

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO
HỌC PHẦN CÔNG NGHỆ JAVA
ĐỀ TÀI:
XÂY DỰNG GAME BRICK BREAKER

Sinh viên thực hiện : Nguyễn Văn Quý
Lớp : CNTT6-K62
Mã sinh viên : 211241738
Giáo viên hướng dẫn : Vũ Huấn

Hà Nội - 2023

MỤC LỤC

MỤC LỤC	1
LỜI CẢM ƠN	2
NỘI DUNG.....	3
1. Giới thiệu đề tài.....	3
2. Công nghệ sử dụng.....	3
3. Thiết kế game	3
4. Phát triển game.....	9
5. Kiểm thử và đánh giá game	15
KẾT LUẬN.....	17
TÀI LIỆU THAM KHẢO	18

LỜI CẢM ƠN

Em xin gửi lời cảm ơn sâu sắc đến thầy Vũ Huấn vì đã tận tình giảng dạy, chỉ bảo và hỗ trợ chúng em hết mình trong học phần Công nghệ Java. Em cũng cảm ơn thầy vì em đã có những kiến thức, kỹ năng quan trọng trong quá trình học học phần này. Qua học phần này, em hiểu rõ hơn về các kỹ thuật lập trình và cung cấp cho em những kiến thức quan trọng nhất để em có thể phát triển các ứng dụng và chương trình trên nền tảng Java.

Em cảm ơn thầy vì đã truyền đạt cho em những kinh nghiệm quý giá trong việc xử lý các vấn đề trong quá trình lập trình, giúp em đạt được mục tiêu của học phần và hoàn thành các bài tập một cách tốt nhất. Những kỹ năng và kinh nghiệm mà em đã học được trong quá trình này sẽ rất hữu ích cho công việc của em trong tương lai.

Tổng kết lại, em rất biết ơn và cảm kích sự giúp đỡ và hỗ trợ nhiệt tình của thầy trong quá trình học tập và phát triển kỹ năng của chúng em. Em hy vọng sẽ có cơ hội được tiếp tục học tập và làm việc cùng thầy trong các học phần sắp tới.

NỘI DUNG

1. Giới thiệu đề tài

Trong đề tài báo cáo bài tập lớn Java này, em sẽ tập trung vào việc phát triển game Brick Breaker. Brick Breaker là một trò chơi arcade kinh điển, trong đó người chơi sẽ phá huỷ các khối gạch bằng một quả bóng. Em sẽ mô tả chi tiết về cách thiết kế và phát triển game trong các phần tiếp theo của báo cáo này.

Mục tiêu của đề tài là tạo ra một phiên bản game Brick Breaker tương đương với các phiên bản trên nhiều nền tảng khác nhau, đáp ứng được nhu cầu giải trí của người chơi, và cung cấp cho người chơi những trải nghiệm tuyệt vời nhất. Em sẽ đặt ra các thử thách và nhiệm vụ khác nhau cho người chơi, giúp trò chơi trở nên phong phú và thú vị hơn.

2. Công nghệ sử dụng

Trong quá trình phát triển game Brick Breaker, em đã sử dụng các công nghệ sau:

- Java: Ngôn ngữ lập trình chính được sử dụng để phát triển game.
- Eclipse: Công cụ phát triển được sử dụng để lập trình game.
- Git: Hệ thống quản lý phiên bản được sử dụng để quản lý code của game.
- MySQL: Là hệ thống quản lý cơ sở dữ liệu được sử dụng để lưu trữ dữ liệu của game.

3. Thiết kế game

Các thành phần của game

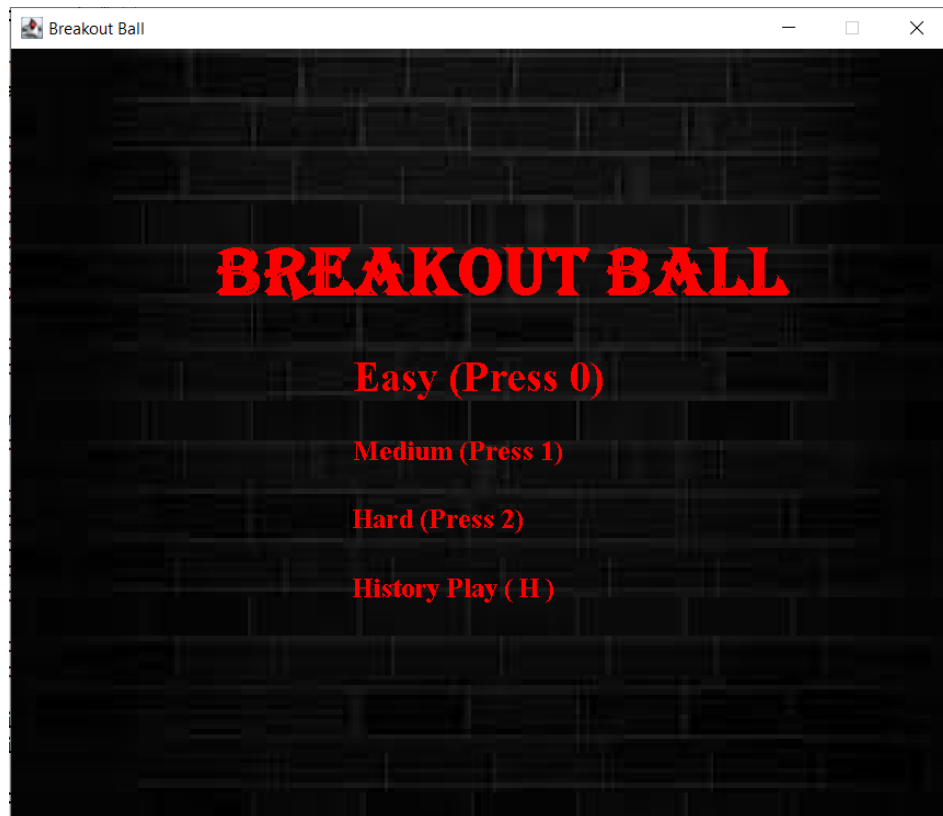
- Ball: Quả bóng di chuyển trên màn hình và va chạm với các khối gạch.
- Paddle: Thanh chắn di chuyển ngang dưới đáy màn hình để đẩy quả bóng lên trên.
- Brick: Các khối gạch cần phá huỷ để hoàn thành màn chơi.

Cách thức hoạt động của game

- Ban đầu, quả bóng được đặt ở giữa màn hình và thanh chắn được đặt ở dưới cùng.
- Người chơi di chuyển thanh chắn sang trái hoặc phải để đẩy quả bóng lên trên và phá huỷ các khối gạch.
- Nếu quả bóng rơi xuống đáy màn hình, người chơi sẽ thua.
- Người chơi chiến thắng nếu phá huỷ tất cả các khối gạch trên màn hình.

Giao diện người dùng và các tính năng của game

- Màn hình chính: Chứa các nút để bắt đầu trò chơi với ba mức độ (easy, medium, hard), xem lịch sử chơi.



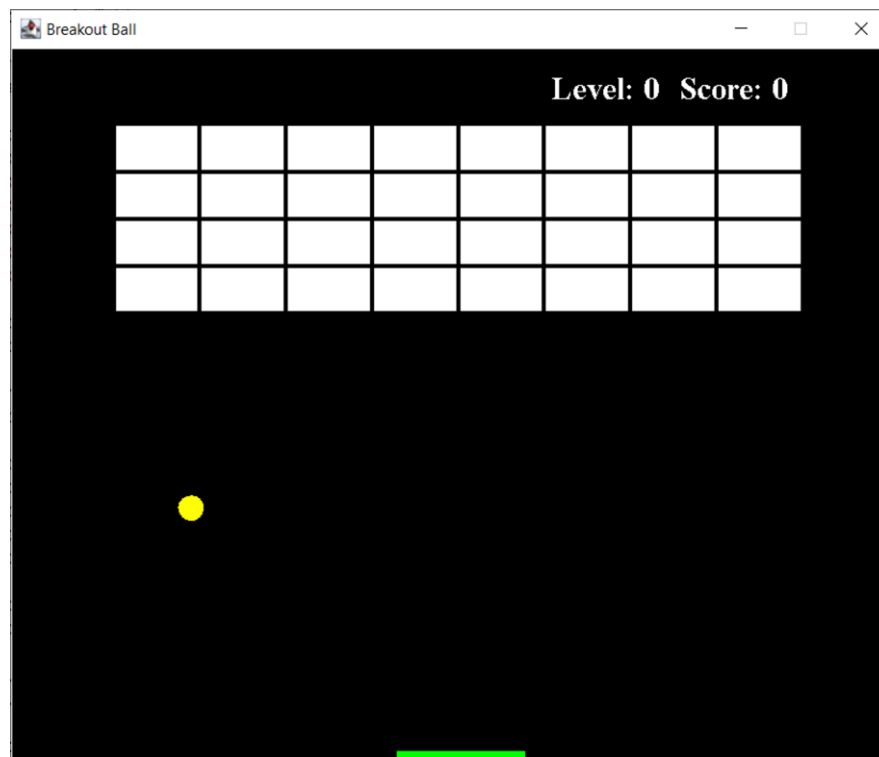
- Màn hình lịch sử chơi: Chứa các thông tin như: Thời gian chơi, điểm số, mức độ, kết quả.

No.	Date	Level	Score	Result
01	26/04/2023 10:44:30	1	155	Lose
02	26/04/2023 10:42:57	1	100	Lose
03	26/04/2023 10:42:13	2	70	Lose
04	26/04/2023 10:41:57	2	50	Lose
05	26/04/2023 10:41:19	2	30	Lose
06	26/04/2023 10:41:08	2	50	Lose
07	26/04/2023 10:38:11	0	40	Lose
08	26/04/2023 10:37:46	0	155	Lose
09	26/04/2023 10:36:47	0	15	Lose
10	26/04/2023 10:36:00	0	160	Win

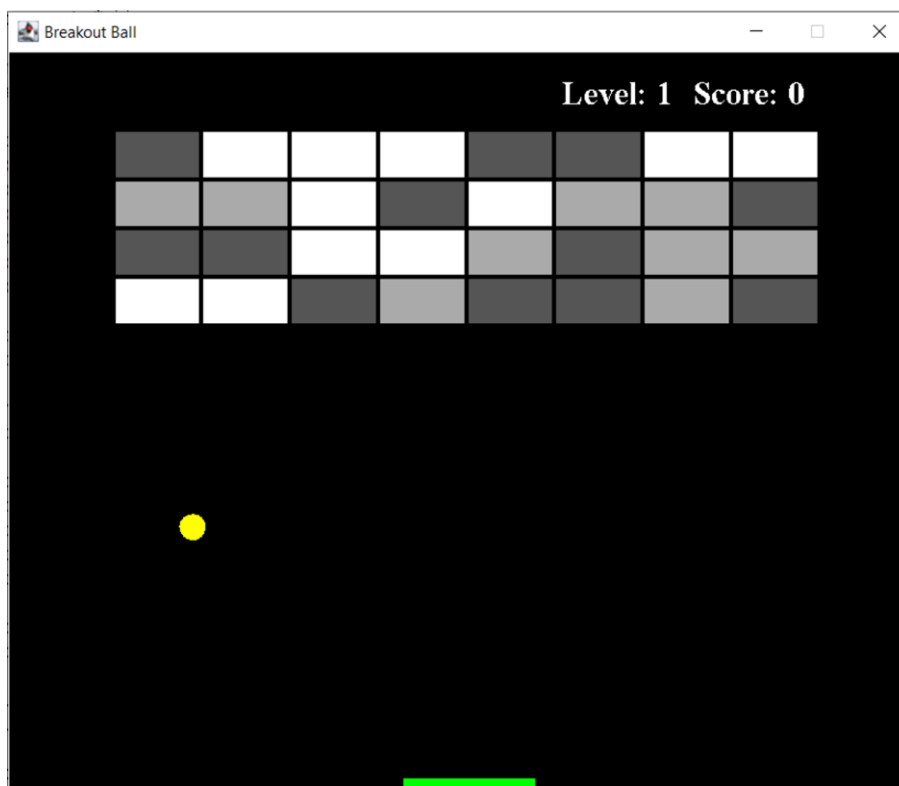
<-- 01 -->

- Màn hình chơi: Hiển thị quả bóng, thanh chắn và các khối gạch với ba mức độ từ dễ đến khó.

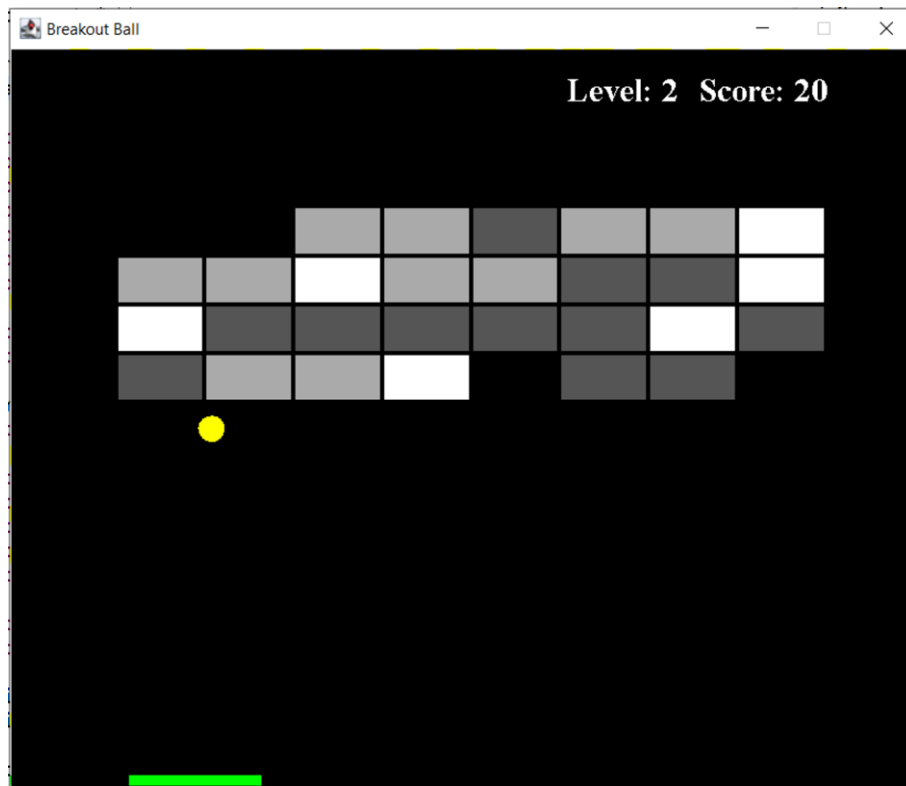
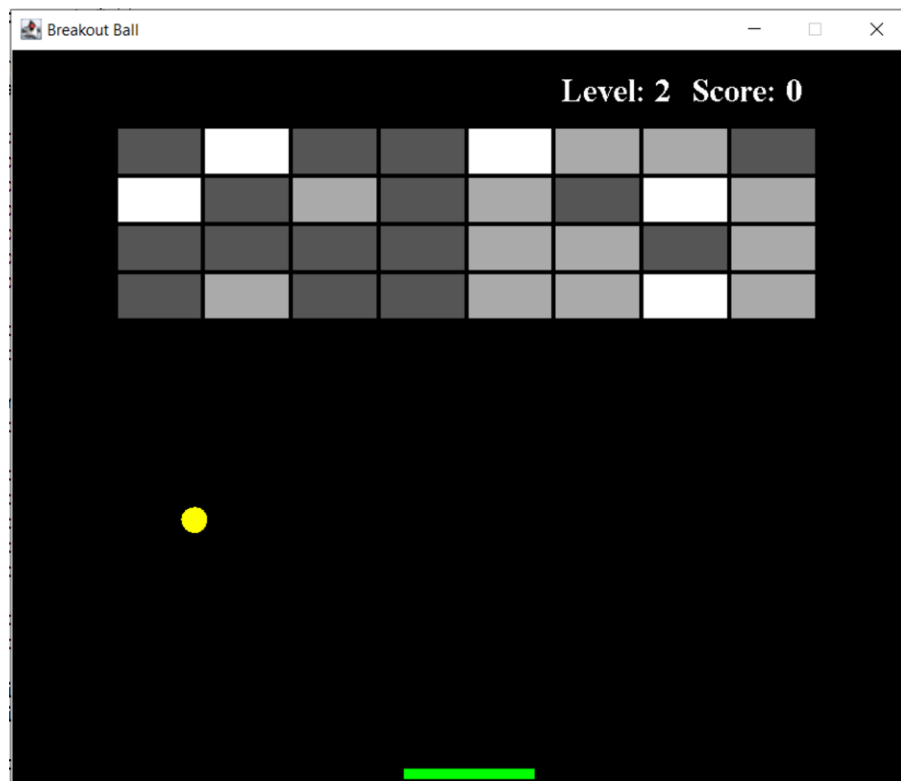
Level 0 (Easy): Mỗi viên gạch sẽ bị phá hủy sau mỗi một lần chạm.



Level 1 (Medium): Mỗi viên gạch sẽ bị phá hủy sau ít nhất một lần chạm và tối đa ba lần chạm. Mỗi lần chạm viên gạch sẽ đổi màu tùy theo giá trị của viên gạch đó.



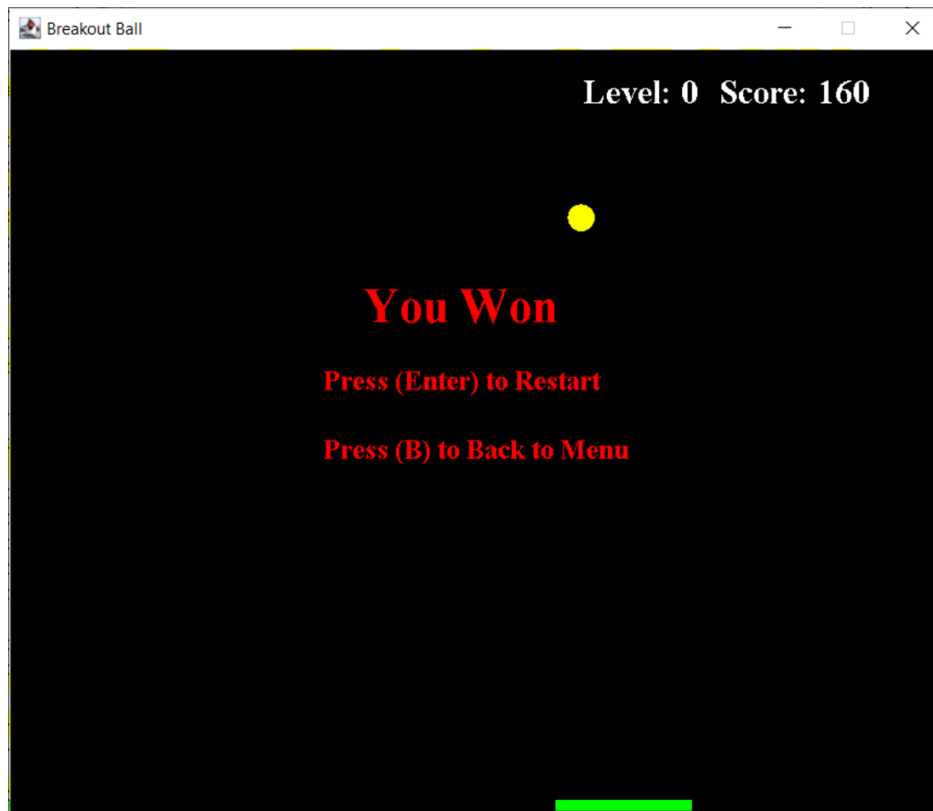
Level 2 (Hard): Mỗi viên gạch sẽ bị phá hủy sau ít nhất một lần chạm và tối đa ba lần chạm. Mỗi lần chạm viên gạch sẽ đổi màu tùy theo giá trị của viên gạch đó (tương tự level 1). Nhưng sau mỗi 12s toàn bộ khối gạch sẽ di chuyển xuống một khoảng nhất định. Nếu có một trong toàn bộ khối gạch chạm vào vạch dưới màn hình thì người chơi sẽ thua.



- Màn hình dừng: Hiển thị nút để tiếp tục, chơi lại hoặc quay lại màn hình chính.



- Màn hình kết thúc: Hiển thị điểm số và các nút để chơi lại hoặc quay lại màn hình chính.





Các quyết định thiết kế game

- Tạo các màn chơi khác nhau với độ khó tăng dần để giữ cho người chơi quan tâm và đảm bảo tính thử thách của trò chơi.
- Thêm các tính năng thú vị để mang lại trải nghiệm mới cho người chơi.
- Sử dụng giao diện đơn giản và thân thiện với người dùng để giúp người chơi dễ dàng điều khiển trò chơi.

Các vấn đề cần giải quyết trong quá trình thiết kế game

- Điều chỉnh độ khó của mỗi màn chơi để đảm bảo tính thử thách phù hợp với người chơi.
- Tối ưu hóa hiệu suất trò chơi để đảm bảo trải nghiệm chơi tốt nhất cho người chơi.

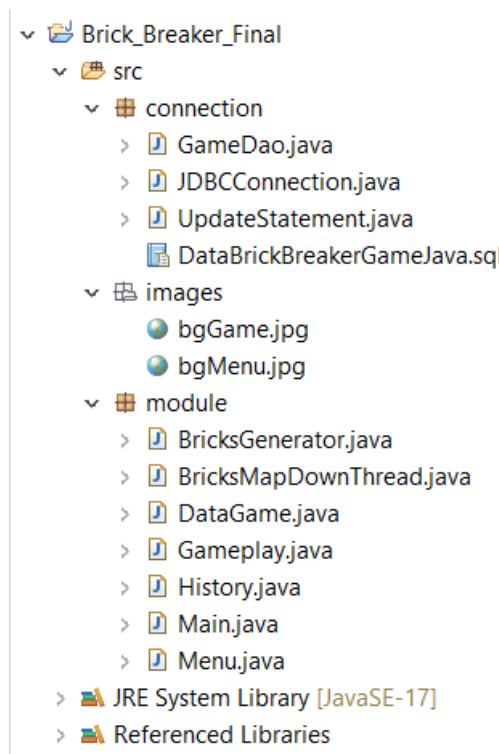
4. Phát triển game

Các bước thực hiện

Để phát triển game Brick Breaker, em đã thực hiện các bước sau:

1. Thiết kế giao diện người dùng: Thiết kế các màn hình chính, lịch sử chơi, màn hình chơi và kết thúc để hiển thị các thành phần của trò chơi, bao gồm quả bóng, thanh chắn và các khối gạch.
2. Lập trình các thành phần của game: Các thành phần của trò chơi, bao gồm quả bóng, thanh chắn và các khối gạch. Em đã sử dụng Java và các thư viện đồ họa để lập trình các thành phần này.
3. Thiết kế các màn chơi: Các màn chơi khác nhau với độ khó tăng dần để giữ cho người chơi quan tâm và đảm bảo tính thử thách của trò chơi.
4. Tối ưu hóa hiệu suất: Tối ưu hóa hiệu suất của trò chơi để đảm bảo trải nghiệm chơi tốt nhất cho người chơi.

Các thành phần có trong chương trình game Brick Breaker



- **Referenced Libraries:** Chứa file .jar mysql connector để kết nối Java với mysql
- **JRE System Library:** Chứa các thư viện của Java
- **Src:** Source code chính tạo nên chương trình

- **Package connection:**

Class JDBC: Dùng để kết nối với SQL

Class GameDao: Dùng để đọc dữ liệu từ sql vào Java

Class UpdateStatement: Dùng để insert dữ liệu từ Java vào sql

- **Package images:** Chứa các file ảnh của game

- **Package module:**

Class BricksGenerator: BricksGenerator là lớp để vẽ khối gạch ra màn hình bằng class Graphics2D trong Java. Với giá trị truyền vào tương ứng, giá này sẽ quyết định số lần chạm để phá hủy mỗi viên gạch.

```
▼ BricksGenerator
  • map : int[][]
  • brickWidth : int
  • brickHeight : int
  • brickColorValue : int
  • BricksGenerator(int, int, int)
  • draw(Graphics2D, int) : void
  • setBrickValue(int, int, int) : void
```

Class BricksMapDownThread: BricksMapDownThread là một lớp kế thừa Thread. Em tạo ra một biến đếm bricksMapDown, biến này sẽ tăng 1 đơn vị sau mỗi 12s. Khoảng cách của khối gạch so với lề trên sẽ tăng lên tỉ lệ thuận với biến bricksMapDown của lớp này.








```
▼ BricksMapDownThread
  ▪ bricksMapDown : int
  ▪ delay : int
  ▪ isPlay : boolean
  • BricksMapDownThread(int, int)
  • run() : void
  • stopThread() : void
  • getBricksMapDown() : int
```

Class DataGame: DataGame là 1 cấu trúc dữ liệu của game. Mỗi khi game kết thúc, dữ liệu của game sẽ được lưu vào database theo dạng cấu trúc này.

```
▼ DataGame
  ▫ timePlayGame : String
  ▫ levelGame : int
  ▫ score : int
  ▫ result : int
  ● getTimePlayGame() : String
  ● setTimePlayGame(String) : void
  ● getLevelGame() : int
  ● setLevelGame(int) : void
  ● getScore() : int
  ● setScore(int) : void
  ● getResult() : int
  ● setResult(int) : void
```

Class Gameplay: Gameplay là lớp tạo ra màn hình chơi bao gồm các thành phần như: quả bóng, thanh chắn, khối gạch và màn hình kết thúc mỗi khi người chơi thắng hoặc thua.

```

▼  Gameplay
  ▫ level : int
  ▫ gameDisplay : boolean
  ▫ isPlay : boolean
  ▫ isDisplayMessPause : boolean
  ▫ isBricksMapDownRun : boolean
  ▫ isBricksMapDownRun1 : boolean
  ▫ dataInsertOnece : boolean
  ▫ score : int
  ▫ totalBricks : int
  ▫ timer : Timer
  ▫ delay : int
  ▫ playerX : int
  ▫ ballposX : int
  ▫ ballposY : int
  ▫ ballXdir : int
  ▫ ballYdir : int
  ▫ brickValue : int
  ▫ bricksMapDown : int
  ▫ bricksMap : BricksGenerator
  ▫ bricksMapDownThread : BricksMap
  ▫ dataGame : DataGame
  ●  Gameplay()
  ● setLevelGame(int) : void
  ● gameStart() : void
  ●  paint(Graphics) : void
  ● isLose() : boolean
  ● insertData(int) : void
  ●  keyPressed(KeyEvent) : void
  ●  keyReleased(KeyEvent) : void
  ●  keyTyped(KeyEvent) : void
  ● moveRight() : void
  ● moveLeft() : void
  ● isGameDisplay() : boolean
  ● setGameDisplay(boolean) : void
  ●  actionPerformed(ActionEvent) : voi

```

Class History: History là lớp tạo ra để hiển thị màn hình xem lịch sử chơi. Trên màn hình sẽ hiện ra các cột thông tin như thời gian chơi, mức độ, điểm và kết quả.

```

v [C] History
    □ pageIndexCurrent : int
    □ pageIndexMax : int
    □ pageNumber : int
    □ historyDisplay : boolean
    □ dataGetOnece : boolean
    □ data : List<DataGame>
    ●c History()
    ● isHistoryDisplay() : boolean
    ● setHistoryDisplay(boolean) : void
    ● isDataGetOnece() : boolean
    ● setDataGetOnece(boolean) : void
    ● paintHistory(Graphics) : void
    ● handleKeyPress(KeyEvent) : void

```

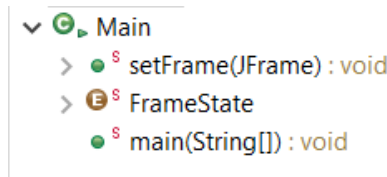
Class Menu: Menu là lớp tạo ra để hiển thị màn hình chính với các nút để chọn các mức chơi hoặc xem lịch sử chơi.

```

v [C] Menu
    □ level : int
    □ valueFont : double[]
    □ backGround : Image
    □ menuDisplay : boolean
    □ history : History
    ●c Menu()
    ● getLevel() : int
    ● isMenuDisplay() : boolean
    ● setMenuDisplay(boolean) : void
    ●▲ paint(Graphics) : void
    ●▲ keyPressed(KeyEvent) : void
    ●▲ keyTyped(KeyEvent) : void
    ●▲ keyReleased(KeyEvent) : void

```

Class Main: Main là lớp để thực hiện chương trình. Trong hàm main của lớp này, em đã tạo ra 2 JFrame là màn hình chính và màn hình chơi game. Để người chơi có thể thao tác di chuyển giữa 2 màn hình này, em đã tạo ra 1 vòng while để có thể luôn luôn lắng nghe các sự kiện mỗi khi người chơi đang ở màn hình chính hoặc màn hình chơi game.



Các kỹ thuật lập trình được sử dụng trong quá trình phát triển

Trong quá trình phát triển game Brick Breaker, em đã sử dụng các kỹ thuật lập trình sau:

- Sử dụng lập trình hướng đối tượng để thiết kế các thành phần của trò chơi.
- Sử dụng Java để lập trình các thành phần của trò chơi.
- Sử dụng các thư viện đồ họa của Java để hiển thị các thành phần của trò chơi trên màn hình.
- Sử dụng Git để quản lý phiên bản của code của trò chơi.

5. Kiểm thử và đánh giá game

Bài tập lớn của em tập trung vào việc phát triển game Brick Breaker. Mục tiêu của đề tài là tạo ra một phiên bản game Brick Breaker tương đương với các phiên bản trên nhiều nền tảng khác nhau, đáp ứng được nhu cầu giải trí của người chơi, và cung cấp cho người chơi những trải nghiệm tuyệt vời nhất.

Quá trình kiểm thử game

Một số vấn đề gặp phải trong quá trình kiểm thử game, bao gồm:

- Hiệu suất của trò chơi không đạt yêu cầu: Trong quá trình kiểm thử, em đã phát hiện rằng trò chơi chạy chậm trên một số máy tính. Em đã giải quyết vấn đề này bằng cách tối ưu hóa hiệu suất của trò chơi.
- Khó khăn trong việc thiết kế các màn chơi: Việc thiết kế các màn chơi khác nhau với độ khó tăng dần là một thách thức rất lớn. Em đã giải quyết vấn đề này bằng cách thử nghiệm và điều chỉnh các màn chơi để đảm bảo tính thử thách phù hợp với người chơi.

Các kỹ thuật kiểm thử được sử dụng

Trong quá trình kiểm thử game, em đã sử dụng các kỹ thuật kiểm thử sau:

- Kiểm thử đơn vị: Kiểm thử các thành phần của trò chơi, bao gồm quả bóng, thanh chắn và các khối gạch.
- Kiểm thử tích hợp: Kiểm thử tích hợp các thành phần của trò chơi để đảm bảo tính tương thích và tính ổn định của trò chơi.
- Kiểm thử hệ thống: Kiểm thử toàn bộ hệ thống của trò chơi để đảm bảo tính thử thách và tính ổn định của trò chơi.

Đánh giá chất lượng của game

Em đã đánh giá chất lượng của trò chơi dựa trên các tiêu chí sau:

- Tính thú vị của trò chơi: Tính thú vị của trò chơi được đánh giá dựa trên độ khó, tính thử thách và tính đa dạng của các màn chơi.
- Tính ổn định của trò chơi: Tính ổn định của trò chơi được đánh giá dựa trên số lượng lỗi và sự cố trong quá trình chơi.

KẾT LUẬN

Bài tập lớn này tập trung vào việc phát triển game Brick Breaker. Mục tiêu của đề tài là tạo ra một phiên bản game Brick Breaker tương đương với các phiên bản trên nhiều nền tảng khác nhau, đáp ứng được nhu cầu giải trí của người chơi, và cung cấp cho người chơi những trải nghiệm tuyệt vời nhất.

Em đã thiết kế các màn chơi khác nhau với độ khó tăng dần để giữ cho người chơi quan tâm và đảm bảo tính thử thách của trò chơi. Em đã thêm các tính năng thú vị để mang lại trải nghiệm mới cho người chơi. Em đã sử dụng giao diện đơn giản và thân thiện với người dùng để giúp người chơi dễ dàng điều khiển trò chơi.

Để phát triển game, em đã lập trình các thành phần của trò chơi, bao gồm quả bóng, thanh chắn và các khối gạch. Em đã sử dụng Java và các thư viện đồ họa để lập trình các thành phần này. Trong quá trình kiểm thử, em đã giải quyết các vấn đề về hiệu suất của trò chơi và khó khăn trong việc thiết kế các màn chơi khác nhau với độ khó tăng dần.

Về phần đánh giá tính thú vị và tính ổn định của trò chơi. Em đã đánh giá tính thú vị của trò chơi dựa trên độ khó, tính thử thách và tính đa dạng của các màn chơi. Em đánh giá tính ổn định của trò chơi dựa trên số lượng lỗi và sự cố trong quá trình chơi.

Trong báo cáo này, em đã trình bày về quá trình phát triển game Brick Breaker bằng Java, từ việc thiết kế đến phát triển và kiểm thử. Em đã mô tả chi tiết về cách hoạt động của game và các tính năng của game, bao gồm các cấp độ khác nhau và giao diện người dùng. Tuy nhiên, game còn một số hạn chế và em cần giải quyết các vấn đề về tối ưu hóa hiệu suất và thiết kế các màn chơi phù hợp hơn với nhu cầu giải trí của người chơi. Tuy nhiên, em tin rằng với các định hướng phát triển tiếp theo, em có thể cải thiện trò chơi để đem lại trải nghiệm tốt nhất cho người chơi. Bài tập lớn này đã giúp em rèn luyện và cải thiện kỹ năng lập trình của mình, đồng thời cũng giúp em hiểu sâu hơn về quá trình phát triển phần mềm bằng Java.

TÀI LIỆU THAM KHẢO

1. Java Documentation Oracle - <https://docs.oracle.com/en/java/>
2. Java Tutorial W3Shool - <https://www.w3schools.com/java/default.asp>
3. Java Tutorial Zetcode - <https://zetcode.com/all/#java>