
CA2: Object Oriented Software Engineering

National College of Ireland – <<Internet of Things>>

Michal Borkowski

Student no 17164532

2019



Table of Contents (Green – group assignments; blue – individual)

Project Background and Description.....	3
1. Identify the actors.....	3
2. Construct a Use Case Model.....	4
3. Describe in detail any use case from the use case model. The use case must contain an alternate flow.....	4
4. Create a Project Plan to deliver your application. The plan must include a minimum of 3 nesting levels and include all the major tasks and deliverables. Describe how you monitor your project plan as you develop your project. Use Microsoft Project to produce the Gantt chart and any diagramming tool of your choice to produce the Work Breakdown Structure (WBS)	7
5. Create a conceptual class diagram modelling the architecture of the proposed system. The conceptual class diagram should demonstrate the use of many of the following: attributes, relationships, navigability, association class, multiplicity and composition.....	9
6. Create a glossary that lists and defines all the terms that require clarification.....	9
7. Draw a System Sequence diagram from the conceptual class diagram.....	9
8. Develop Contracts for a minimum of two of the system operations in the system sequence diagram.....	11
9. Draw Communication diagrams based on the above 2 contracts. The communication diagrams should demonstrate the use of design patterns.....	11
10. Presentation (how well does the package of models look?)	12
11. Use of software chosen.....	12
12. Put together a Testing Plan outlining how you propose to validate the application and verify that it is free of defects + Game Summary.....	13

JULY 7 2019

National College of Ireland

Authored by: Michal Borkowski

Student No: 17165432

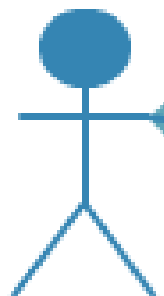
Project Background and Description

Using the Use Case Diagram and Project Plan which was created in CA1, in this work are presented diagrams to extract what system does and how. Reports and descriptions are made to support of the Project Plan to most accurate business trend.

1. Identify the actors.

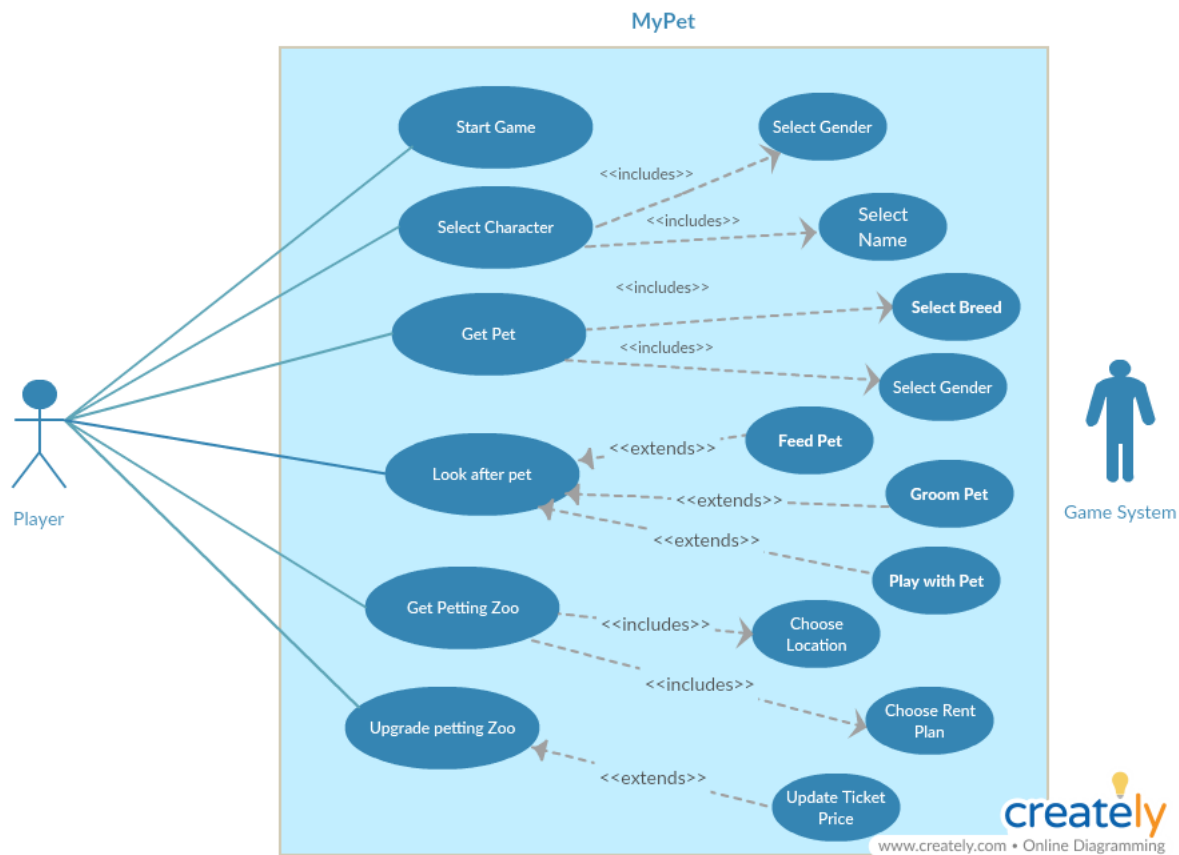
The following script creates a primary actor which is outside the game system. In the game “MyPet” the actor is a “The Rabbit” chosen player’s character with special ability. There are no secondary actors within the game.

How primary actor is represented in diagrams:



Player

2. Use case diagram.



3. MICHAL BORKOWSKI's use case.

"FEED PET"

Purpose

After feeding player can use character as an AI character as an opponent to play with him different games.

Scope

The scope of the use case is to show how works the feeding pet within the System.

Actors

Primary Player.

Description

This use case describes how to feed chosen pet after selecting character and after selecting pet.

[Flow Description]

Precondition

The player should have already completed previous levels: started the game, logged, selected character, selected the human pet. To feed the human pet player first needs to have one. Feeding is the cardinal condition for having next development of human pet in future. Without proper feeding human pet cannot exist.

Activation

Player is pushed to feeding (look after his pet) after finishing previous three stages. This use case starts when player launched the game, has already chosen character of himself (from pets – select character use case) and when player has decided which human pet he wants to play with. So, this use case cannot be started separately from mentioned others. Use case is activated when human pet begins to live in the game and asks for food supplies.

Main Flow.

0. The Player log in the System, access() game.
1. The System greets the Rabbit (player) in new level of the game;
2. The System tutorial explains the extra features of chosen player (the Rabbit) – ability to speed up loading processes.
3. The System tutorial explains the extra features of chosen character (Antarctica male) – ability to slow down the metabolism.
4. The System prompts player to choose character chooseCharacter();
5. The Player choses the character;
6. The System loadingCharacter()
 - o 6a. The System recommends to the Rabbit (player) tutorial for feeding human pet (A1);
7. The Rabbit (chosen character for player) clicks on Antarctica male (chosen player`s human pet) activateCharacter().
8. The Antarctica male food bar shows the current energy feeding status (low), with warning: only able to walk and harvest small plants (energy bar needs to be fulfilled with fat, carbohydrates and protein as human pet needs proportionally);
9. The Rabbit (player) use character to seed the field with plants;

-
10. The Rabbit (player) while waiting for plants clicks on Antarctica male to push him to harvest fruits from nearest forest;
 11. The Rabbit (player) uses his internal feature to speed up plants growing speedGrow();
 12. The Character eats collected berries from forest;
 13. The Rabbit (player) tries again to speed up the bar for growing bars (no sufficient EnergyBar status) and System informs: status energy bar is changed, level of carbohydrates increased 15%;
 14. The Rabbit (chosen character for player) clicks on Antarctica male (chosen player's human pet) activateCharacter();
 15. Unlocked ability: The Antarctica male is now able to climb and The Rabbit (player) clicks on Antarctica male and uses him to climb on three to collect bird eggs;
 - o 15a. The Rabbit used Antarctica male not properly and damaged eggs (E1);
 16. The system tutorial displays information to the Rabbit (player) that every 15-minutes Antarctica male loses 1% of each carbohydrates, fat, and protein levels;
 17. The Rabbit (player) feeds the Antarctica male with the bird eggs;
 18. The Character has eaten collected eggs;
 19. The Antarctica male energy bar protein level has grown to 20 % and fat bar grown 10%;
 - 20. The Antarctica male becomes an explorer: promoteExp();**
 - o 20a. System informs Player - character reached new level and has new abilities: storingEggs(), buildFarm() (E2);
 21. LoadingCharacter();
 22. The Player useCharacter new energy to hunt for meat setTrap();
 23. Character hunts for reptiles;
 - o 23a. Character uses incorrect trap for hunting (E3);
 24. The Rabbit (player) feeds the Antarctica male with fresh meat: feedPet();
 25. The Antarctica male energy bar proteins level has grown 20 %;
 26. The Antarctica male slowMetabolism() down to keep EnergyBar at high level;
 27. The Rabbit (player) decided to feed the Antarctica male with planted fruits;
 28. The System informs: status energy bar is changed, level of carbohydrates increased 15% EnergyBar checkRequirement();
 - 29. The Antarctica male becomes a trader: promoteTrader();**
 30. LoadingCharacter();
 31. System informs player new ability of character is unlocked: buildFarm(), buyFridge();
 32. The player activateCharacter() to sell collected goods (plants) on the market;
 - o 32a. Character setTrap() for meat;
 - o 32b. Character climb() three for eggs;
 - o 32c. Character buildPlantation() for fruits;
 33. The Antarctica male put products on the market and trade() and The Antarctica male lost half of energy from EnergyBar for hunting (setTrap());
 34. The Character receives new good for trading;
 - o 34a. The Character does not receive new good for trading (A2);
 35. The System congrats to player and informs due to achievement trader new game with character
 36. is unlocked: Monopoly;
 37. The System saving changes and loading main menu: saveChanges();
 38. The System informs Player: playing with Character is unlocked now;

Alternative Flow

(A1)

1. The Rabbit (player) clicks “yes” button for tutorial;
2. System presents all relevant information for feeding human pet;
3. The Rabbit (player) clicks “OK” button and quits the tutorial;
4. Screen backs to main flow of the game to point 6.

(A2)

- a. The Character is robbed by another human pet;
- b. The Character lost 70% energy to escape from market.
- c. The Player returns to point 33 but needs to start from Explorer level again.

Exceptional Flow

(E1)

- a. The Rabbit feed the Antarctica male only with fruits;
- b. The Antarctica male is lack of proteins and fat after one hour;
- c. The Antarctica male collapsed on the ground and died;
- d. The System informs the Rabbit (player) the game is over.

(E2)

- a. The Player tries to enter cheat code;
- b. The System ban Player for cheating;
- d. Resume at the main menu and unlock for Player due to high EnergyBar status: grooming pet.

(E3)

- a. The Rabbit (player) used an Antarctica male to set a trap for too big reptile;
- b. The Antarctica male was eaten by reptile;
- c. The System display a widow to announce the “game is over”;
- d. The System returns player to main menu.

Termination

The feeding program for human pet will only be closed after decision of the player when character have reached particular level (explorer, trader) or when character achieved the highest level of satisfaction after eating required number of valuable products. When the feeding is reach at any of three levels player can play with character different games accordingly to his progress in energetic balance.

Post Condition

The feeding must be repeated constantly - every 15 minutes character loses energy bar level.

4. Work Breakdown Structure (WBS) – Gantt Chart*.

* Two weeks before submission deadline one Team Members left the course, therefore we decided to focus on individual parts due to lack of time.

MyPet - Project Plan		
Team: Amy , Michael , Suheab		
	Start: 01/01/2019	Deadline: 30/09/2019
Activity	Due Date	Activity Number
1,Research & Information Gathering	2/4/2019	1
1.1,-Game idea		
1.2,-Market research		
1.3,-Software code initial plan		
1.4,-Requirements gathering		
1.4.1,-Interview target audience with game idea		
1.4.2,-Take feedback from target audience(gamers)		
2,Planning/Initializing	3/21/2019	2
2.1,- Initial Project Meet		
2.2,- Define Project Scope		
2.3,-Obtain and Identify Project Sponsors		
2.4,- Draft Software Scheme		
2.4.1,Create Use Cases		
2.4.2,-Define Actors & Use cases		
2.4.3,-Create Conceptual Diagram		
2.4.4,-Draft animation/graphics scheme		
3,- Review	4/4/2019	3
3.1,-Review graphic design requirements		
3.2,-Define budget for graphic designers		
3.2.1,-Review resources for graphical requirements		
3.3,-Review software requirements		
3.3.1,-Review budget for software developers and testers		
4,Analysis	5/6/2019	4
4.1,-Define project budget		
4.2,-Develop preliminary data and process models		
4.3,-Develop delivery roadmap		
4.4,- Secure project sponsorship		
5,Development	8/5/2019	5
5.1,- Develop functional specifications		
5.2,-Develop overall colour scheme		
5.3,-Develop app logo and icon		
5.4,-Develop in-app GUI layout		
5.4.1,-Define layout of navigation pane		
5.4.2,-Define icons for each use case		
5.5,-Develop Objects		
5.5.1,-Develop Character design		
5.5.1.1,-Colour Code gender selection buttons		
5.5.1.2,-Develop Character name field with String limitation		
5.5.2,-Pet		
5.5.2.1,-Colour Code gender selection buttons		
5.5.2.2,-Develop breeds		
5.5.2.3,-Develop name field with String limitation		
5.5.3,-Inventory		
5.5.3.1,Develop brush		
5.5.3.2,Develop ball		
5.5.3.3,Develop food		
5.5.4,-Zoo		
5.5.4.1,-Develop rent time		
5.5.4.1,-Define and design locations		
5.5.5,-Ticket price		
5.5.5.1,-Ensure functionality for future updates to price		
5.5.6,-Develop JFrame pop-up window design		
5.6.1,-Develop JFrame pop-up window design		
5.6.2,-Define error message scenario		
5.6.3,-Define error message verbiage		
6.1,-Develop prototypes		
6.2,-Develop mockup game		
6.3,-Develop remaining graphics and animations		
6,Testing	9/9/2019	6
7.1,-Developer Testing		
7.2,-Testing Evaluation		
7.3,-Black Box testing		
7.3.1,-Black Box testing evaluation		
7.4,Development & Testing Review		
7.4.1,-Bug fixing		
7.4.2,-Code improvements from black box testing		
7. Go Live! - Delivery	9/12/2019	7
9.1,-User feedback		
9.2,-Post implementation review		
8.Completion & Maintenance	9/30/2019	8
10.1,-User Support		
10.2,-Quarterly Review		
10.2.1,-Code improvements for future delivery		

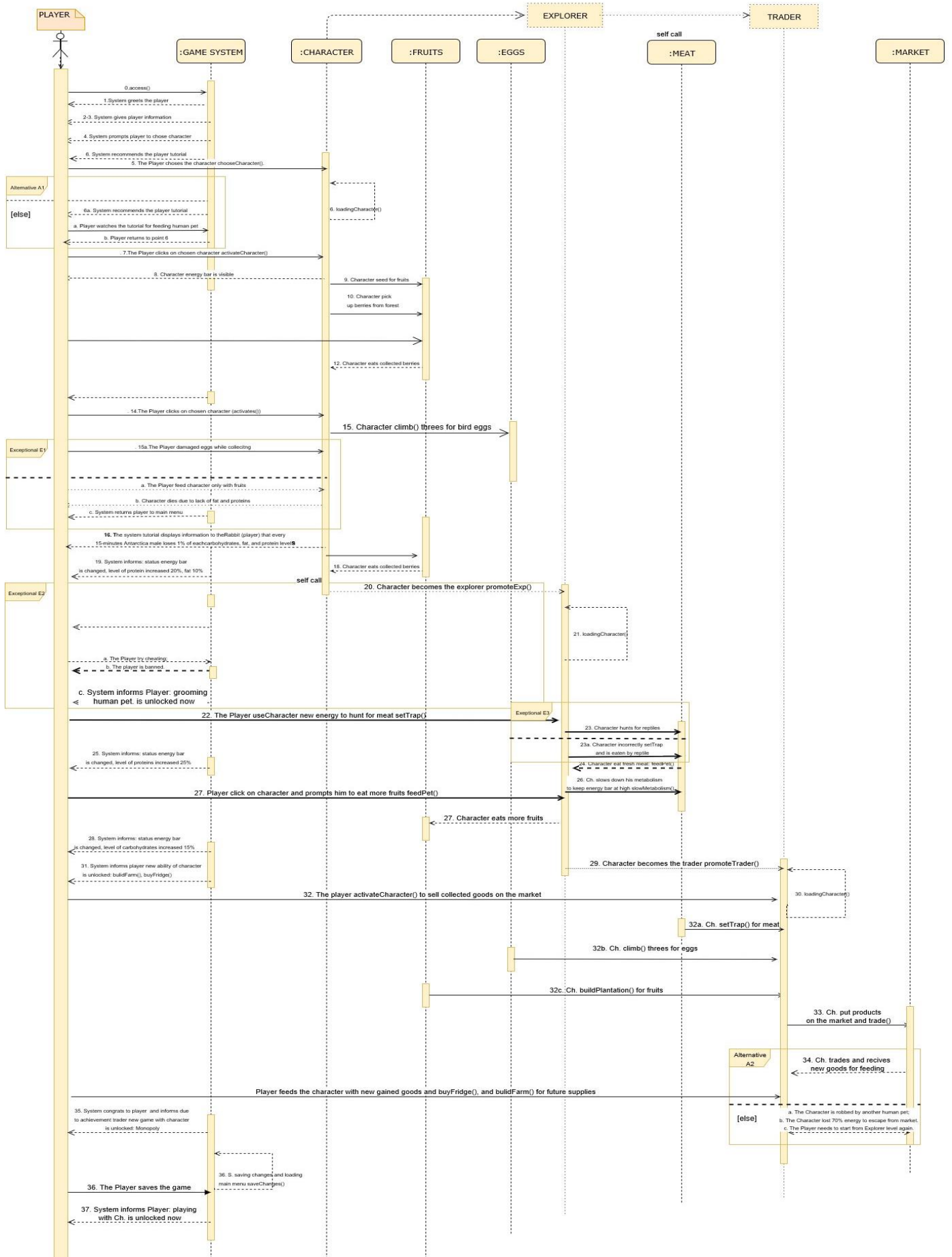
4.

Michał Borkowski 17165432

Object Oriented Software Engineering 2019



7. System Sequence diagram.



8. Contracts.

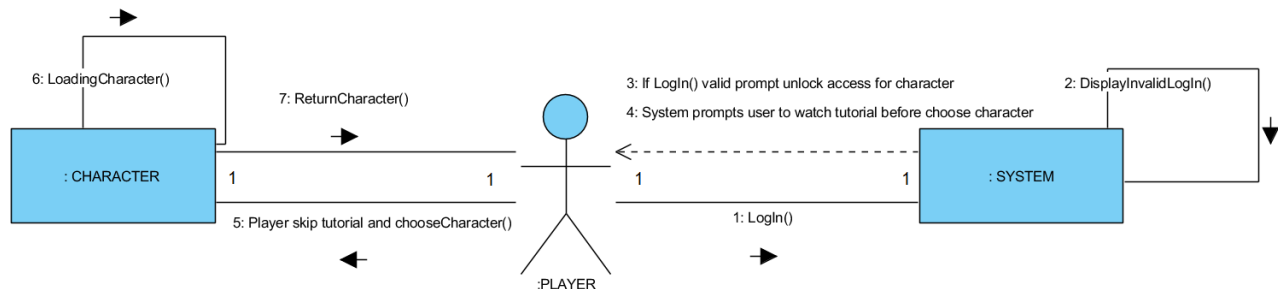
Contracts:

Name	1. Player choose the character: chooseCharacter()
Responsibilities	When player has started the game, logged in his account then can choose desired character. New character will be assigned to the Player
Type	System
Pre-Condition	Player must have been previously been logged in
Post-Condition	The planned structure is complete via 'build' action.

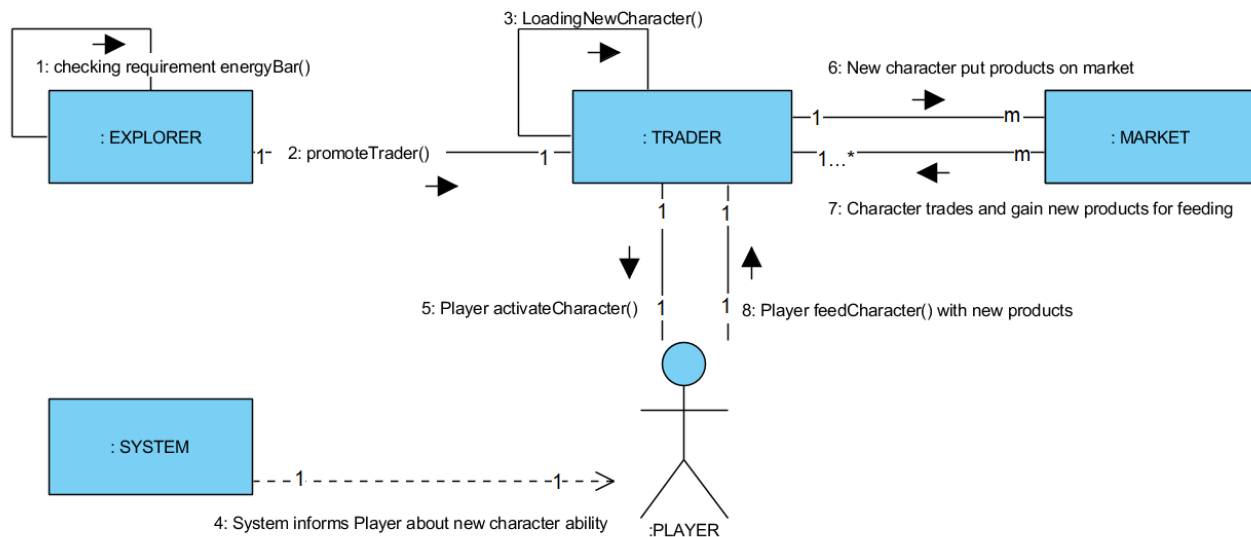
Name	2. Unlock new character: promoteTrader()
Responsibilities	Function unlocks new upgraded version Character with new abilities
Type	Game
Pre-Condition	Player`s character must be at above level of an Energy Bar to be able to climb threes and gather bird eggs.
Post-Condition	Player must remember to feed regularly character – unless character will be returned to previous version

9. Communication diagrams.

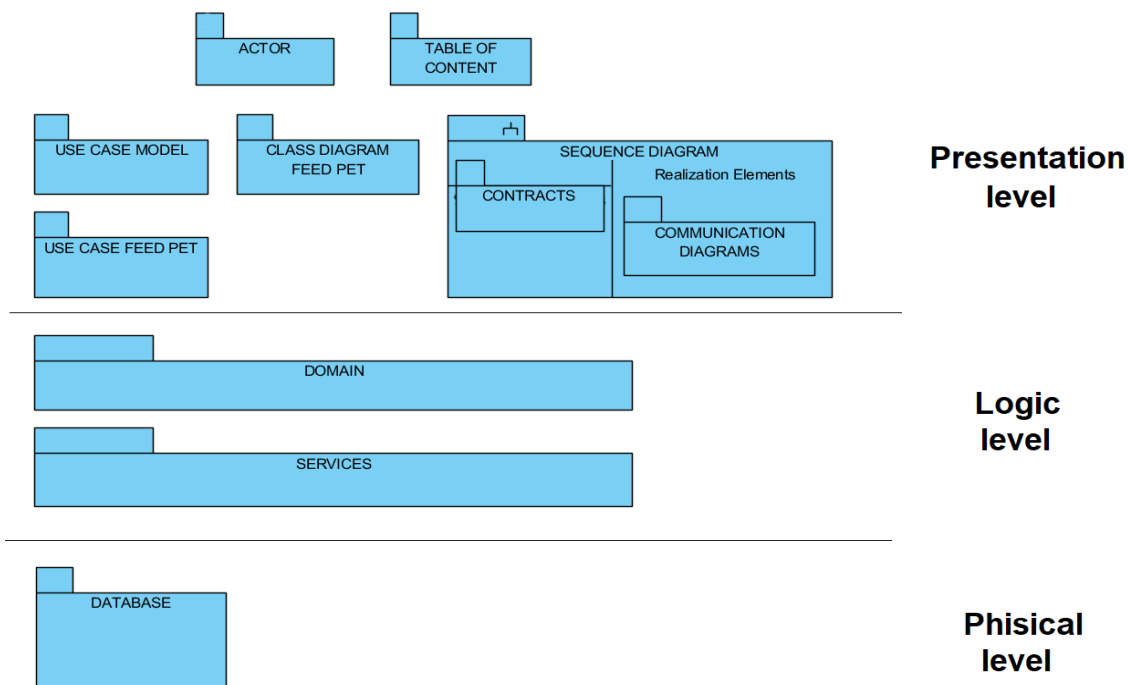
1. Player choose the character: chooseCharacter() scenario



2. Unlock new character: promoteTrader() scenario



10. Presentation.



11. Use of software chosen

Use Case Model: [www.createely.com] online;

Class Conceptual Diagram: [www.LucidChart.com] online;

Sequence Class Diagram: [www.draw.io] online;

Communications Diagrams, Presentation: Visual Paradigm software – on device;

Presentation base document: Microsoft Word, PDF – on device.

Use Case Model was built in easy-to-use cloud based online software createely.com. The Project Plan was created using Microsoft Excel as it is handy to manipulate while making progress on its statistics view construction. Lucid Chart software was used for Class Conceptual diagram as a first choice due to its intentional interface. Lucid Chart is very user-friendly easy-to-use free trial version and straightforward, but our trial after 30 days version of Lucid Chart was expired. Therefore, next choice program – from drawo.io was used for the System Sequence Diagram and at the final stage Communication Diagram was created in Visual Paradigm downloaded from official website launched in trial version as well. There is very interesting feature in Visual Paradigm – synchronization of Communication and Sequence Diagrams, which easily rebuilt one diagram into another based on used components. Microsoft Word is a program to implement document presentation along with submitted PDF file.

12. Testing Plan

We have chosen a functional tests method better known as a black box tests for a main testing plan. Black box test gives user who has no access to information about the structural construction of the program capability to test an in-built system external behavior and usability. Unique sequences of testing performed by non-programmers (called *attackers*) who reside outside the program during implementing code. Usually black box testing applies software on respondents who do not have any knowledge of software coding. Without awareness of the internal structure of the program, (end-) user focuses on functional rather than structural features of the final product. However, program should be acknowledged in accordance with the documentation.

Aforementioned arguments mean – this method can work independently and gives the better chance of detecting problems in functionality for end-users. From the

other hand this method does not usually provide a sophisticated data on the source of the errors in the main flow software. Due to this subtle difficulty for more reliable testing in various information systems, we decided to support our testing with that so-called white box testing sc. structural tests to bring complexity to our method. White-box testing - is also known as white or glass box tests. They consist in testing the program by providing such data on the input so that the program passes through each implemented path. These rules are defined by the criteria for coverage of all loops and all conditions. White box tests cannot show the lack of implementation of the function that the target system should have. However, they thoroughly check the operations performed in the implemented methods (specialist checking) and combined with the black box testing method return solid results and data before implementing product on the market.

Game Summary

The game is intended to be a mobile app game for Android operating systems programmed in Java. Java was chosen as it is versatile and can run on all platforms (for future iOS development plans), holds powerful IDEs which assists with error handling and is an object orientated language. The idea behind the game is that the players learn to care for their pets who are humans. Changed role is the core idea of the game. Players can choose between many anthropologic categories and can select the gender of their human pet.

All navigation panes are based on the right. This navigation pane is constant and remains even in play mode. All navigations features can be clicked or tapped, once one is tapped or clicked it will drop down into a list, minimizing the other features.

Interface:

