BUSINESS ACADEMY SOUTHWEST

Compulsory Assignment #1: Pong

Hand-in by:

Ahmed Mohamad Issa Asfour

Davide José da Silva Jorge

Katja Tamstrup Strunck

Luís Carlos Pacheco Cartucho

Friday, 20th September 2024

Table of Contents

1. Introduction

1.1. Background

After three weeks of lectures in all the first semester courses in Computer Science, namely Software Construction 1, Software Design 1 and IT and Business Development, a first compulsory assignment has been given to assess and further cement the knowledge, methods, use of required tools and expected real-life working experiences presented by the teachers.

For the compulsory assignment number one, a *Pong* video game is to be developed on Greenfoot IDE using Java programming language. Apart from the video game and what is to be considered the final product, every aspect of its development as a team/group is to be elaborated and well documented. This applies to every step of the process, from the initial idea and proposition to the game itself.

This assignment will be done by Group 1 of the International Class in Computer Science at Erhvervsakademi Sydvest (Business Academy Southwest) in Esbjerg during week 38 of the year 2024. The forming body of Group 1 of the International Class in Computer Science is composed of the following students: Ahmed Mohamad Issa Asfour, Davide José da Silva Jorge, Katja Tamstrup Strunck and Luís Carlos Pacheco Cartucho.

1.2. Product Vision

To get to the final product a list of mandatory tasks must be fulfilled. These tasks apply to the game and how it should work. The first of the given mandatory tasks, for example, determines the game's basis by stating that it should be a single-player game with a paddle at the floor operated by the player by pressing the “left” and “right” keys on the keyboard and a ball that bounces off the ceiling and walls of the game screen.

There are also optional tasks to further enhance the overall gaming experience and make it a version of our own.

We are set however to achieve, firstly and above all, the best version of the game that ticks/checks all the boxes in The Mandatory Tasks list and later consider the Optional Tasks list and any other features or bonuses in the game.

2. Pre-Game

2.1. Project Organization

After first acknowledging the group, its name and its members, the first meeting and discussion that took place regarding the given compulsory assignment, Pong, served to level our expectations, understand our individual strengths and weaknesses (i.e. the fields and subjects provided by our teachers that we are more or less comfortable with) and start laying the ground work.

We first worked on our Working Agreement and set in writing the Team Agreement and Ground Rules, a project description, the products that would spawn out of it and the necessary research to get to the final product. We also introduced a Proposed Timeline with all the necessary parts/steps towards the finish line and who within the group is primarily responsible for them respectively.

A copy of the Working Agreement can be found in Appendix 6.1. - Working Agreement.

2.2. Overall Project Schedule

The following Gantt Diagram and Work Breakdown Structure (WBS) depict the project's overall schedule and its products and major deliverables, respectively.

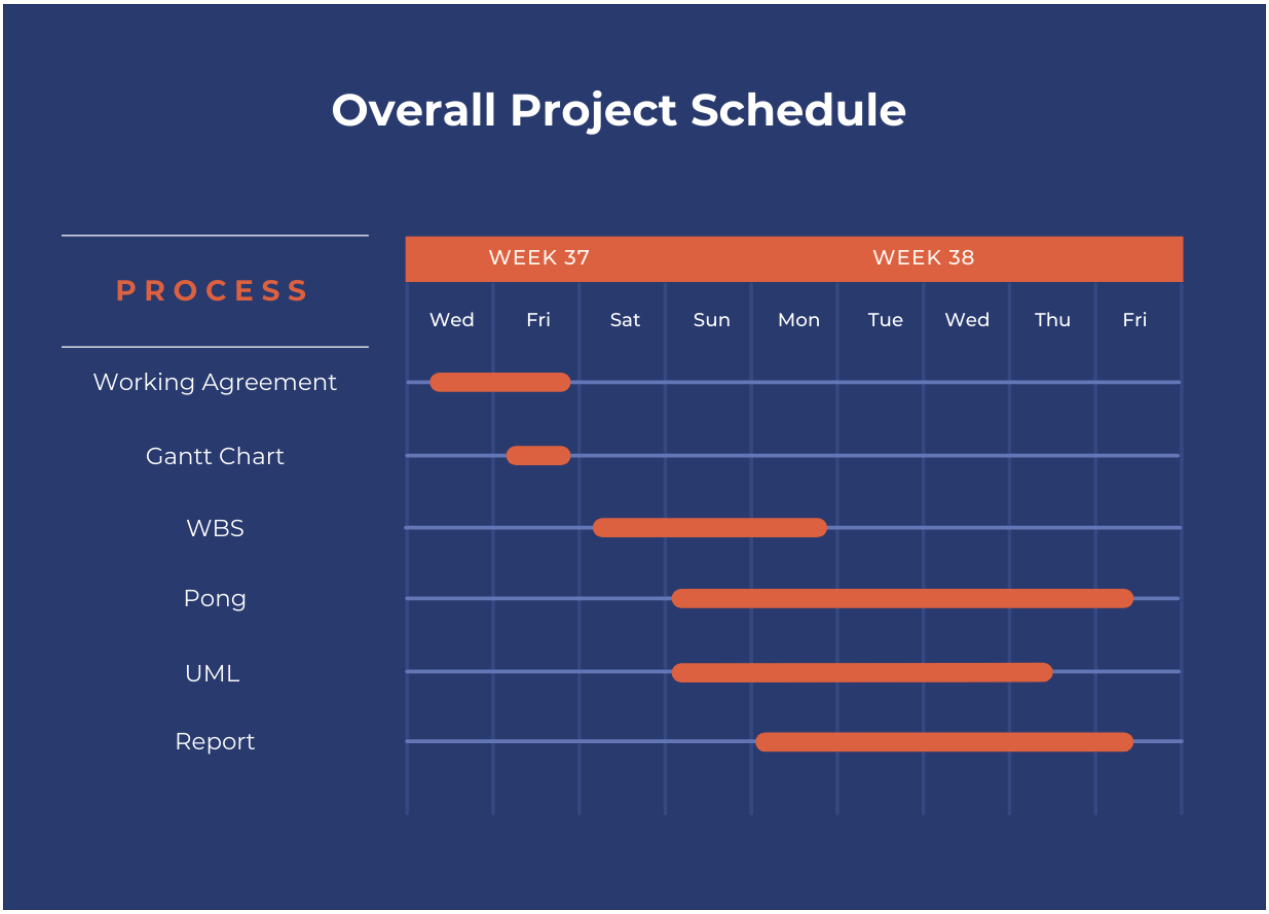

Figure 1 – Gantt Diagram for Pong

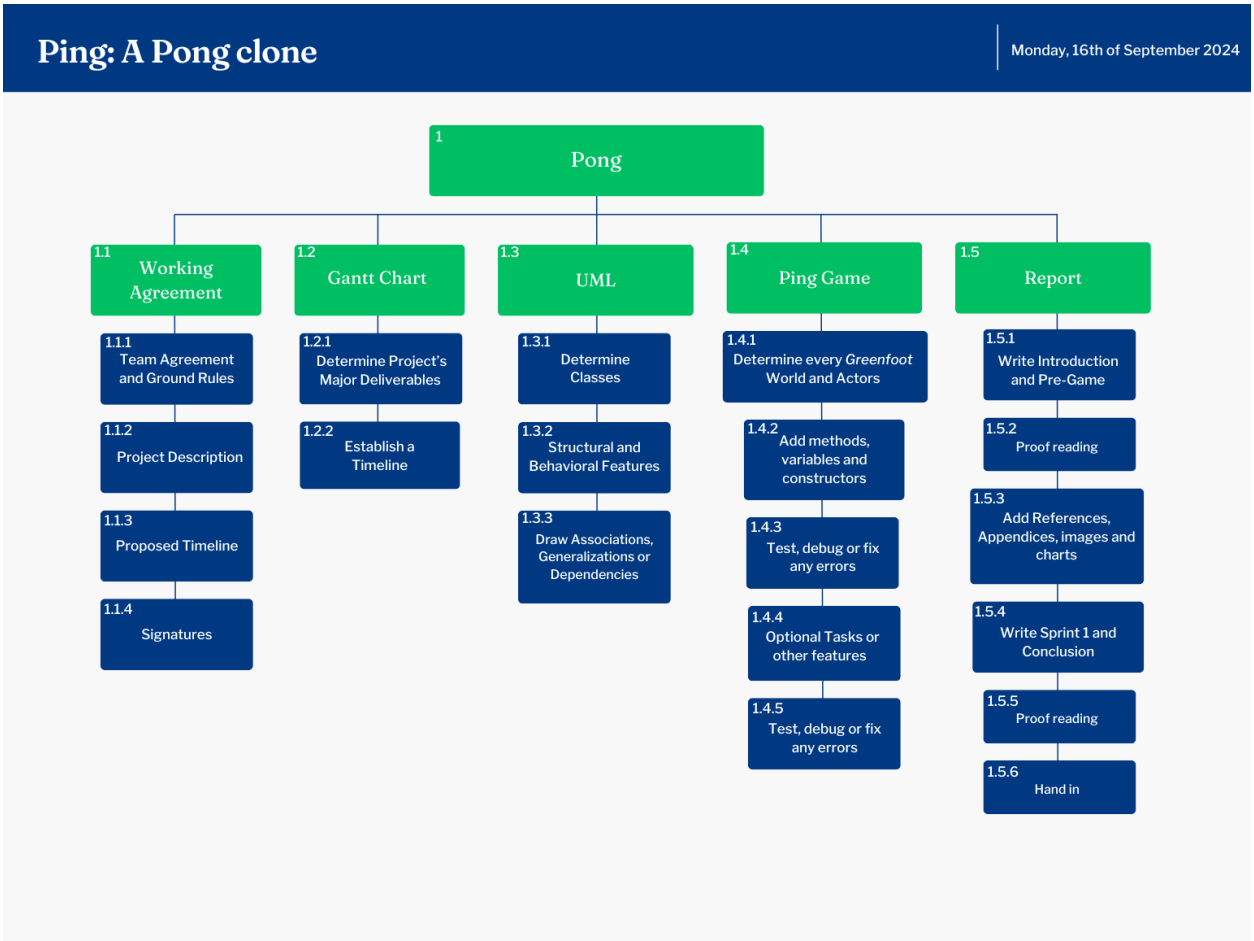

Figure 2 – Work Breakdown Structure (WBS) for Pong

2.3. Initial Product Backlog

The Mandatory Tasks

1. Construct a single-player game, with the paddle at the floor and the ball will bounce off the ceiling and walls. The “left” and “right” keys should operate the paddle.
2. Add a self-moving paddle, which enters the world at a random position. The self-moving paddle moves horizontally. When the paddle moves out of a wall, a new paddle should show up at a random height on the other wall. Make sure the self-moving paddle is not created too near the gamers paddle.

3. When the ball hits a self-moving paddle from below it will bounce off the paddle. If the ball hits the self-moving paddle from above the ball will go right through the paddle.
4. Change the images of the game so it looks nice, such as new images for the ball and the paddles.
5. Put sounds to the game for ball hitting the paddle, ceiling, walls, and the floor (game over).
6. Each time the ball has been hit by the paddle 10 times, the speed of the ball is increased slightly.
7. Include a text field with the text "Game Level: 1" in the upper right corner of the game. Each time the speed is increased, the game level is increased too. Make sure the ball can pass in front of the text.

Optional Tasks (not prioritized)

a) Use an opening screen with an image and maybe some instructions. Get the screen to be shown for a certain amount of time, or until the player hits the "enter"-key.
b) Display a game-over image and play a sound when the game is over. Get the screen to be shown a certain amount of time, or until the player hits the "enter"-key, and then restart the game.
c) Add different sizes of self-moving paddles.
d) Include the second computer-controlled paddle at the ceiling. Each time the player-controlled paddle is returning the ball, the computer-controlled paddle should start moving towards the point of contact. When the computer-controlled paddle hits the ball, the paddle stays in position awaiting the returning ball from the player-controlled paddle.
e) Include two score fields instead counting the number of games won by the player and the computer (see the image of the original game).
f) Make the computer-controlled paddle work better. Let it seek back towards the middle position after it has returned a ball. If the player returns the ball before the middle point has been reached, the paddle is off course moving towards the point of ball contact.
g) And whatever crosses your mind to make the game more fun…

2.4. Architecture

We started building Pong around the Ping game provided by the teachers. This version contained class IntroWorld and class PingWorld as part of Greenfoot's and Ping's World and class Ball and class Paddle as Actors in the game.

To fulfill the provided Mandatory Tasks, we added a new Actor called MovingPaddle and imported the SmoothMover class as an extension of class Ball to improve the game's ball movement.

The classes, their Classifiers, Structural and Behavioral Features and all the Associations, Generalizations or Dependencies between them are presented in the following Unified Modeling Language (UML).

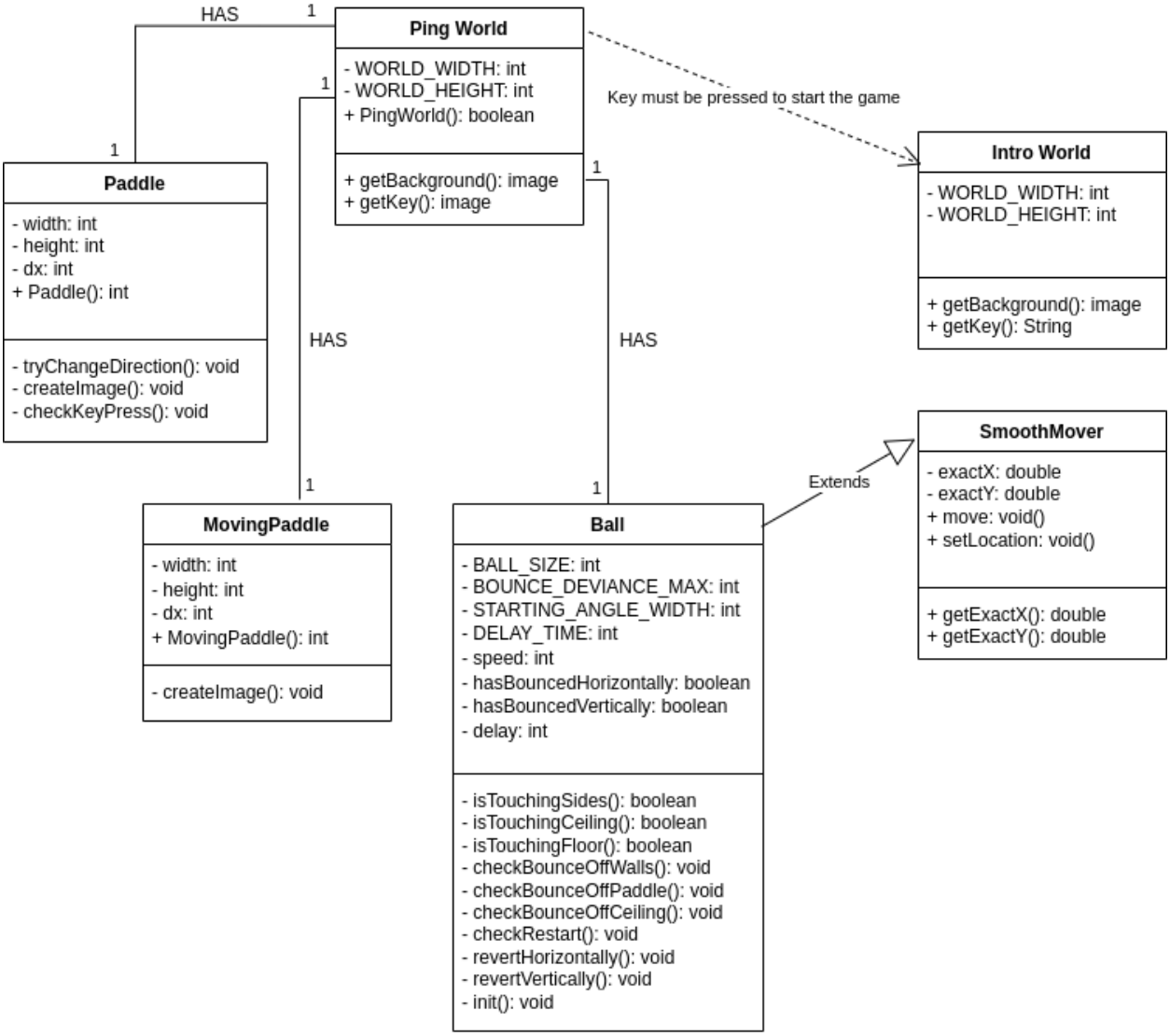

Figure 3 – Pong UML

3. Sprint 1

3.1. Graphical User Interface (GUI)

The following images depict the Graphical User Interface (GUI) for each of the two possible worlds in the game's *Greenfoot* Scenario.

Figure 4 – The Space title screen to the left and the Safari title screen to the right

One of the ideas we had for the game upon completing our list of The Mandatory Tasks was to allow the Player to choose multiple themes. We therefore chose a Space theme with a background image filled with stars and a spaceship for the computer's paddle. Our second theme is called Safari and transports the Player back to our beloved planet Earth, to this beautiful harmonious place surrounded by critters as the sun sets, with no Wi-Fi or phone signal but a cloud as the computer's paddle to distract him.

Apart from the change in scenery and a change of colors for both the ball and the Player's paddle, there are no changes in the game's rules or mechanics.

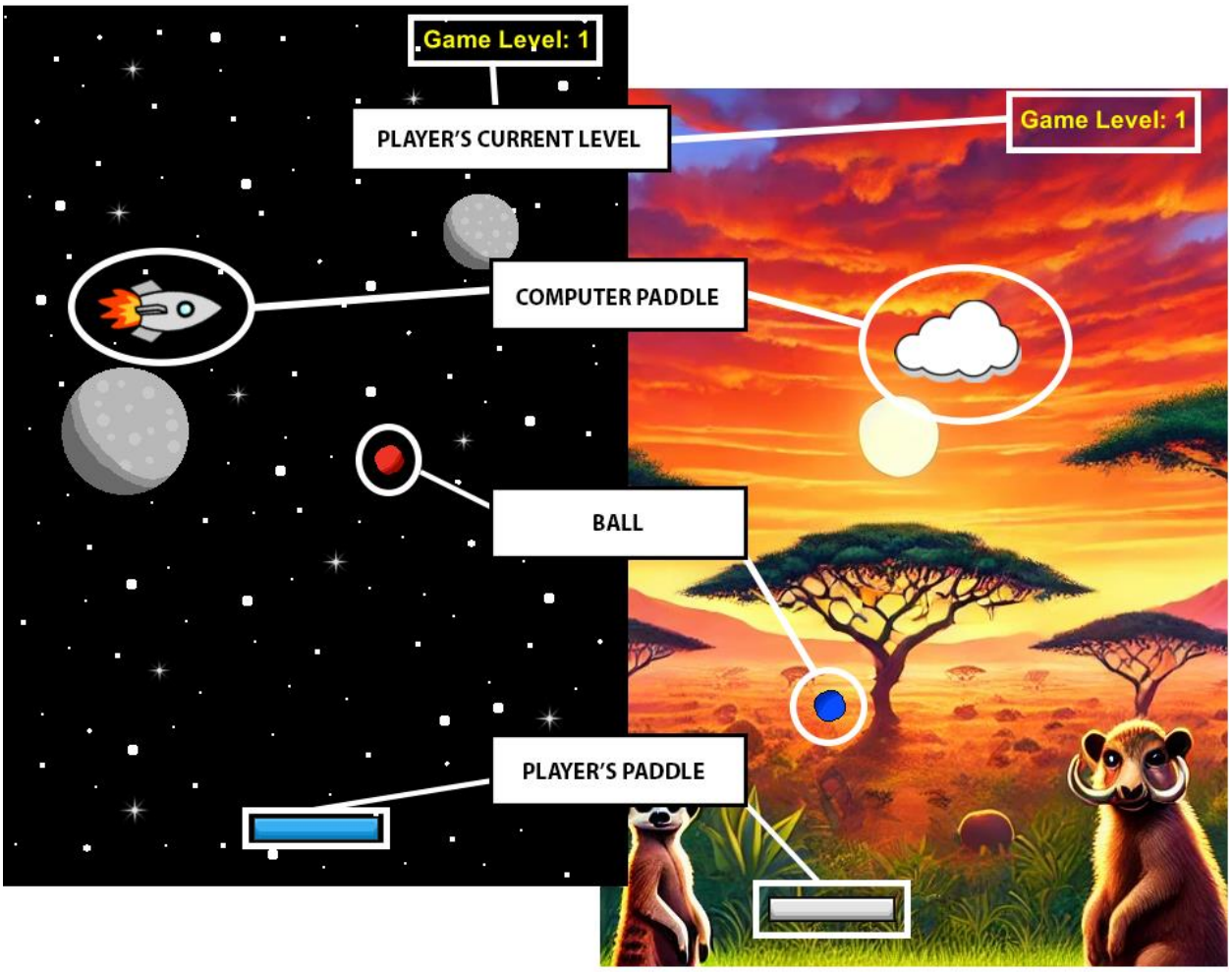

Figure 4 – Graphical User Interface (GUI) for Pong

The figure above shows what the Player sees in both environments. Both include a ground paddle, the Player's Paddle, that is controlled by the latter by pressing the "left" or "right" keys on the keyboard (left and right arrows, respectively), a Ball, a Computer Paddle and a Game Level counter.

3.2. Implementation

3.2.1. Code examples

One of The Mandatory Tasks that we took some time to complete was task number three. Here if the ball hits the self-moving paddle (i.e. the Computer Paddle) from below it will bounce out of it but if the ball hits it from above the ball should go right through the paddle.

The solution isn't perfect, but we were able to figure out what method we should alter and how to achieve the desired effect.

```
private void collisionWithSelfMovingPaddle()
{
SelfMovingPaddle selfPaddle = (SelfMovingPaddle) getOneIntersectingObject(SelfMovingPaddle.class);

if (selfPaddle != null) {
    // If the ball is below the paddle, bounce
    if (getY() > selfPaddle.getY() + selfPaddle.getImage().getHeight() / 2) {
        if (!hasBouncedVertically) {
            revertVertically();
            hasBouncedVertically = true;
        }
    }
    // If the ball hits the paddle from the top, allow it to pass through
    else if (getY() <= selfPaddle.getY() - selfPaddle.getImage().getHeight() / 2) {
        // Move the ball down past the paddle to ensure it completely passes through
        hasBouncedVertically = false;

        //setLocation(getX(), selfPaddle.getY() + selfPaddle.getImage().getHeight());
    }
} else {
    hasBouncedVertically = false; // Reset bounce flag for future collisions
}
}
```

Figure 5 – Our first take on Mandatory Task #3

```
private void collisionWithSelfMovingPaddle()
{
SpaceSelfMovingPaddle selfPaddle = (SpaceSelfMovingPaddle) getOneIntersectingObject(SpaceSelfMovingPaddle.class);

if (selfPaddle != null) {
    // Ball moving upwards (negative vertical direction)
    if (getY() < selfPaddle.getY() && getRotation() > 180 && getRotation() < 360) {
        if (!hasBouncedVertically) {
            revertVertically(); // Bounce the ball downward
            Greenfoot.playSound("PADDLE.wav");
            hasBouncedVertically = true;
        }
    }
} else {
    hasBouncedVertically = false; // Reset bounce flag when not colliding
}
}
```

Figure 5.1 – Our second and final take on Mandatory Task #3

Mandatory Task number two proposed also an interesting challenge. The following screenshot depicts a successful attempt in adjusting the self-moving paddle's touching the edge of the *Greenfoot* World behavior, respawning on the opposite side in a random position within the playable area.

```
// Respawn the paddle at a random height on the opposite wall
private void respawnAtRandomHeight()
{
    World world = getWorld();
    SpacePaddle gamer = (SpacePaddle) world.getObjects(SpacePaddle.class).get(0);  // Assuming there's only one gamer paddle

    int newY;
    do {
        // Random Y position for the new paddle
        newY = Greenfoot.getRandomNumber(world.getHeight());
    } while (Math.abs(newY - gamer.getY()) < SAFE_DISTANCE_FROM_GAMER);  // Ensure it doesn't spawn too close to the gamer paddle

    int newX;
    if (getX() <= 0)  // If it hit the left wall, respawn on the right wall
    {
        newX = world.getWidth() - 1;
        direction = -1;  // Move left
    }
    else  // If it hit the right wall, respawn on the left wall
    {
        newX = 0;
        direction = 1;  // Move right
    }

    // Set the new position of the paddle
    setLocation(newX, newY);
}
```

Figure 6 – Our take on Mandatory Task #2

3.3. Sprint Review

As determined, we were able to complete all seven Mandatory Tasks. Our final version of the game Pong, built from *Greenfoot* Scenario Ping provided by the teachers, offers a controlled paddle, operated by the "left" and "right" keys of the keyboard, a self-moving paddle that enters the world at a random position when exiting the screen, on the opposite side, and allows the ball to bounce from the bottom but let it go through from the top. There are different images for both paddles and the ball, sounds when the ball hits different surfaces or the bottom of the screen – game over – and increased speed after ten successful saves by the Player and when the Game Level (displayed on the upper right corner) increases.

The final version includes Optional Tasks a) and b).

3.4. Sprint Retrospective

We believe the project was challenging enough to test our knowledge and understanding of the subject matter. There were some interesting challenges within The Mandatory Tasks that led us to a stall, especially when we couldn't fully understand where and how to alter the code from Ping provided by the teachers. It led to a lot of trials and errors, but we were able to overcome those obstacles collectively, bouncing ideas and possible solutions on and off from each other, maintaining a positive attitude and not giving up and acknowledging it is all part of the process and we can simply gain and grow out of them.

A realization in work productivity and flow arose however when all party members of Group 1 agreed and understood halfway through the process, that versioning by dropping files and folders on Discord or on Google Drive is not ideal.

There is still much to be done and to improve – the ball movements and behavior when hitting different sections of either the self-moving or the controlled paddle (corners or middle), for example – but we are pleased with what we've accomplished and had fun while doing so.

4. Conclusion

In this first compulsory assignment, we were able accomplish what we set out to do.

It was our first time working in groups and with only three weeks since we all first met and started this journey there was naturally a lot to assess and get a feel for. Everything came together however, we were able to follow our Working Agreement, stick to our Overall Project Schedule and Pre-Game "checklist" and deliver a final product that meets the demands and that we are very proud of.

Acknowledgment needs to be given to the importance of planning and taking time in discussing and setting models and systems that undoubtedly helped us navigate through the path towards our goals.

5. References

Algonquin College Library (Jan 28, 2020). *Creating a group contract (archived)*. Accessed in: Tuesday 10th of September 2024: https://www.youtube.com/watch?v=5YB8DfwBtNw.

Online PM Courses - Mike Clayton (Oct 14, 2021). *How to Create a Work Breakdown Structure: A WBS Masterclass.* Accessed in: Tuesday 10th of September 2024: https://www.youtube.com/watch?v=PyR2VLP3xnA&t=2s.

Online PM Courses - Mike Clayton (Feb 28, 2019). *Project Management in Under 5: What is a Gantt Chart?*. Accessed in: Tuesday 10th of September 2024: https://www.youtube.com/watch?v=fB0wsdmV3Sw.

Christian Pfeiffer Jensen (Jul 1, 2019). *5R model*. Accessed in: Tuesday 10th of September 2024: https://www.youtube.com/watch?v=r6X_MwOKaYk.

Lucid Software (Aug 10, 2023). *UML class diagrams*. Accessed in: Wednesday 11th of September 2024: https://www.youtube.com/watch?v=6XrL5jXmTwM.

Greenfoot. greenfoot (greenfoot API). Accessed in: Monday 16th of September 2024: https://www.greenfoot.org/files/javadoc/greenfoot/package-summary.html.

6. Appendices

6.1. Working Agreement

GROUP CONTRACT

Project Name: Pong

Project Due Date: Friday, 20th September 2024, 2:00 PM CEST (UTC +2)

Team Members:

Name	Phone number	Email
Ahmed Mohamad Issa Asfour	+45 53 34 49 51	ahmedasfoor13@gmail.com
Davide José da Silva Jorge	+45 50 33 19 51	davidejsjorge@gmail.com
Katja Tamstrup Strunck	+45 30 29 71 91	katjats2004@hotmail.com
Luis Carlos Pacheco Cartucho	+45 26 25 74 87	cartucho18.01.2000@gmail.com

Team Agreement and Ground Rules:

Issue	Agreements
Personal Interactions	1. We will meet daily within our regular class schedule. 2. We will meet at the campus and determine where to sit and work. 3. We will check in daily with each other during our meetings and on Discord. 4. We will make use of Google's Drive and Docs as repository and shared online document editor, respectively.
Roles and Responsibilities	• The Leader: leads discussion and checks for consensus or questions from group members.

	• The Organizer: schedules the group meetings, creates agendas and takes notes at meetings to send to everyone. • The Researcher(s): looks for project topics and finds sources and information used to write and fulfil the assignment. • The Troubleshooter: thinks about the positives and negatives of the ideas presented by the group and produces viable solutions to problems. • The Writer(s): writes the project and integral parts. • The Editor: compiles different pieces of the report from the group members to make them flow together as one consistent work; edits and proofreads the completed work before submission.
Distribution of the workload	✓ Katja: Leader, researcher, writer. ✓ Ahmed: Organizer, researcher, writer. ✓ Davide: Troubleshooter, researcher, writer, editor. ✓ Luís: Troubleshooter, researcher, writer.
Managing conflict	❖ Try to work out conflict between group members before bringing any problems to the teacher. ❖ Be open to dialogue. ❖ Bring always constructive criticism.

	❖ Adopt a 3 Strike System where at a third instance of unjustified absence, unruly behavior or anything that sabotages the assignment, members are kicked out of the group.
Others	♦ Whoever is late unannounced or misses a meeting will have to bring cake. ♦ Cake is always welcome. ♦ In extreme cases of prolonged absence, a clear lack of commitment, unruly behavior and/or failure to comply, and by unanimous decision, group members will be expelled.

Description:

For our first compulsory assignment we are to create a version of the video game *Pong* in Java.

Apart from making the proposed game we will elaborate on our collective thoughts and ideas making use of a Gantt chart and a Work Breakdown Structure (WBS).

What products will the group generate to complete this project?

Greenfoot application, UML model and report.

What research is needed to do this project?

Research in Java programming language and the Greenfoot IDE and its features. Research in Unified Modeling Language (UML). Research the original video game *Pong* and its features and possibilities for expanding the game.

Proposed Timeline:

	Date of completion	**Who is Primarily responsible?**
Part 1 Working Agreement	Friday, 13th of September 2024	All
Part 2 Gantt Chart	Friday, 13th of September 2024	All
Part 3 Work Breakdown Structure	Monday, 16th of September 2024	Katja, Luís
Part 4 UML	Thursday, 19th of September	Ahmed, Luís
Part 5 Report	Friday, 20th of September	All
Final Product Pong	Friday, 20th of September	All

Signatures:

Ahmed Mohamad Issa Asfour

Katja Tamstrup Strunck

Luis Carlos Pacheco Cartucho

Davide José da Silva Jorge