

Student name:	Nahal Kianpour					
Student number:	3124767					
Faculty:	Computing Science					
Course:	BSCH/BSCO/EXCH			Stage/year:	2	
Subject:	Software Development 2					
Study Mode:	Full time	<input checked="" type="checkbox"/>		Part-time	<input type="checkbox"/>	
Lecturer Name:	Gemma Deery					
Assignment Title:	Report 3					
Date due:	30/04/2025					
Date submitted:	30/04/2025					

Plagiarism disclaimer:

I understand that plagiarism is a serious offence and have read and understood the college policy on plagiarism. I also understand that I may receive a mark of zero if I have not identified and properly attributed sources which have been used, referred to, or have in any way influenced the preparation of this assignment, or if I have knowingly allowed others to plagiarise my work in this way.

I hereby certify that this assignment is my own work, based on my personal study and/or research, and that I have acknowledged all material and sources used in its preparation. I also certify that the assignment has not previously been submitted for assessment and that I have not copied in part or whole or otherwise plagiarised the work of anyone else, including other students.

Signed: _Nahal Kianpour Lirharani_

Date: _30/04/2025_

Please note: Students MUST retain a hard / soft copy of ALL assignments as well as a receipt issued and signed by a member of Faculty as proof of submission.

1. Features I Worked On.....	3
1.1 Features I Contributed To.....	3
2. What Was Completed.....	3
2.1 Completed Functionality.....	3
3. Who Did What.....	3
3.1 Team Role Breakdown.....	3
4. Challenges Faced / What I Learned.....	4
4.1 Challenges and Key Learnings.....	4
5. Git Log.....	5
5.1 Git Commit History.....	5



1. Features I Worked On

1.1 Features I Contributed To

In this project, I created a full countdown timer system using a custom `Timer` class and `HeadsUpDisplay`, which shows a live mm:ss countdown on screen. When time expires, the game shows a centered “Time’s Up!” message using a `showGameOver` flag to ensure it’s triggered only once. I wrote a full suite of unit tests to validate all time-related behaviors.

I also fixed broken menu functionality by removing the old `MainClass.main(null)` call and launching the `Game` instance directly when the Start button is clicked. To improve gameplay balance, I increased the jump height by adjusting `jumpSpeed`, allowing players to reach upper platforms.

For win conditions, I added the `allCoinsCollected()` method in `CoinManager` to detect when all collectible items are gone. I connected it with `Game.java` to trigger a centered green “You Win!” message and stop the timer. I also debugged and fixed an issue where coins spawned off-screen and remained uncollectible by adding bounds-checking in the spawn logic.

2. What Was Completed

2.1 Completed Functionality

- **Timer and HUD System:** Designed and implemented a custom `Timer` class with mm:ss formatting, frame-accurate countdown using delta time, and a single-trigger `onTimeUp()` method to handle game-over logic. Created a `HeadsUpDisplay` class to draw the live timer in the top-left of the screen. Ensured clean state handling using `showGameOver` and `timeUpHandled` flags.
- **"You Win" Condition with CoinManager:** Developed the `allCoinsCollected()` method to detect win condition when all coins are picked up. Integrated this logic into the `Game` class to stop the timer and display a centered green "You Win!" message using a `showWinMessage` flag.
- **Coin Spawn Bug Fix:** Fixed an issue where some coins were spawning outside the visible screen area by adding screen-bound checks in `addTestCoins()` using `Game.GAME_WIDTH` and `Game.GAME_HEIGHT`.

3. Who Did What

3.1 Team Role Breakdown

- **Nahal Kianpour (me):**
Built the level system and integrated environment rendering using LevelManager and LoadSave. Implemented the UPS (Updates Per Second) performance tracker and

set up a modular testing framework using a `testcases` package. Refactored core Game logic to prevent null exceptions and maintain layout consistency with global constants.

In the later stages, I developed a complete countdown timer system with HUD integration, frame-accurate delta time handling, and a centered “Time’s Up!” message. I also implemented the coin win condition using `CoinManager`, added the “You Win!” screen logic, and fixed a bug where coins spawned outside the screen.

- **Polina Shtefan:**

Implemented the animation system for characters, organized sprite constants, structured the `utilz` package, and handled visuals. Later, she also built the core `CoinManager` and `Coin` classes to support coin collection logic and in-game currency behavior.

- **Thanh Phuong Hoang:**

Created the menu system and connected it to the main game flow. She fixed the menu-to-game transition using button listeners and ensured the Start button correctly launches the gameplay session.

- **Veronika Hrytsyk:**

Implemented the initial collision detection system and created hitboxes to enable player interaction with level objects. She later finished the gravity system and finalized character–platform collision logic for smoother movement and physics.

4. Challenges Faced / What I Learned

4.1 Challenges and Key Learnings

- **Countdown timer and game state handling:**

The timer system originally froze the game after reaching 0. I implemented a `showGameOver` flag and a one-time `onTimeUp()` trigger, which taught me how to manage time-based events cleanly without affecting core updates.

- **Win condition detection and visual feedback:**

Implementing the win state based on coin collection required careful coordination across game states. I introduced a `showWinMessage` flag and refined rendering logic to prevent overlaps with game-over messages.

- **Coin positioning bug:**

Some coins were spawning off-screen, so I updated the placement logic to check bounds using `Game.GAME_WIDTH` and `GAME_HEIGHT`. This taught me how to validate object spawning using global constraints.

- **Collaboration and modularity:**

Working through Git and syncing regularly helped me see the value of modular class design and commit clarity. It was easier to integrate features like the timer and coins into the team codebase without breaking others’ work.

5. Git Log

5.1 Git Commit History

add tests for minute/second extraction and formatted time string	5bedd6f	<>
● Nahal-Kianpour-gcd committed 1 hour ago		
add test to ensure timer clamps to zero and reports finished	1842b9b	<>
● Nahal-Kianpour-gcd committed 1 hour ago		
add test to verify timer countdown after update	c2a6683	<>
● Nahal-Kianpour-gcd committed 1 hour ago		
add basic test to verify initial time formatting	0c8c64a	<>
● Nahal-Kianpour-gcd committed 1 hour ago		
add skeleton JUnit tests for Timer	78c552a	<>
● Nahal-Kianpour-gcd committed 1 hour ago		
increase jumpSpeed to -3.5 to reach higher platforms, preserved old value in comments	bd25771	<>
● Nahal-Kianpour-gcd committed 1 hour ago		
Commits on Apr 28, 2025		
updated the MainClass to launch Menu and fix Start button to correctly launch Game	88a80c2	<>
● Nahal-Kianpour-gcd committed yesterday		
Merge branch 'TimerFunction'	3f53d6a	<>
● Nahal-Kianpour-gcd committed yesterday		
extend render() to display 'Time's Up!' when timer finishes	28ca98d	<>
● Nahal-Kianpour-gcd committed yesterday		
add showGameOver flag and update onTimeUp() to set it	1767033	<>
● Nahal-Kianpour-gcd committed yesterday		
add timeUpHandled flag to ensure onTimeUp() triggers only once	2f4364a	<>
● Nahal-Kianpour-gcd committed yesterday		

add onTimeUp() method to handle countdown completion	133044c	<>
● Nahal-Kianpour-gcd committed yesterday		
extend render method to draw HeadsUpDisplay (timer)	fc4113f	<>
● Nahal-Kianpour-gcd committed yesterday		
updated game loop to include the timer countdown and the time-up check	143a909	<>
● Nahal-Kianpour-gcd committed yesterday		
initialize Timer, HeadsUpDisplay, and lastNanoTime in constructor	d6f2169	<>
● Nahal-Kianpour-gcd committed yesterday		
add Timer, HUD, and lastNanoTime fields for countdown and display	09c7d79	<>
● Nahal-Kianpour-gcd committed yesterday		
actually include HeadsUpDisplay changes	60881a2	<>
● Nahal-Kianpour-gcd committed yesterday		
add HeadsUpDisplay class to render timer in the UI	8e9037a	<>
● Nahal-Kianpour-gcd committed yesterday		
add getSeconds() and getTimeString() for mm:ss formatting	c1b473c	<>
● Nahal-Kianpour-gcd committed yesterday		
add getMinutes() to return total whole minutes left	2781e7d	<>
● Nahal-Kianpour-gcd committed yesterday		
Apr 28, 2025, 3:44 PM GMT+1		
add isFinished() to indicate timer has reached zero	498f0a8	<>
● Nahal-Kianpour-gcd committed yesterday		
implement update(deltaSeconds) to subtract elapsed time and floor at zero	7846b16	<>
● Nahal-Kianpour-gcd committed yesterday		
add Timer class with timeRemaining field and constructor for countdown duration	582f204	<>
● Nahal-Kianpour-gcd committed yesterday		

Commits on Apr 30, 2025		
Added win condition for collecting all coins + game timer stops when win	4a67c7a	<>
● Nahal-Kianpour-gcd committed 2 minutes ago		
Enhance CoinManager with bounds checking, debug borders, and win detection	60f4185	<>
● Nahal-Kianpour-gcd committed 12 minutes ago		



GRIFFITH COLLEGE DUBLIN