Assignment 2 - Report

6 Possible Points

10/10/2021

Attempt 1

IN PROGRESS
Next Up: Submit Assignment

Add Comment

Unlimited Attempts Allowed
13/09/2021

Details

Problem Overview

You are required to design and implement an application model for a classic shoot 'em up video game *Ballboy* in three stages. The video game consists of a character on the screen that bounces up and down, can move left and right, and has the ability to modify how high they bounce. There will also be a json configuration file that will be read to specify initial behaviour.

In later stages, the character will interact with the world. There may be a floor to bounce on, obstacles to interact with (bounce, hit, etc), enemies to avoid, and a way to maintain a score. The levels of the game may have different behaviours such as allowing Ballboy to temporarily be able to hover rather than bounce, or have a companion ball that can damage enemies.

You will need to ensure that your application is configurable with a JSON text file. You are given an example JSON file format here (https://canvas.sydney.edu.au/courses/36379/files/19057268/download) that you can follow. The sample may need some modification to fit into your concept of multiple levels.

Q: What is a classic "shoot 'em up" video game?

A: The most relevant example:
https://www.youtube.com/watch?v=n4Vpeoaw6g4 (https://youtu.be/xxhP6vD3unY)

(https://youtu.be/xxhP6vD3unY)

Assignment 2 Requirement

In assignment 2, you are going to implement your Ballboy game and refactor your UML model according to your code. You must use design patterns in your implementation. All Entities will be constructed using a Factory method that takes a snippet of json as input. Behaviour of entities will be controlled using the Strategy design pattern. Please find the detailed tasks below:

Submit Assignment

You will use the Java Programming Language to implement the UML model you designed in assignment 1. In this assignment, you are now responsible for the implementation of the entire application code including everything (e.g., model package, view package, resources, etc).

What we provide to you

- Codebase: the same codebase you used in your assignment 1 (https://canvas.sydney.edu.au/courses/36379/files/19057240/download) is the base for this assignment. You are allowed to decide by yourself to use as much or as little of this existing code in your implementation. Please note that, this provided codebase was not given to you as an example of good design. You are also allowed to source images or draw images by yourself for your game (ensure you acknowledge any image source in a readme file). It is likely that the provided image resources will not be sufficient.
- JSON file: an example JSON file format is provided to you here (https://canvas.sydney.edu.au/courses/36379/files/19057268/download) that you can start with for implementing your configuration file.
- gradle file: a sample build.gradle file is provided to you here (https://canvas.sydney.edu.au/courses/36379/files/19057255/download) .
- 2D AABB (Axis-Aligned Bounding Box) Function: it may be helpful for you in your implementation.
  - A simple collision/box intersection detection function allows your program to detect when two rectangular shapes have intersected. This is a general algorithm that checks for an overlap between two rectangles, returning *true* if an overlap has occurred.

```
aabbintersect (box1, box2):
  return (box1.x < (box2.x + box2.width)) and
  ((box1.x + box1.width) > box2.x) and
  (box1.y < (box2.y + box2.height)) and
  ((box1.y + box1.height) > box2.y)
```

What we expect from you

Your Ballboy game is now expected to support the following features in your code:

- Ballboy can be created in three sizes which are defined within your configuration file. The size of the Ballboy is specified as a string whose options are small, medium. and large. Ballboy's starting position is as an x-y coordinate. Your Ballboy will be continuously bouncing. Ballboy has some control over the height of the bounce, and can accelerate left and right (until top speed is reached). The camera follows Ballboy's movement, both horizontal and vertical. (A demo of camera movement, using last year's assignment (https://canvas.sydney.edu.au/courses/36379/files/18462221/download?wrap=1) (https://canvas.sydney.edu.au/courses/36379/files/18462221/download?download_frd=1)

Submit Assignment

)

- Level definitions
  - Level definitions must be loaded from your JSON configuration file which gives instructions for the various assets to be loaded and positioned.
  - Level information must include immobile objects that Ballboy will bounce off, 'enemies' that harm Ballboy if touched, cloud information, a finishing object, and Ballboy's starting location. When Ballboy encounters the finishing object the Level is deemed to be complete. In this assignment finishing the level will exit the game. The location of all objects must be configurable .
  - Levels have a floor to bounce on, with a location and appearance defined in the JSON.
- Levels will contain enemy agents which will harm Ballboy
  - Enemies have different movement personalities (e.g., some enemies move from left to right, some run or fly away from/toward Ballboy, etc)
  - If Ballboy touches enemies, Ballboy will be moved back to the originally configured start location.
- Clouds exist and move at a constant speed and are behind other objects (i.e. background scenery)
  - Ballboy cannot interact with clouds.

Note that, there are some features that you don't need to implement in your current assignment, and should not :

- Level transitions during execution of the game
  - Currently the game loads a single given level file on startup and exits when that level is complete.
- 'Lives'
  - Currently Ballboy just resets back to the beginning of the level.
- Score
- Timer
- Load/Save

Report Task

You are allowed a maximum of 1000 words report in this assignment which must concisely cover the followings:

1. A discussion on how your design for assignment 1 helped or hindered your extensions made in this assignment
   - Rationalise changes you have made to your assignment 1 design
2. A discussion on each design pattern you have used including
   - Where you used it (be explicit as to what classes are involved and in what roles)

Submit Assignment

general)
    - What drawbacks this pattern causes (be specific to your code, not the pattern in general)
3. An updated UML class diagram describing your whole system, including design patterns.
4. Any acknowledgement/reference required.

Submission Details

You are required to submit all assessment items by the due date listed on Canvas.

- Report. Submit your UML class diagram and your report as a SINGLE pdf document on this portal. If your UML diagram is too large, then you may want to include enlarged versions of the key components when you refer to them.
- Code. Your code should be submitted as a zip file containing only your src folder, build.gradle, example json configuration files, and readme.txt .
    - The readme,txt file will cover any point you would like your marker to know
        - how to run your code (e.g., any quirks to run your application)
        - a description of your JSON format
        - example json configurations and what they demonstrate
        - which files and classes are involved in each design pattern implemented

Academic honesty

While the University is aware that the vast majority of students and staff act ethically and honestly, it is opposed to and will not tolerate academic dishonesty or plagiarism and will treat all allegations of dishonesty seriously.

Further information on academic honesty, academic dishonesty, and the resources available to all students can be found on the academic integrity pages on the current students website: **https://sydney.edu.au/students/academic-integrity.html (https://sydney.edu.au/students/academic-integrity.html)** .

Further information for on research integrity and ethics for postgraduate research students and students undertaking research-focussed coursework such as Honours and capstone research projects can be also be found on the current students website: **https://sydney.edu.au/students/research-integrity-ethics.html (https://sydney.edu.au/students/research-integrity-ethics.html)** .

Compliance statement

In submitting this work, I acknowledge I have understood the following:

Submit Assignment

RendNum=0) .

- The work is substantially my own and where any parts of this work are not my own I have indicated this by acknowledging the source of those parts of the work and enclosed any quoted text in quotation marks.
- The work has not previously been submitted in part or in full for assessment in another unit unless I have been given permission by my unit of study coordinator to do so.
- The work will be submitted to similarity detection software (Turnitin) and a copy of the work will be retained in Turnitin's paper repository for future similarity checking. Note: work submitted by postgraduate research students for research purposes is not added to Turnitin's paper repository.
- Engaging in plagiarism or academic dishonesty in coursework will, if detected, lead to the University commencing proceedings under the **Academic Honesty in Coursework Policy 2015** (https://sydney.edu.au/policies/showdoc.aspx?recnum=PDOC2012/254&RendNum=0) and the **Academic Honesty Procedures 2016** (http://sydney.edu.au/policies/default.aspx?mode=glossary&word=Academic+honesty) .
- Engaging in plagiarism or academic dishonesty in research-focussed work will lead to the University commencing proceedings under the **Research Code of Conduct 2013** (https://sydney.edu.au/policies/showdoc.aspx?recnum=PDOC2013/321&RendNum=0) and the **Academic Honesty Procedures 2016** (http://sydney.edu.au/policies/default.aspx?mode=glossary&word=Academic+honesty) .
- Engaging another person to complete part or all of the submitted work will, if detected, lead to the University commencing proceedings against me for potential student misconduct under the **University of Sydney (Student Discipline) Rule 2016** (http://sydney.edu.au/policies/showdoc.aspx?recnum=PDOC2017/441&RendNum=0) .

View Rubric

Webcam | Files | Dropbox | Google LTI | Office365 LTI | Studio

Drag a file here, or

Choose a file to upload

File permitted: PDF, ZIP

Submit Assignment

☐ I agree to the tool's End-User License Agreement (https://api.turnitin.com/api/lti/1p0/user/static_eula)
This assignment submission is my own, original work

Submit Assignment