Assignment 3 - Group 23

Exercise 1

1. Requirements extensions:

    Must haves:

    - The game shall have three different types of aliens
    - One type shall shoot bullets that move at a low speed, is killed with 1 spaceship bullet and is worth 10 points
    - Another type shall shoot bullets that move at a medium speed, is killed with 2 bullets and is worth 20 points.
    - The last type shall shoot bullets that move at a high speed, is killed with 3 bullets and is worth 30 points.

    Should haves:

    - The game shall have multiple levels.
    - The game shall advance to the next level when there are no aliens left.
    - The game shall end when there are no more levels left.
    - The game shall have more than 1 level.
    - Each level shall increase the difficulty.

NOTE: We did not have enough time to fully complete the implementation of the levels but we left the requirements and CRC card so we can complete the implementation and UML later.

2. CRC cards & UML

    Aliens:

Class name: Alien	
Superclass: /	
Subclasses: AlienType1, AlienType2, AlienType3	
Purpose:	**Collaborators:**
Superclass to provide the game with multiple types of aliens	
Shoot bullets downwards	Bullet
Disappear when hit by spaceship bullet	Game
Move vertically and downwards	Game

Class name: AlienFactory	
Superclass: /	
Subclasses: /	
Purpose:	**Collaborators:**
Provide the Game class with the right aliens	AlienType1, AlienType2, AlienType3, Game

Class name: AlienType1	
Superclass: Alien	
Subclasses: /	
Purpose:	**Collaborators:**
Appear at the beginning of the game	AlienFactory
Shoot bullets downwards at low speed	Bullet
Disappear when hit by spaceship bullet	Game
Move vertically and downwards	Game

Class name: AlienType2	
Superclass: Alien	
Subclasses: /	
Purpose:	**Collaborators:**
Appear at the beginning of the game	AlienFactory
Shoot bullets downwards at medium speed	Bullet
Disappear when hit by spaceship bullet	Game
Move vertically and downwards	Game

Class name: AlienType3	
Superclass: Alien	
Subclasses: /	
Purpose:	**Collaborators:**
Appear at the beginning of the game	AlienFactory
Shoot bullets downwards at high speed	AlienFactory
Disappear when hit by spaceship bullet	Bullet
Move vertically and downwards	Game

Levels:

Class Name: Levels	
Superclass(es):	
Subclasses:	
Purpose	Collaborators
Multiple levels	Game
Move to next level	Game
Increase difficulty	Alien
End game when no levels left	Game

Exercise 2

a. Singleton pattern LogFile class:

1. Description

   To log all the things that happen in the game there is a LogFile class with different write functions for different types of log messages. In this LogFile class is the singleton pattern implemented, this pattern makes sure that there can only be one instance of the class. This ensures that there is only one log file and that all the log messages are written to that same log file. The implementation of the singleton pattern in the LogFile class is accomplished by creating a method which creates an instance of the LogFile class if there isn't one already and return this LogFile class. If there is already an instance of the LogFile class that instance is returned. The constructor of the LogFile class is also private so it can only be called from within the LogFile class.

2. Class diagram

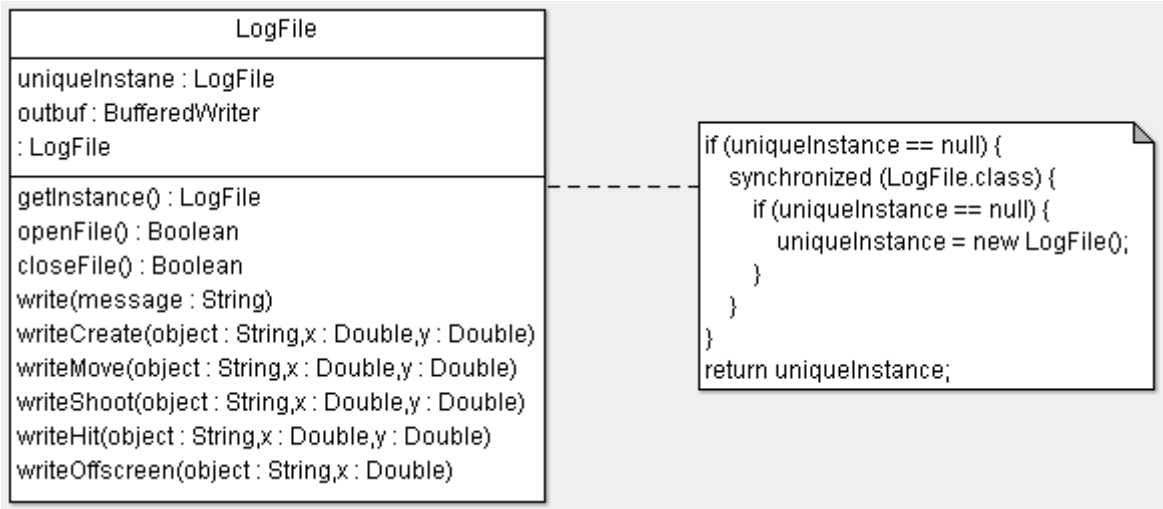

Figure 1: Class diagram of Singleton pattern in LogFile class

3. Sequence diagram

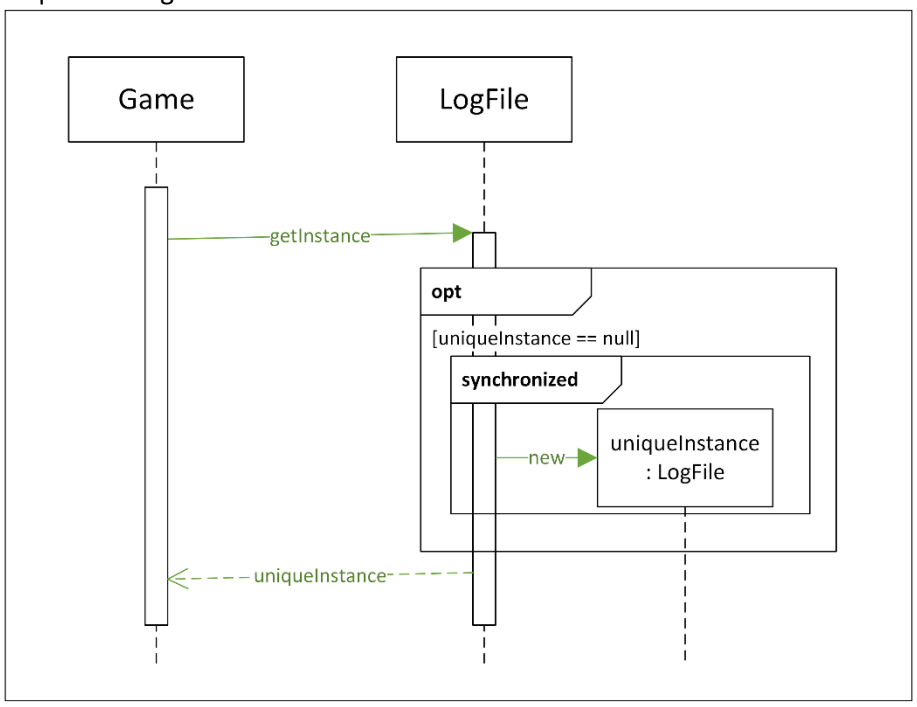

Figure 2: Sequence diagram of Singleton pattern in LogFile class

b. Factory pattern Alien class:

1. Description

   We decided to implement multiple types of aliens and multiple levels. So not only do we need to be able to create multiple types of aliens we also need to create them during run time (we want to be able to switch between levels almost seamlessly). That is why we decided to use the factory pattern to create the aliens. Because with this new factory pattern the object creation is encapsulated and if we were to add another alien type later in the project we simply have to add it to the factory class. If we didn't have this factory class, we have to change the game class (were all the aliens are created) and because the game class has a lot more dependencies this would be a lot more work. The implementation in our code is simple, we have three alien types which are all children of the abstract class alien. And there is an alien factory class that handles the alien object creation. So if the game class needs a new alien object it calls the alien factory class which will call the super class to create the specific alien subtype this newly created alien in the factory class is then returned to the game class.

2. Class diagram

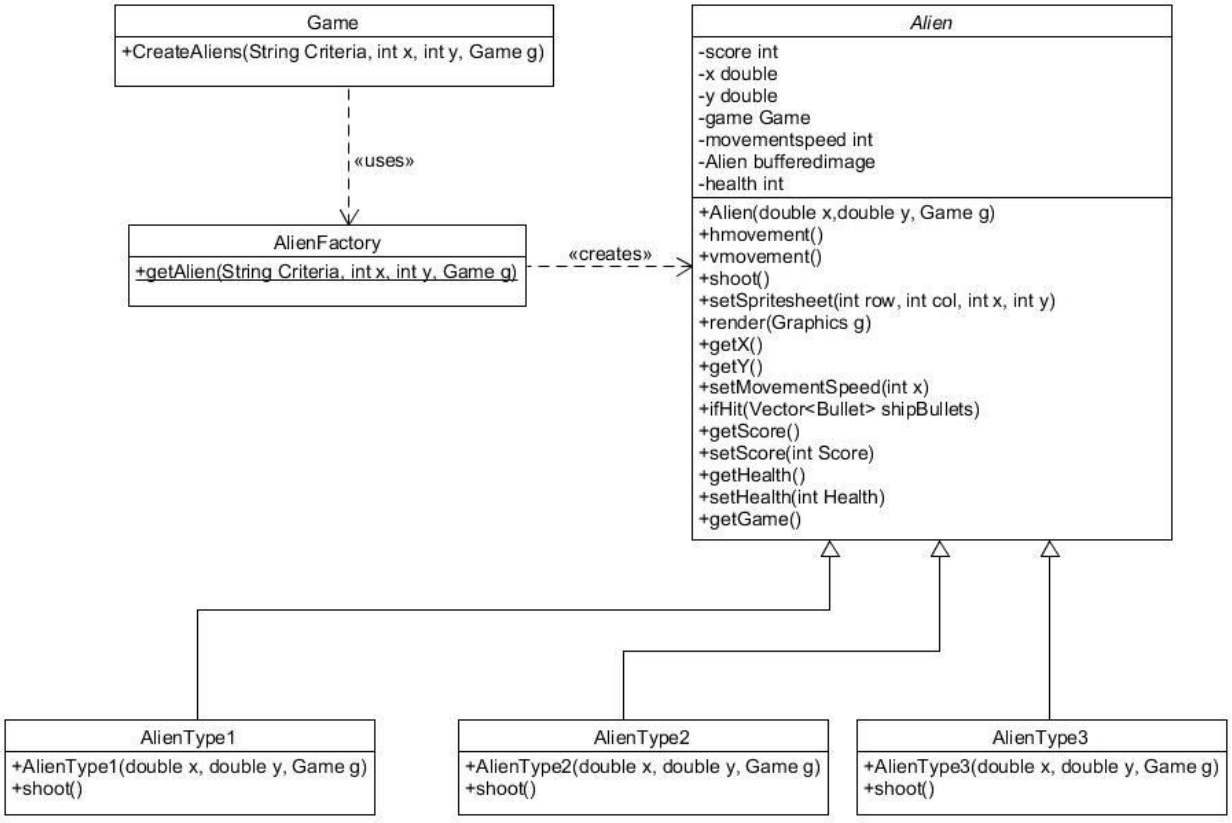

Figure 3: Class diagram of Factory pattern for the aliens

3. Sequence diagram

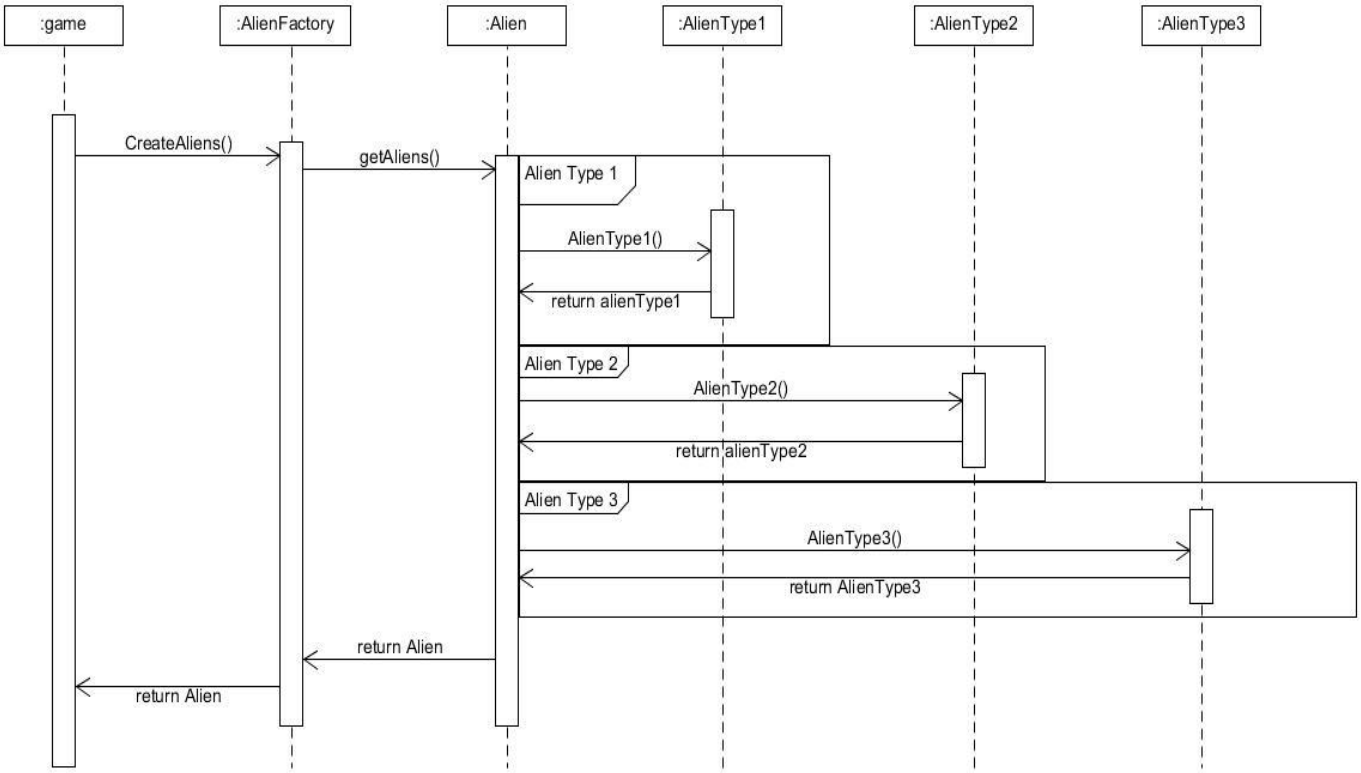

Figure 4: Sequence diagram of Factory pattern for the aliens

Exercise 3

1. Explain how good and bad practices are recognized.

In the context given by the paper, a good practice can be recognized when the project performance on both cost and duration is better than the average cost and duration corrected for the applicable project size.

Bad practices can be recognized when the performance on both cost and duration is worse than the average cost and duration corrected for the applicable project size.

The better and worse than average cost, effort and duration performance also defines the success and failure of a project.

2. Explain why Visual Basic being in the good practice group is a not so interesting finding of the study.

Because the visual basic project environments on average are less complex than others, and due to that end up in the Good Practice quadrant more often (similar to a false positive), and because the study could not identify the cause behind the fact that Visual Basic turns out to be a success factor.

3. Enumerate other 3 factors that could have been studied in the paper and why you think they would belong to good/bad practice.

This three factors belongs to good practice:

- Project complexity: By considering this factor, the amount of false positives in good and bad practices can be reduced and the metric Cost / duration can be adjusted to show a more realistic panorama of a project (A complex large project can cost more and last longer than a simple large project, but that does not make the latter a success /good practice project).
- Customer Satisfaction: As good practices also mean the success of the project, it needs to be measure in which degree the project is meeting the expectative of the customer.
- Profit margin: With this factor, the actual impact in cost can be measure, since we can have project classified as good practices with a low margin profit and a bad practice project (that was costlier than the average) but with a bigger profit margin.

4. Describe in detail 3 bad practice factors and why they belong to the bad practice group.

- Inexperienced team: Most members of a team are graduates, or inexperienced in disciplined software development. It is a bad practice because typically during a resource allocation, team members are picked based on availability or budget, not necessarily on skill set as a consequence newly graduates who are relative cheap (compared to senior programmers) are picked for large and complex projects.
- Security: Refers to until what extend an application is protected against vulnerabilities. It belongs to the bad practice group, because really often developers don't test their software against attacks due to budget and time constrains. The security and quality measures are often disregarded in the development process by the managers and business stakeholders.
- Migration: Often refers to the change or upgrade of an application into another. It is considering a bad practice mainly because often they include the need to get a "quick fix" to an application as well as utilize migration technology that is available. While it can be beneficial to utilize automated migration tools, the tendency is to assume that these tools are "magic bullets" and to miss the overall benefits that can be achieved with migrating legacy applications. Lack of documentation, obsolete technology and low knowledge about the new technology are also reason for bad practice.