a lot more positive changes to the system could have been made. It is generally accepted that most members gave a lot of themselves to finish the workshop on time.

Communication

Generally the group discussed every step together. Cooperation and communication between the members were good. Everyone managed to fit in a group and to reach their aim at the parts of the project they were supposed to do.

Evaluation of group dynamics

Leaders

It was decided that that the group is not supposed to have a leader. Members worked as a team without formal or informal leader.

Supervision

There was not a project manager of the group. All members took different initiatives in order to contribute to the finishing of the workshop.

Vision of the project

Members believe that if more time was available more changes could have been done. However, the general attitude is positive and team workers are satisfied.

Testing the system

All of the members at some time or another tested the system. If a bug was found, the members that did the programming tried to fix it as soon as possible.

Evaluation of group dynamics

Rules and regulations

All rules and regulations were informal. Members were present and ready for work.

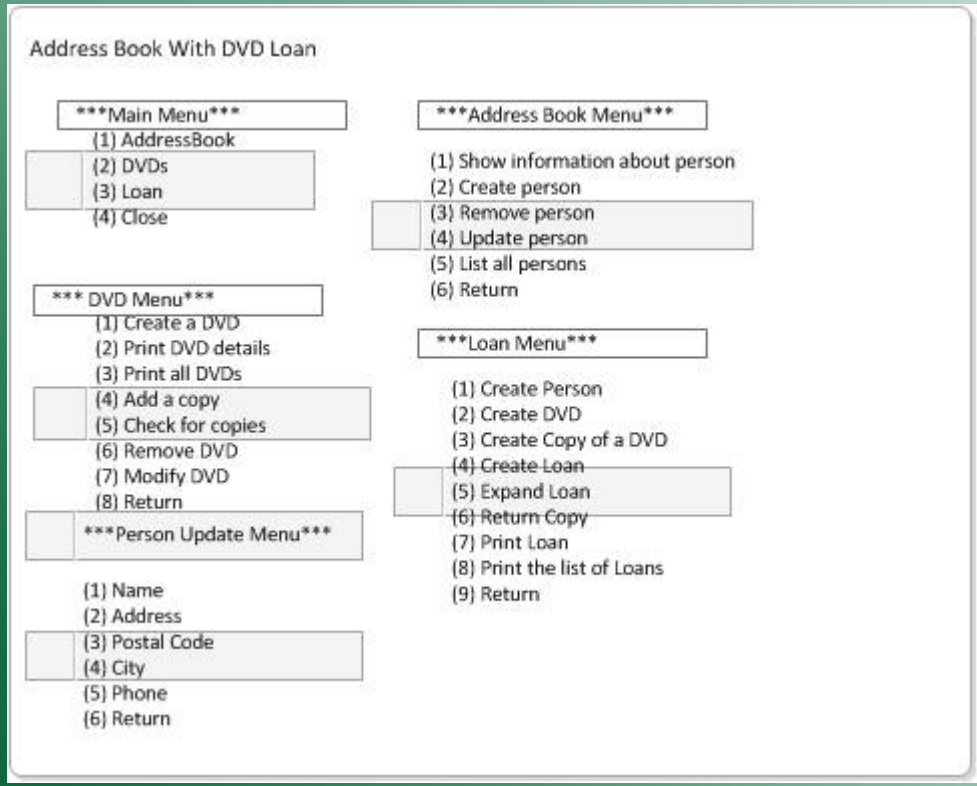
Respect

The atmosphere in the group was good. Members managed to find a good way of cooperation and working as a team.

Requirements of the system

- The address book should be able to hold an ID number
- The address book should be able to hold, Name,Address, and Phone Number
- You should be able to register DVD's
- You should have the ability to modify information and get information about your friends and the DVD's
- You should have the ability to lend out DVD's
- You should be able to see what person has borrowed which DVD

Main Menu TUI



Fully Dressed Use Case

Use Case: Handling of an address book

Level:
User-Goal

Primary actors:
Private Person

Stakeholders and Interests:
Person wants to be able to register personal information, to change it, to get it and to delete it.

Preconditions:
User has program opened.
User wants to register in the address book.

Success guarantee (post conditions):
Personal information is registered in the address book.
Personal information can be accessed.
Personal information is modified.
Personal information is deleted.

Main success scenario:
1. User opens the program.
2. System asks for personal information.
3. User inputs personal ID, name, address, postal code, and city and phone number.
4. System saves the information to database.
5. User accesses his information by inserting his ID.
6. System displays the information.
7. User wants to modify his personal information.
8. System acknowledges the modification.
9. User deletes the entry by inserting ID
10. System deletes the entry.

Extensions:

2a: Same id entered
1. System asks for valid ID, ID already exists.
2. User types another ID.

2b: Same phone number entered.
1. System asks for valid phone number.
2. User types a valid phone number.

4a, 6a, 8a: ID not found.
1. System asks for valid ID.
2. User inserts a valid ID.

Frequency of occurrence: regularly

Use Case: Handling of DVD UC2

Level:
User-Goal

Primary actors:
Private Person (User)

Stakeholders and Interests:
User wants to be able to register a DVD in the system update and delete it.

Preconditions:
User has chosen to register DVD.

Postconditions:
DVD is registered in system.
DVD information can be accessed in the system.
DVD information can be modified.
DVD information can be deleted.

Main success scenario:
1. User wants to create a DVD.
2. User inputs DVD information id, title, artist, and publication date.
3. System saves the information to database.
4. User accesses the DVD by title.
5. System displays the information wanted.
6. User wants to modify the DVD information.
7. System acknowledges the modification.
8. User wants to delete DVD from system.
9. System deletes the entry.

Extensions:

2a: DVD has already been registered in the system.
1. System prints that the DVD has already been registered.
2. User inserts new attributes.

4a, 6a, 8a: DVD is not found in the system.
1. System prints that DVD is not found in the system.
2. User inserts another title.

Frequency of occurrence: regularly

Fully Dressed Use Case

Use Case: Handling of an address book

Level:
User-Goal

Primary actors:
Private Person

Stakeholders and Interests:
Person wants to be able to register personal information, to change it, to get it and to delete it.

Preconditions:
User has program opened.
User wants to register in the address book.

Success guarantee (post conditions):
Personal information is registered in the address book.
Personal information can be accessed.
Personal information is modified.
Personal information is deleted.

Main success scenario:
1. User opens the program.
2. System asks for personal information.
3. User inputs personal ID, name, address, postal code, and city and phone number.
4. System saves the information to database.
5. User accesses his information by inserting his ID.
6. System displays the information.
7. User wants to modify his personal information.
8. System acknowledges the modification.
9. User deletes the entry by inserting ID
10. System deletes the entry.

Extensions:

2a: Same id entered
1. System asks for valid ID, ID already exists.
2. User types another ID.

2b: Same phone number entered.
1. System asks for valid phone number.
2. User types a valid phone number.

4a, 6a, 8a: ID not found.
1. System asks for valid ID.
2. User inserts a valid ID.

Frequency of occurrence: regularly

Use Case: Handling of DVD UC2

Level:
User-Goal

Primary actors:
Private Person (User)

Stakeholders and Interests:
User wants to be able to register a DVD in the system update and delete it.

Preconditions:
User has chosen to register DVD.

Postconditions:
DVD is registered in system.
DVD information can be accessed in the system.
DVD information can be modified.
DVD information can be deleted.

Main success scenario:
1. User wants to create a DVD.
2. User inputs DVD information id, title, artist, and publication date.
3. System saves the information to database.
4. User accesses the DVD by title.
5. System displays the information wanted.
6. User wants to modify the DVD information.
7. System acknowledges the modification.
8. User wants to delete DVD from system.
9. System deletes the entry.

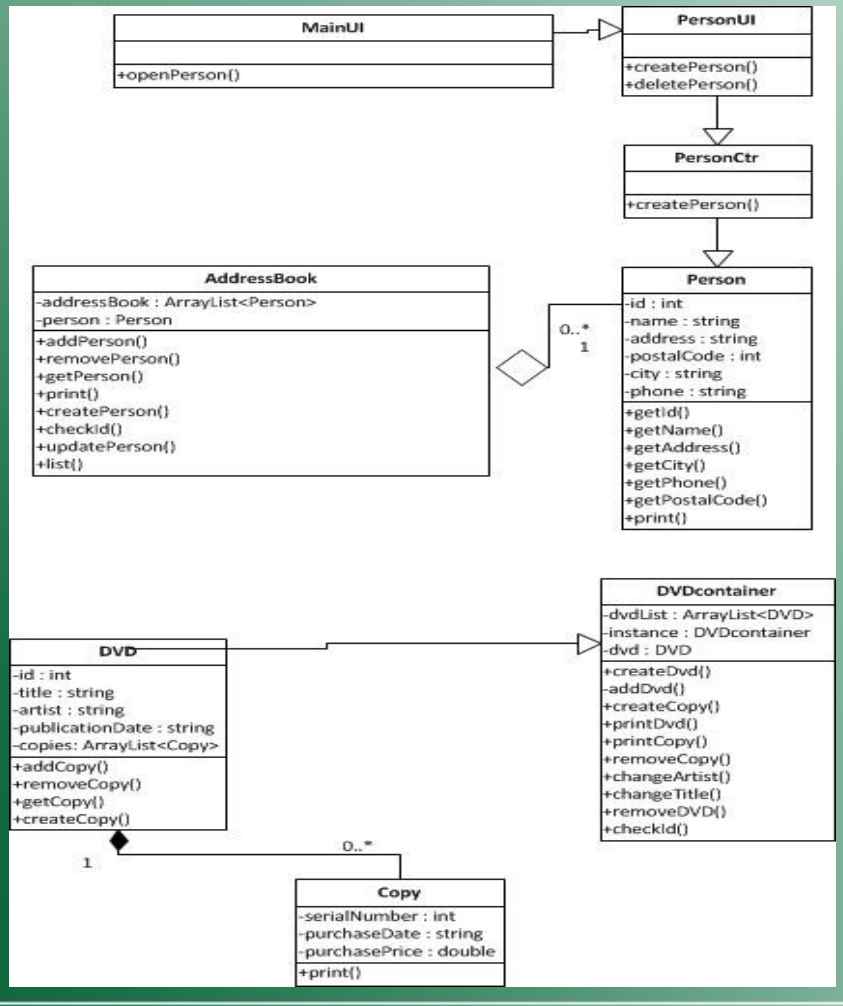
Extensions:

2a: DVD has already been registered in the system.
1. System prints that the DVD has already been registered.
2. User inserts new attributes.

4a, 6a, 8a: DVD is not found in the system.
1. System prints that DVD is not found in the system.
2. User inserts another title.

Frequency of occurrence: regularly

Main Design Class Diagrams



Observations of coding

- Our biggest challenge in writing the coding was working as a team because our coders are use to working alone it was a very good practical experience for them.
- The next challenge we experienced was in writing the loan class due to the fact that it points to most of the classes and getting it to function the way we wanted it to presented a challenge.
- We used an array list to store the information about the users and also the DVD's

Code Standards

When writing the code, the part of the group in charge took into account several rules about naming, layout, documentation and language use.

As for what **naming** is concerned, they used meaningful names, they used singular nouns class names (which they started with capital letters) and they made sure that the method and variable names started with low-case letters.

For the **layout** part, they made sure that braces were used in a correct way, blank lines were added between methods and constructors and spaces were placed around operators and before the opening brace of a control structure's block.

Since **documentation** is also an important part, the members decided to add comments only when necessary, such as at the top of every class and for every method, while at the same time making them Javadoc-readable.

Last, but not least, they tried to use proper **language**: the order of declarations was followed, both private and public fields were created, the classes were imported separately and a constructor that initialized all the fields was always added, regardless of the content of the body.

Description of tests made

In order to check whether the source code was running properly and to correct the possible mistakes, the group as a whole ran a few tests that proved to be essential in the process.

- **Test 1:** One of the tests performed involved trying to add a person to the address book by inserting an ID that already existed, knowing that this should return an error (negative testing). As expected, the message 'ID already exists.' was printed.

- **Test 2:** Another test included adding DVDs to the DVDcontainer class and then printing a list of all DVDs to check if the correct information is printed (positive testing). Consequently, all DVDs that had been added were printed.

Conclusion

- To summarize the work we did was in a cooperative approach we had no group leader and just split all of the work in half to be completed.
- If we had more time we would have done the reports first then started the coding but due to time constraints we had to begin coding from the start of the project