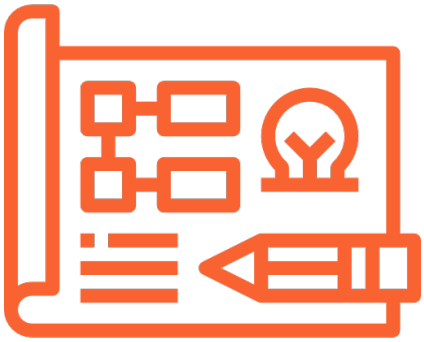
LOST MY PET



PLANNING REPORT

Piotr Przechodzki
Graded Unit Project

TABLE OF CONTENTS

INTRODUCTION

PROJECT BACKGROUND

The City of Hillwood is a mid-size city based in the UK. Recently City Council has a problem with lost pets (dogs and cats) and inhabitants are asking for the system where they could report a missing or found pet. At this moment, there is no any solution for that and if anyone wants to report a pet then need to phone directly, email or contact in person to one of many animal shelters that operate in the city. Then shelters usually contact (by email or phone) with others pet rescues and share information about lost or found pets. This method is slow and not guarantee that all details of lost pet will be passed correctly.

INITIAL REQUIREMENTS

The new system should include:

- Registration of members online.
- Verify user registration details.
- Members should be able to post an announcement for lost and found pet.
- Everyone should be able to view the catalogue of missing and found pets.
- Members should be able to change registration details.
- Members should be able to change the status of lost or found pet (edit, update and delete).
- Members should be able to contact with other members in case of ask questions or share contact details.
- The system should send a notification to members about lost or found pet in the area where the member lives.
- System Administrator should be able to generate reports for statistics.
- System Administrator should be able to moderate announcements and user accounts.

Platforms:

- There should be created a web and mobile version (Android) of the system.

INVESTIGATION

FACT-FINDING

In order to get as much information as we need to complete the project, we need methods that will help us to collect the most relevant knowledge to determine the functionality of the software which we create and perform the objectives. This process is called fact-finding or data collection and it is used in the early stage of System Development Life Cycle. It contains techniques that help us to better understand not only the business needs but also the future users of the software.

QUESTIONNAIRES

The questionnaire is a very useful fact-finding technique. It is used for research to collect information. It contains a set of questions that the respondents answer. Based on the collected materials conclusions, objective and functionality are formulated. Because in the case of doing the research with the use of a questionnaire, lack of communication and interaction with the person interviewed may exist so the collected data may not be significant. Therefore, the value of the conclusion derived from the research depends on the proper design of the questionnaire.

Advantages of Questionnaires:

- Simplicity of form
- Easy to reach respondents
- Affordability in analysis
- Convenience in the presentation of results
- Low cost
- They are not time-consuming

Disadvantages of Questionnaires:

- Inability to deepen the problem
- Sketchy and "quantitative" character
- No possibility of direct contact with the respondent
- Body language and gestures excluded from the analysis

INTERVIEWS

An interview is one of the basic and most commonly used methods to collect data. The interview involves asking the interviewer about more or less formalized questions. It helps

in identifying problems, needs, ideas and opinions. Thanks to meeting face to face, we are able to find out more about unclear issues and get more precise results. The interview allows free explanation of various doubts but is also more time-consuming.

Advantages of Interviews:

- Free expression of opinions
- The information is more objective
- Data collected are more precise

Disadvantages of Interviews:

- High price
- Research is time-consuming
- Limit the number of respondents

OBSERVATIONS

Observation is used as a method of collecting primary data, which consists in making systematic observations in a planned and intended way. Information gathered during the observation help in understanding the functioning and the needs of the business and the basic processes involved in its operation.

Advantages of Observation:

- Versatility and flexibility
- We can capture the business in a full context
- Usually high credibility

Disadvantages of Interviews:

- Expensive and time-consuming method of obtaining data
- Collected data may be subjective

INTERVIEW QUESTION AND RESPONSE

PROJECT SCOPE

1. Can you confirm the scope as outlined in the Project Brief is still the same?

1) A database system that incorporates:
   a) User details
   b) Pet deteils
   c) A list of missing pets
   d) A list of found pets
2) A program that allows user to:
   a) Register new user
   b) Login user
   c) Report a pet
   d) View list of pets
   e) Update user details
   f) Delete user account
   g) Update pet report
   h) Delete pet report
   i) Send a message to the owner/finder

2. Is there anything that you feel has been missed out?

No.

3. Then is the de data for the project to be complete?

8th June 2018

4. What is the budget for the project?

£20.000

MEMBERS REGISTRATION

1. What data is a customer required to provide for registration?

- First Name(s)
- Last Name(s)
- Location
- Email
- Password

2. What will data need to be validated to successfully finish registration?

All of the above except for the last name and location.

POSTING AN ANNOUNCEMENT

3. Who can post an announcement about lost or found pet?

All registered users (members).

4. What pet information is required to report a missing pet?

- Name
- Type (dog or cat)

- Sex
- Location (where was lost)
- When (missing on)
- Description (circumstances, situation)
- Chipped (yes or no)
- Collar (yes or no)
- Colour
- Age
- Photos

5. What pet information is required to report a found pet?

- Type (dog or cat)
- Sex
- Location (where was found)
- When (found)
- Colour
- Description (circumstances, situation)
- Photos

PETS CATALOGUE

6. Who can browse pets lost and found catalogue?

All registerd and unregistred users.

7. Who can see report details?

All registerd and unregistred users.

8. Who can contact with the announcement owner?

All registerd users.

9. How can user contact/send a message to the announcement owner?

Every report will have a button to send a message.

10. Will the catalogue be divided into city districts?

Yes, you can filter results by city districts (location).

11. By what data user can filter catalogue?

- Location
- Type

12. The list will be sorted by what order?

- By Missing Date
- By Date Added

13. What's happen when lost or found pet will reunion?

User should delete the announcement from the catalogue. If not then Administrator should delete old (more then 6 months) reports.

14. Can user change a details of the announcement?

Yes. Can change or add these details:

- Sex

- Location
- Colour
- When missing/found
- Description

User can't change te pet type and pet name.

15. Is there is any limit of announcement per member?

No.

ANDROID VERSION

16. What functions will have a mobile version?

Same functions as the web version except the Administrator rols.

17. Can user register using a mobile version?

Yes.

18. Can members send a message to each other using mobile version?

Yes.

REPORTS

19. Who can generate reports?

Administrator

20. What information needs to be include in report?

A PDF or XLS report will be generated from/to given date by reports added date. Report will contain:

- Type
- Name
- Sex
- Location
- Chipped
- Colour
- Collar
- When lost/found
- Added date (announcement)
- Age
- Total number

ADMINISTRTOR ACCOUNT

21. What's the Administrator role?

Accept every announcement before it will show (moderate function). User can also report (button on the announcement page) if there is something wrong with the announcement.

22. Is Administrator role same in Web and Android version?

Administrator cannot login on Android version. Whole Administrator Panel is only for a web version.

23. Should the system allows to register a new Administrator?

Yes. Adminstrator should be able add a new account with administrator role.

NON-FUNCTIONAL REQUIREMENTS

24. Will the old (completed) announcements stored in the database for ever?

No.

25. Can unregistred user browse catalogue using a Web/Mobile version?

Yes.

26. How members should be notified about lost/found pet in the area where their lives?

Notification icon (alert) will appear.

27. What is the colout scheme to be used for the Application?

No any particular colour, but the page should be responsive, have clean and modern look.

FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

FUNCTIONAL REQUIREMENTS

User can register a new account using web and mobile version, required to provide the following details: first name, email and password. The last name and the area where the member lives are not mandatory. Once the user has successfully registered the system will send out confirmation email.
User can log in to his account using web and mobile version, required to provide the following details: email and password.
Registered user can place a new announcement for lost pet.
Registered user can place a new announcement for found pet.
Registered user can update their announcement.
Registered user can delete their announcement.
Registered user can edit their account details.
Registered user can send a message (contact) to the owner of the announcement.
Registered user can delete the account.
All users can view lost/found pets.
The system should send a notification to members about lost or found pet in the area where the member lives
Registered user can report an announcement to the administrator.
When placing a missing pet announcement user must enter pet name, type, location and the missing date. Other not mandatory details are breed, gender, pictures, colour, age, is pet chipped, does the pet have a collar, description.
When placing a found pet report user must enter type, location and found date.
Other not mandatory details are breed, gender, picture, colour and description.
When deleting a user account, all this user announcement need to be deleted as well.
Have separate access facilities for users and administrators.
Administrators can generate lost and found pet report.
Administrator can generate a PDF or XLS reports "from – to" by date added or missing/found date.
Administrator report will contain pet type, name, sex, location, chipped, colour, collar, date, age and total number.

NON-FUNCTIONAL REQUIREMENTS

Budget is £20.000.
Deadline is 8th June 2018.
The web version needs to be compatible with old and up-to-date browsers.

Piotr Przechodzki

The web version needs to be responsive design for multiplatform such as phone, tablet and desktop/laptop computers.
The mobile version needs to be compatible with currently most popular Android platform API version (6.0 Marshmallow).
System will provide only two pet types: dogs and cats.
User when adding pet's colour can choose up to 4 colours at once.
User can edit or update all announcement details except the pet type and pet name.
All users can sort the catalogue by missing date or by date added.
All users can filter the catalogue by pet type and location.
Administrator panel is available only on a web version.
Only administrator can register a new account with administrator role.
When adding a new administrator, the administrator needs to provide a username and password.
All announcements need to be approved by the administrator before they will be available in the catalogue.
Administrator will get a notification in the administrator panel when a new announcement needs to be approved.
Assign registered user a unique ID.
Assign administrator members a unique ID.
Assign announcement post a unique ID.
Administrator can edit or delete user's announcement.
Administrator can delete or suspend user account.
Users need to have same functions on web and mobile version.
Password stored in the database must be encrypted (hashed) and salted to protect against rainbow tables.

USER STORIES

As a/n	I want to...	So that...	Notes
User	Register	I can start posting lost/found pet announcements	Registration should include: • First name • Last name • Email • Location • Password
User	Log in	I can post lost/found pet announcements, write messages and update my details	
User	Post lost pet announcement		Announcement should include: • Pet name • Type • Breed • Gender • Location • Missing Date • Colour • Is Chipped • Have Collar • Image • Description
User	Post found pet announcement		Announcement should include: • Type • Breed • Gender • Location • Found Date • Colour • Image • Description
User	Update my lost or found pet announcement		Edit announcement should include: • Breed • Gender • Location • Missing Date • Colour • Is Chipped • Have Collar

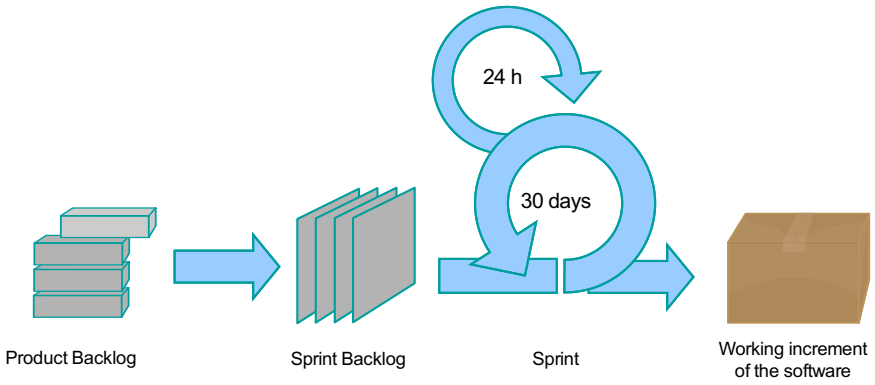
			• Image • Description
User	Delete my lost or found pet announcement		
User	Update my account details		Update details should include: • First name • Last name • Email • Location • Password
User	Send a message to the announcement author	I could contact with another member about lost or found pet	
User	Replay to the message	I could contact with another member about lost or found pet	
User	Delete my account		
User	View lost or found pets catalogue		
User	Be able to sort the lost or found catalogue by date		
User	Be able to group the lost or found catalogue by pet type or location		
User	Give a feedback to Administrator about selected announcement	I could report abuse or inappropriate announcement	
User	Get notification about lost or found pet in my area (location)		
Administrator	Log in	Access to separate facilities – Administrator Panel	
Administrator	Display a list of all registered users	View user account details or delete selected account	
Administrator	Display a list of all announcements	View or delete selected announcement	

Administrator	Display a list of all new posted announcements	I could moderate announcements	
Administrator	Register a new administrator account		
Administrator	Generate from selected date report on lost or found pets	I have an overview for statistics	Report should include: • Pet name • Type • Breed • Gender • Location • Added date • Missing/Found Date • Colour • Is Chipped • Have Collar • Description
User/Administrator	Log out		

DEVELOPMENT APPROACHES

SCRUM

Scrum methodology is one of the most popular Agile methodologies in IT Project Management, based on Agile principles. It is a methodology that gives the opportunity to solve complex problems and adapt the product to the customer's requirements and needs. Scrum allows the efficient and innovative creation of the product with the highest possible quality for the customer, due to the iterative (incremental) control process.



SCRUM HISTORY

For the first time, the general assumptions of the Scrum methodology were published in 1986 in the article "The New Product Development Game" in the Harvard Bussines Review magazine (Takeuchi & Nonaka, 1986). The article described the results of research on the methods of creating a new product carried out in top companies (Canon, Honda, Xerox, Brother, 3M, Epson, HP). These companies were observed during the implementation of the Lean Management method and the theory of complex adaptive systems, hence the foundations of Scrum refer to the experience gathered at that time (Sutherland, 2004).

The first Scrum team was registered at Easel Corporation in 1993. On the basis of this experience arose the article: "Agile Development: Lessons Learned from the First Scrum" by Ken Schwaber, who presented a unified course of the use of Scrum and also contributed to the promotion of this methodology on a global scale.

ROLES IN SCRUM

Scrum is based on the work of a **Development Team** that has all the necessary skills and resources to deliver a product such as analysis, programming and testing.

In the team, we also have a **Product Owner** who represents the interests of the client and also has a vision of the product and thus can decide what is most important at the moment. It depends on Product Owner whether the given functionality will be approved, returned or remedied.

The final role is the **Scrum Master**, which focuses on how the team operates. Scrum Master supervises the Scrum process of the project. It is also a support for the team, helps in the case of emerging problems, prevents their occurrence by investigating the activities performed by the team. It ensures that the Scrum process is carried out and focuses on ensuring that all tasks are carried out during the Sprint period. Scrum Master tries to make the Scrum process to bring as many benefits as possible for a given project.

SPRINT PLANNING

Sprint Planning is usually divided into two parts:

- Determination of the product register
- Determination of the task register

This stage is referred to as optional, but it is intended to minimize the amount of work in determining the scope of the project in its later stages. Product Owner, based on the customer's vision, defines the functional and non-functional features of the product. After specifying the product register, the task register is defined. At this stage, the Product Owner defines the priorities of the work and the Development Team asks questions that will help in the organization of work.

SPRINT

The team provides subsequent versions (increments) of a working product in short cycles called Sprint. Each Sprint is limited to a maximum of one month. This ensures Product Owner that he/she knows how much work invested and if something wrong happens then will not lose more than the length of Sprint.

PRODUCT BACKLOG

Product Backlog is the main list of all the functionalities desired in the product, which changes over time along with the process of specifying the requirements at the stage of product creation and with its subsequent development. The Product Owner is responsible for providing the requirements and the main purpose in the form of a product backlog. It defines the tasks to be carried out and determines their priorities.

SPRINT BACKLOG

Spring Backlog is a list of tasks that must be performed by the Scrum team during the next sprint. The list is created as a result of obtaining information from the Product Backlog. The

team selects from the list tasks to be performed in specific sprints - starting with those with the highest priority. The register is fixed and does not change - this is due to the fact that sprint requirements remain unchanged during the sprint.

DAILY SCRUM

Daily Scrum is a short meeting that takes place every day of the Sprint, where Scrum Master asks 3 questions to the Development Team:

- What did you do to achieve the Sprint goal?
- What will you do to achieve the Sprint goal?
- What obstacles did you encounter to achieve the goal?

The meeting has a strictly defined agenda so that it can take place in the shortest possible time and with the desired efficiency.

SPRINT REVIEW

Sprint Review is a meeting after each Sprint, which aims to discuss the functionality and characteristics of the product, as well as the course of work only for a fragment of the created product.

DEFINITION OF DONE

Definition of done is a standard/checklist that specifies the state of the completed work. When the Development Team says something is done then this means that each completed Product Backlog Item (PBI) becomes a functional increment of product. The Definition of Done is clearly determined by Product Owner and Development Team. Usually, an organization has its own basis for the construction of the Definition of Done. Example of Definition of Done:

- Unit Test at the level of 95%
- Code went through a review or was written using Pair Programming method
- The team performed functional tests
- The functionality works fine and does not spoil others (integration and regression)

SPRINT RETROSPECTIVE

Sprint Retrospective is a reflection stage after each Sprint. All members of the Scrum team meet with current issues regarding both product and organization of work.

TEST-DRIVEN DEVELOPMENT

Test-Driven Development (TDD) is a technique of writing and building software classified as the Agile methodology. Originally it was part of the extreme programming (XP), but now it is an independent technique.

OBJECTIVE OF TDD

The main objective of TDD are:

- Preserving high-quality design in classes.
- Avoiding misinterpretation of business requirements.
- Preserving simplicity in the code using two popular principles (Miguel, 2017):
    - YAGNI - "You aren't gonna need it"
    - KISS - "Keep It Simple, Stupid!"

In TDD, it's not about testing. Testable code and consequently an automatic and immediate error feedback is an extra thing in TDD. The key aspect of TDD is to write a test before writing the target code. You can write tests in parallel while writing business logic, you can also write tests after implementation, but then it is not Test-Driven Development. In TDD we write tests always first, before the code (Borg & Williamson, 2016).

RED-GREEN-REFACTOR

The key aspect of TDD is the test writing cycle. First, we write tests, then implement the functionality and finally, we refactorize. The cycle is usually called Red-Green-Refactor or TDD Mantra and it consists of three stages, which as a whole are repeated:

1. Red: Write a test that fails.
    a. Write tests for empty but existing classes and methods.
    b. Run the test and expect it to fail. This test cannot succeed because the functionality itself is not yet implemented. The situation in which unit tests do not pass is very often marked in red in the IDE.
2. Green: Write code that tests are successful.
    a. Implement the code (according to the documentation). At this point, it is important that this code is not perfect. It's about the quickest possible implementation that satisfies the assumptions of the test, which was written in the previous phase.
    b. Run tests. All tests must be successful. IDE indicates unit tests in green. It is important at this stage to run all unit tests written so far.
3. Refactor: Refactoring code - implement changes that improve the quality of the code, but do not change its functionality. An example of refactoring may be separation method, which removes duplicate code or to create an entirely new class which is responsible for a certain part of the tasks of the class. It may happen that the refactoring phase is not always necessary. Improving a good code does not necessarily lead to good results.

a. After refactoring run all tests to check if something has been broken.

ADVANTAGES

Advantages of Test-Driven Development:

- The main advantage of TDD is the quick capture of errors.
- Errors detected and corrected by the author of the code cost little because only one person is involved.
- The ability to test the functionality without running the entire software - the code can be run partially without running the entire application
- A very interesting advantage of TDD is the creation of specific documentation - the resulting of tests are simple and act as documentation for the code.
- Better code management over time.

DISADVANTAGES

Disadvantages of Test-Driven Development:

- It requires additional time to create unit tests - can be very discouraging for management when the developer needs more time to do the same tasks.
- It takes time to maintain the tests - for the tests to be useful, they must be appropriately managed and updated as the application logic changes.

WHEN NOT TO USE TDD?

Practice shows that there is no point in using TDD for short, few-line projects. In this situation, the redundancy of time is simply too big. It also makes no sense to use this methodology for applications that will never be developed or for existing code (the quality of such tests is usually very low) so it remains in existing projects to create unit tests for new functionality and to use the full benefits of Test-Driven Development in new projects.

OBJECT-ORIENTED APPROACH

Object-oriented analysis and design (OOAD or OOA&D) is one of the approaches used to create software. In a narrower sense of the word, this is a description of the system as a group of separate objects that act differently on each other. In a broader sense, this is a comprehensive approach to application creation that includes best practices for virtually every aspect of application creation that ultimately meets customer requirements (which are subject to constant change) while being easy to maintain and expandable.

CONCEPTS AND TECHNIQUES

Object-oriented approach in its essence contains many different techniques used to create computer applications such as:

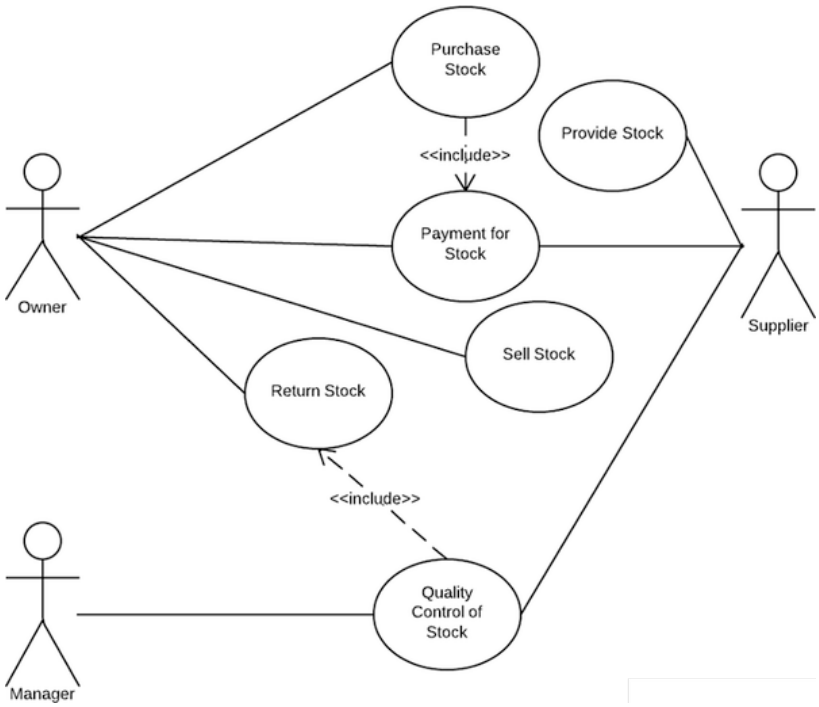
- Abstraction is the most often identified with the class. It is a model that does not actually represent any existing object, but only on the basis of which other objects are defined.
- Inheritance is a programming technique that allows the use of an existing class to create a new one, based on its predecessor. Classes inherit from other classes or objects from objects (in JavaScript).
- Polymorphism is otherwise a multiformity. It's a method that allows a virtual function to adopt different ways of its implementation. It gives the possibility to call the same method on different objects, but they (objects) can respond in different ways depending on the type.
- Encapsulation is a way of isolating selected data and functions (operating on these data) stored in one structure from the environment. Only the necessary parts of the program are visible, while the variables and functions are hidden and unavailable from the outside.

OBJECT-ORIENTED ANALYSIS

Object-Oriented Analysis (OOA) examines the problem and aims to create a conceptual model of information that belongs to a given area of research. Analytical models do not take into account any restrictions we face in implementation or specific form of the future system.

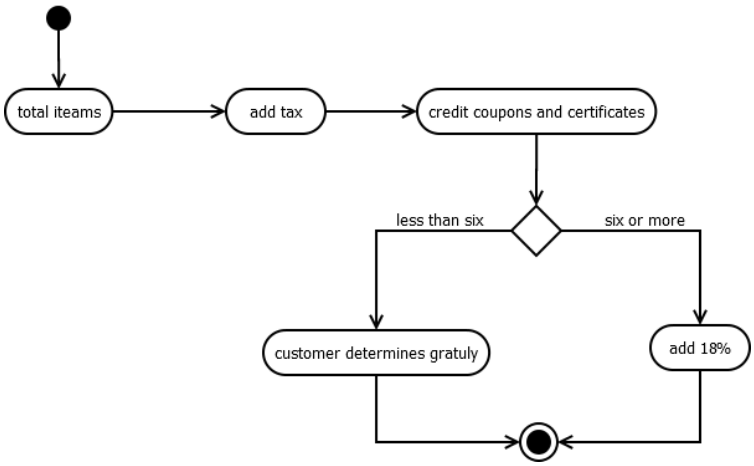
The result of the OOA is a description of functional requirements for the system in the form of a conceptual model. It usually consists of use case diagrams, class diagrams and several interaction diagrams. It can also include user interface previews. The goal of OOA is to create a model that describes the software as it works to meet all customer requirements.

Use case diagrams are intuitively used in traditional design systems during the analysis stage. They model a certain system function in a way that its future users will see. For large systems with many complex and interrelated functions of this kind of approach, it has a huge meaning. It allows you to forget about the structure/system architecture and its technical details and focus on the external features of the system. An example of uses case diagram is shown below.



The next diagram used during the analysis stage is activity diagram. It is used to model the dynamic aspects of the system. It presents the sequential or concurrent steps of the process. It is also possible to visualize the changes taking place in objects in various phases of the control flow.

An example of activity diagram is shown below.



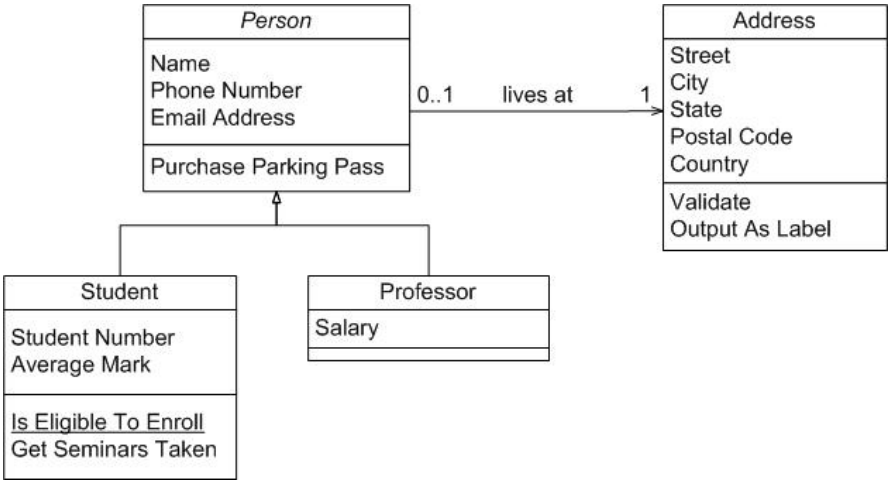
OBJECT-ORIENTED DESIGN

Object-Oriented Design (OOD) transforms the conceptual model created during object-oriented analysis into the implementation class and interface system. The output of object-

oriented analysis becomes an input for OOD. In fact, OOA outputs may not be complete in order to serve as OOD input. These processes can take place simultaneously and in practice, it is often the case - the results of one process can influence and direct the other. Analysis and design can take place in an incremental way, and all materials (documentation, diagrams, etc.) can be created step by step. The goal of OOD is to define how the system is to be created.

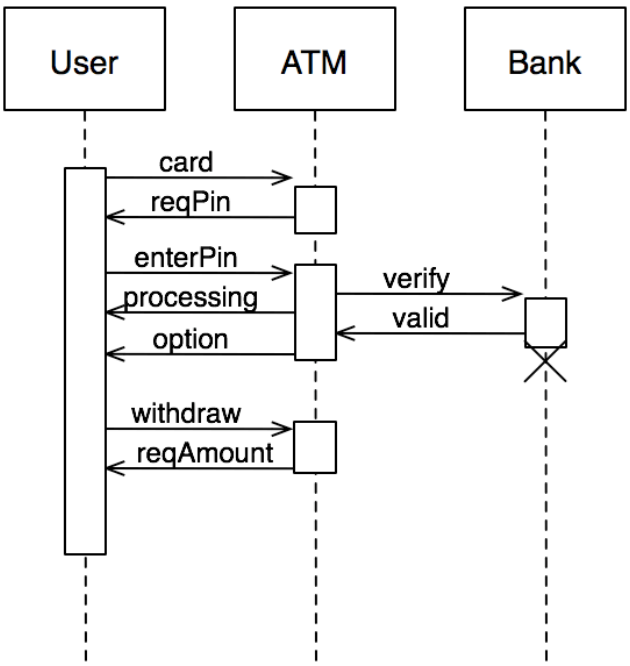
The class diagram is a key concept in all object-oriented methodologies and stage of object-oriented design. Often the class diagram corresponds to the entity-relationship diagram extended with additional elements. Class diagrams have methods assigned to the specified classes and show them in the form of certain graphic signs connected in a network of dependencies belonging to three categories: inheritance, association and aggregation.

An example of class diagram is shown below.



The sequence diagram is used to present the interaction between objects, including the messages that are sent between them during the design stage. The main purpose of sequence diagrams is modelling system behaviour in the context of use case scenarios. Sequence diagrams allow you to get an answer to the question of how communication between objects runs over time. In addition, sequence diagrams are the basic technique for modelling the behaviour of the system, which consists in the implementation of the use case.

An example of a sequence diagram is shown below.



OBJECT-ORIENTED IMPLEMENTATION

Object-Oriented Implementation consists in writing the source code based on the effects of the previous phases of software development (analysis and design). Implementation of the code is the software development stage, which usually absorbs most of the time and other resources. In this phase, we also deal with testing, debugging and implementation of the system.

CHOICE OF DEVELOPMENT PLATFORM

My choice for development approach is Object-Oriented Analysis and Design (OOAD). This methodology is data-oriented, it's fairly easy to use, understand and implement to build object-oriented programs. With a good understanding of the basis of OOAD, I believe that I am able to create in a short time and with a low-cost project that meets all requirements and will have every function described and desired quality.

DEVELOPMENT PLATFORMS

MICROSOFT C#

The C# language was developed by Microsoft and derives from the C/C++ family, although it also contains many elements known by Java programmers, such as the mechanisms of automatic memory management (garbage collector). Developers who use these programming languages on a daily basis will feel perfectly well in this environment. In turn for people who do not know C# it will not be difficult to control, and certainly much easier than such a popular C++.

The main creator of C# is Anders Hejlsberg, which is also a designer of the Delphi package produced by Borland, as well as Turbo Pascal. At Microsoft, Hejlsberg also developed Visual J++ environment. It all had an impact on the most newer product - C#, where we can see the clear relationships with both C and C++ as well as Java and Delphi (Object Pascal). (All About C# Programming, 2012)

C# is fully object-oriented and contains exception handling. It is also closely related to the .NET runtime environment. In the past, using C# language we could only write programs to use in Windows systems. Recently, Microsoft together with other companies decided to invest in other environments, including Mac OS, Linux, iOS and Android. Xamarin allows programming in C# on various platforms of portable devices, and Portable Class Libraries (PCL) is gaining wider and wider recognition. Microsoft ASP.NET is a new set of technologies for creating web applications that can run on the .NET Framework or .NET Core - a new small, fast and open cross-platform runtime environment. (Stackify, 2017)

The most popular working environment for the C # .NET platform is Visual Studio for Windows. There are also other popular IDEs for other operating systems, such as Visual Studio for Mac - as the name suggests we can program on MacOS, Visual Studio Code - an open source code editor for Linux and MacOS and JetBrains Raider - new cross-platform .NET IDE.

JAVA

Java programming language was designed and implemented in the laboratories of Sun Microsystems in Mountain View (California) under the leadership of James Gosling and Patrick Naughton. James Gosling is also the author of the emacs program for UNIX.

James had the task of preparing a new programming language, more suitable for application software than the existing solutions, such as C or C++. Programs written in C/C++ are compiled for a specific processor so that if you want to adapt them to a new processor, you will usually need to recompile them. In Java, we are dealing with an intermediate stage, i.e.

the program is a set of commands to be performed on a virtual machine, and this program interprets commands and translates them into lower-level instructions. For the Java virtual machine (JVM) there were made many more languages, which are also compiled into the intermediate code (bytecode) just like Java and are run on the JVM. These languages are Scala Groovy and Kotlin. (TheServerSide, 2016)

Java from the beginning is a fully object-oriented language and what is not object-oriented is basically just simple types like int or char, but even they have protective types that increase their usability.

Java has many popular IDEs like NetBeans, Eclips, Intellij IDEA and Android Studio which is actually built on JetBrains Intellij IDEA. It is also (with Kotlin since 2017) an official Google supported language for Android developers. (The Verge, 2017)

PYTHON

In the early 1990s, Python was created by Guido van Rossum - as successor to the ABC language, created at Centrum Wiskunde & Informatica (CWI; English: "National Research Institute for Mathematics and Computer Science in Amsterdam"). Van Rossum is the main creator of Python, although a significant contribution to its development comes from other people. Because of the key role that van Rossum plays in making important design decisions, it is often referred to as "Benevolent Dictator for Life" (BDFL).

The name of the language does not come from the animal, as we might suppose. Python comes from a comedy series broadcast by the BBC in the 1970s. This series is called "Monty Python's Flying Circus". (Packt, 2015)

Python is a high-level general-purpose programming language with an extensive package of standard libraries, the idea of which is the readability and clarity of the source code. Its syntax is characterized by transparency and brevity. (Python Software Foundation, 2018)

Python supports various programming paradigms: object-oriented, imperative and to a lesser extent functional. It has a fully dynamic type system and automatic memory management, being similar to Perl, Ruby, Scheme or Tcl. Like other dynamic languages, it is often used as a scripting language. Python interpreters are available on many operating systems. (ThoughtCo., 2017)

Python is being developed as an Open Source project and it is managed by the Python Software Foundation, which is a non-profit organization.

CHOICE OF DEVELOPMENT PLATFORM

I have decided to choose Java as my development platform. I have been using Java for a few years and I'm familiar with the code syntax and frameworks. I know how to create backend – server side using REST architectural style, map an object-oriented domain model to a

relational database with Hibernate ORM, build client side with Spring Boot MVC and Spring Security authentication and access-control framework. Also, Java is the only reasonable choice to create a native Android application. Java has a great community support, can be easily connected with Google Firebase development platform and thanks to Java Virtual Machine I can run the application in every environment which is very important as all the backend and web client side will be working on DigitalOcean Droplets cloud platform.

BIBLIOGRAPHY

All About C# Programming, 2012. *History of C# Programming.* [Online]
Available at: http://aboutcsharpprogramming.blogspot.co.uk/2012/09/history-of-c-programming.html
[Accessed 19 03 2018].

Borg, S. & Williamson, S., 2016. *Test-Driven Development.* [Online]
Available at: https://mva.microsoft.com/en-us/training-courses/testdriven-development-16458
[Accessed 19 12 2017].

Kroenke, D. & Auer, D., 2009. *Database Concepts.* New Jersey: Prentice Hall.

Miguel, J. S., 2017. *KISS, YAGNI & DRY, 3 Principles to Simplify Your Life as a Developer.* [Online]
Available at: https://www.itexico.com/blog/bid/99765/software-development-kiss-yagni-dry-3-principles-to-simplify-your-life
[Accessed 19 12 2917].

Packt, 2015. *A Brief History of Python.* [Online]
Available at: https://www.packtpub.com/books/content/brief-history-python
[Accessed 19 03 2018].

Python Software Foundation, 2018. *What is Python? Executive Summary.* [Online]
Available at: https://www.python.org/doc/essays/blurb/
[Accessed 19 03 2018].

Stackify, 2017. *.NET Core vs .NET Framework: How to Pick a .NET Runtime for an Application.* [Online]
Available at: https://stackify.com/net-core-vs-net-framework/
[Accessed 19 03 2018].

Stair, R. & Reynolds, G., 2001. *Principles of Information Systems.* Boston: Course Technology.

Sutherland, J., 2004. *Agile Development: Lessons Learned from the First Scrum.* s.l.:s.n.

Takeuchi, H. & Nonaka, I., 1986. *The new new product development game.* s.l.:Harvard Business Review.

The Verge, 2017. *Google is adding Kotlin as an official programming language for Android development.* [Online]
Available at: https://www.theverge.com/2017/5/17/15654988/google-jet-brains-kotlin-programming-language-android-development-io-2017
[Accessed 20 03 2018].

TheServerSide, 2016. *Java.* [Online]
Available at: http://www.theserverside.com/definition/Java
[Accessed 20 03 2018].

ThoughtCo., 2017. *What Is Python?.* [Online]
Available at: https://www.thoughtco.com/what-is-python-2813564
[Accessed 19 03 2018].