|  |  |
| --- | --- |
| Assignment Case | Description: LogoBINUS-University |
| JavaH1Special |
| **Periode Berlaku** Semester Ganjil 2023/2024  ***Valid on*** *Odd Year 2023/2024* | **Software Laboratory Center**  **Assistant Recruitment 24-1** |

## Materi

*Material*

* Object Oriented Programming (OOP)
* SOLID Principle
* MySQL

## Soal

*Case*

**AX For AutoPets**

**AX For AutoPets** is an engaging online auto battler game where players strategically assemble and manage teams of diverse animals, each with unique abilities, to compete in automated battles. This innovative game aims to provide a captivating experience, combining strategy and fun as players build their ultimate teams to outsmart and outlast their opponents. The development of **AX For AutoPets** will utilize the **Java Programming Language** with **Object Oriented Programming** concepts such as **Encapsulation**, **Inheritance**, and **Polymorphism**, ensuring a robust and flexible gaming experience. Your mission as a programmer is to create this exciting game, focusing on seamless gameplay, balanced mechanics, and an enjoyable user experience.

* **Design Pattern** :
* Singleton

You must implement Singleton design pattern in your code. It is recommended to use it to make the database connection for the application.

* Strategy

Strategy can be implemented for many cases but it is recommended to use it to make the mechanism for the pets and foods. The pet’s abilities could also be made using strategy design pattern.

* **Database ERD** :

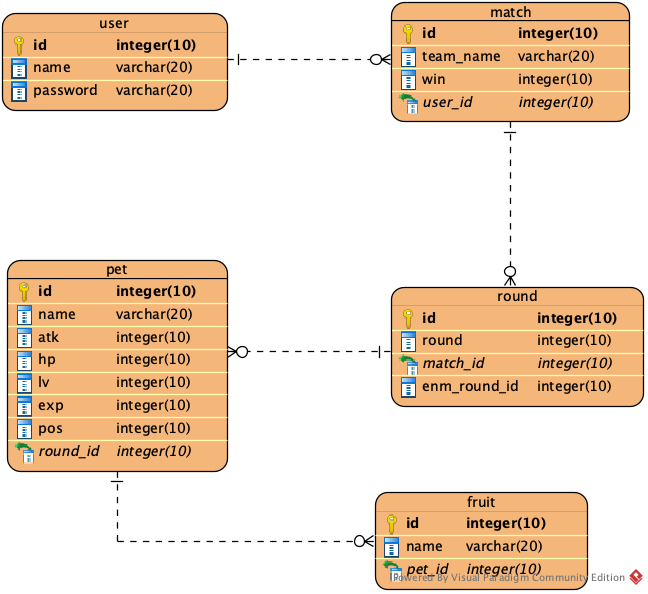
****

Figure 1. Database ERD

* **Auth Page**



Figure 2. Auth Page

* In the beginning, the program will show **3 menus**, which are :

1. **Login**
2. **Register**
3. **Exit**

* If the user chooses **Login (Menu 1) then** :
  + The program will ask the user to input the **username** and **password**.
  + The username and password must be in range **5 – 20 characters** or else it will keep prompting.
  + Then, **validate** the **credentials that** must **exist** in the **database**.
  + If the **credential** **does not exist**, then **show an error message**.
  + **Otherwise**, **redirect** the user **to menu based on their role**.
  + **Save the currently logged in user** in the application, this will be useful in other menus to perform certain tasks.



Figure 3. Login

* If the user choose **menu 2 (“Register”),** then:
  + Ask the user to input the **username**. Validate that the username **length** must be **between 5 and 20** **characters** **(inclusive)**.
  + Validate that the username must also be **unique**.
  + Ask the user to input the **password**. Validate that the password **length** must be **between 5 and 20** **characters** **(inclusive)**.
  + If the **validation** is **successful**, then **insert the new user into the database**.



Figure 4. Register New User

* If the user choose **menu 3 (“Exit”)**, then the **program will be closed.**
* **Main Page**

****

Figure 5. Main Page

* This page consist of **4 menus**, which are :

1. **Arena**
2. **Leaderboard**
3. **History**
4. **Logout**

* **Game (Arena Page)**

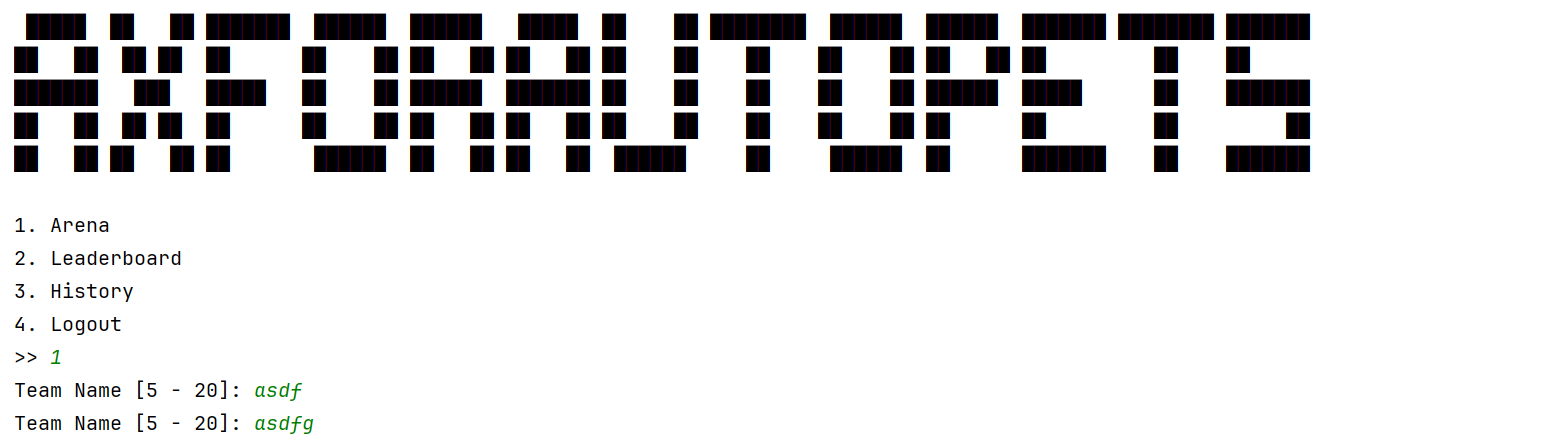
****

Figure 6. Team Prompt

When the **user first choose to go to the arena**, the user will be **prompted** to enter the **team name** to enter the arena. This name will be the **team name for the pets** and will appear when fighting other teams.

The arena has various attributes, below are the details for each of them:

* **Round**

This attribute **indicates** at what round you are in right now. Round will **increase** as you **progress through the game**. Each round will **consist of 2 phase**, namely the s**hopping phase and the battle phase**. Round will also **affect** your shop which will be **explained** in the shop section.

* **Life**

When **your life reaches 0** the game **will end**, and you will be **redirected back to the main page**. Your life will **decrease by 1 everytime you lose a battle**. At the start of the game **you will start with 5 life**.

* **Win**

Everytime you win a battle **your win will increase by 1**. This attribute is to **keep count** of **how many win you got each match**. This attribute will be used in **the leaderboard section of the game** to **compete** with other players. When you reach **10 wins** then the **match will end** and you will be redirected back to the main page.

* **Money**

You will **get 10 money** everytime you **enter shopping phase**. This **may increase using pet abilities or by selling pets**. This will be used to buy pet or food in the shop. When you leave shop the leftover money will not be saved for the next shop.

* **Shop Page**

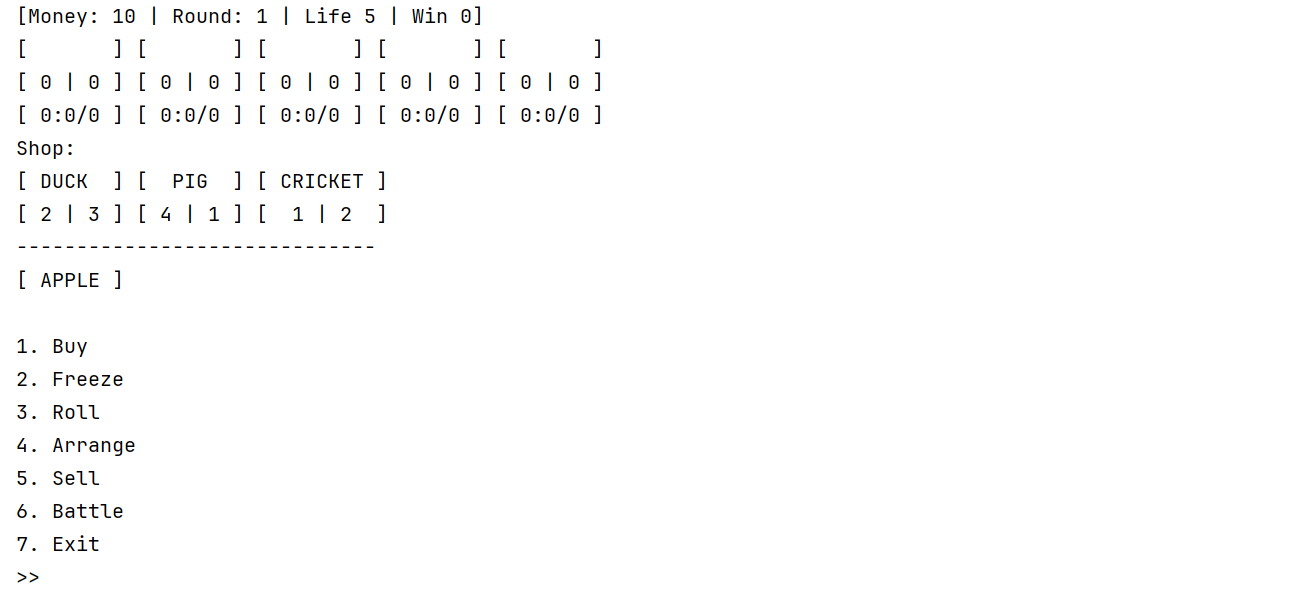


Figure 7. Shop Page

The shop page will **show info** about the player’s current state, namely the **money, round, life, and win**. Below it will be the player's team which will be displayed in **5 slots** which will be **filled with pets**. Every pet will **display info** about its **name, stats, and experience**.

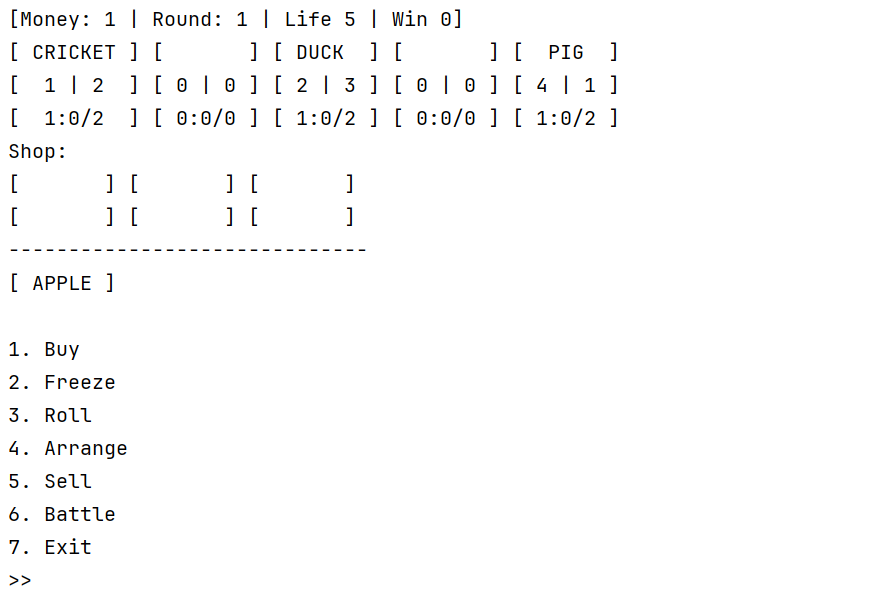


Figure 8. Example display for pet

The **Shop has 6 levels**. The level progresses alongside the round of the match. The shop will **level up from 1 to 6 every 2 rounds.** For example, the shop will be level 1 at round 1 but will level up to level 2 after round 3 or at the start of round 4. The **shop level affects the item** that will be displayed at the shop. Each level will **correspond to the same tier** for the pet and the food. For example, shop level 1 will show tier 1 items at the shop and level 2 shop will show tier 2 item at the shop. Your job is to make sure that you **make the pets and foods up to tier 3**, you may make it to tier 6 if you have the time however it will be optional. For the pets and the food you must make you may **refer to this wiki link**:

* <https://superautopets.wiki.gg/wiki/Turtle_Pack>

For each pet you must make sure to **create the ability correctly**, every **ability will have 3 levels** that will **progress alongside the pet’s leve**l. Every pet also have **different trigger** for their ability so please pay attention to that as well. Every pet can eat an **infinite amount of food** however if the food has perk, then the **perk will attatch to the pet** and **replace previously atttatched perk**. Perk **functions just like pet ability**, it has a trigger and an effect that will activate when the trigger is triggered.

The shop page consists of **7 menus** each will be explained as their own page below this section. The 7 menus are:

1. **Buy**
2. **Freeze**
3. **Roll**
4. **Arrange**
5. **Sell**
6. **Battle**
7. **Exit**

* **Pet Level**

Pet will all have **default level of 1** and will require 2 exp to level up which will be displayed as **“1:0/2”.** Every pet will **only have 3 level**. Below is the detailed explanation of exp required to level up for each level:

* **Level 1**

You will need **2 exp** to level up from level 1 to level 2.

* **Level 2**

You will need **3 exp** to level up from level 2 to level 3.

* **Level 3**

Level 3 is the **maximum level** and you won’t be able to level up past level 3.

A screenshot of a computer

Description automatically generated

Figure 9. Merge pet

The example above shows what will happen when you **merge a pet**. As you can see the pigeon level stat is **“1:1/2”** that means the pigeon is level 1 and has an exp of 1 from 2. The pigeon require 1 more exp to level up to “2:0/3”. You can also **merge a leveled pet with another leveled pet**.

A computer screen shot of a computer code

Description automatically generated

Figure 10. Before merge

A screenshot of a computer

Description automatically generated

Figure 11. Merging pet slot 4 to 5

When you merge a leveled pet then you must take the **highest stat** for the hp and atk and **increment it by 1**. You must also **sum the exp** of both pets. Leveling or merging pet does not only affect its stats, it also **affects the pet ability**. The pet’s ability will level up alongside the pet and gradually gets **stronger each level**.

* **Buy Page**

A white background with black text

Description automatically generated

Figure 12. Buy Page

When the user enters buying page **display all the available pets and food** that the player can choose to buy. Make sure to also **display the stat** for each pets. Also pay attention that each slot is printed with **center alignment**. There is a rule for the shop slots, the **shop slots will increase alongside the shop level**. Below is the detail for the shop slots for each shop level:

* **Level 1**

3 pet slots, 1 fruit slot

* **Level 2**

3 pet slots, 1 fruit slot

* **Level 3**

4 pet slots, 2 fruit slots

* **Level 4**

4 pet slots, 2 fruit slots

* **Level 5**

5 pet slots, 2 fruit slots

* **Level 6**

5 pet slots, 2 fruit slots

A screenshot of a computer

Description automatically generated

Figure 13. Shop after buy pet

When you buy a pet or a food you must first **choose which pet or food** you want to buy then you must **choose which slot** do you want to place it in. In the example above I demonstrated placing a cricket at slot 5. **After you buy pet or food the slot will be empty** and will remain that way until it is regenerated either by rerolling or proceeding to new shop.

A screenshot of a computer code

Description automatically generated

Figure 14. Validate empty slot

You must validate that the player **can’t buy empty slots**. You must also make it so that when the player **buy frozen slot the slot will be back to normal again**. You must also validate when the player is trying to buy shop items then the player must **type either Pet or Food** which must be **case sensitive.** Then prompt the player to **choose within the slot range** for example, in the picture above there are only 3 slots then the player may only choose between 1 to 3 inclusive.

A white background with black text

Description automatically generated

Figure 15. Gold validation

You must validate that the player must have **atleast 3 golds** to purchase item from shop. If the player does not have the required amount then just **print the error message** and **redirect the player back to the shop menu.**

A screenshot of a computer

Description automatically generated

Figure 16. Before buy merge

A screenshot of a computer

Description automatically generated

Figure 17. After buy merge

In the example above I just showed that you can **choose to instantly merge the pet** just as you buy it, thus requiring no additional slots. You must be able to **validate** that you can only **merge the pet of the same type** when you want to merge buy it.

A computer screen shot of a computer code

Description automatically generated

Figure 18. Buy error message

When you buy pet you must validate that you can only merge buy it with the **same pet type.** Or if you want to place it in your team then you must validate that the **target slot is empty**, else then you **must show an error message** to let the user know.

* **Freeze Page**

A white screen with black text

Description automatically generated

Figure 19. Freeze Page

When you enter the freeze page then you must prompt the user to **choose either to freeze pet or food.** After that then you must prompt the user to **choose the slot to freeze** the option is **based on the amount of slots available** in the shop.

A screenshot of a computer

Description automatically generated

Figure 20. After freeze

After freezing the shop item then you **must change the brackets** from square brackets to curly brackets. The brackets will **stay curly as long as the slot is frozen**, it will only go back to normal once the slot is back to normal.

A screenshot of a computer program

Description automatically generated

Figure 21. Reroll frozen slot

When you reroll the **frozen slot will not be rerolled** and will stay the same as long as it is still frozen. **Freezing a slot does not cost you anything**. You can unfroze a slot by either choosing to freeze the same slot or by buying the frozen slot.

* **Roll Page**

**A black text on a white background

Description automatically generated**

Figure 22. Reroll gold validation

When you choose to reroll then you will **regenerate all the shop slots** except the frozen slots. Rerolling **costs the player 1 gold**. You must validate that the player has **atleast 1 gold** to be able to reroll.

* **Arrange Page**

**A close up of a number

Description automatically generated**

Figure 23. Arrange page

When the player choose the **arrange menu** then the player will be prompted to **choose 2 slots.** You must validate the chosen slot must be **between 1 to 5 inclusive**. From the example above since we choose to arrange a different pet (pigeon and beaver) then the **pet will swap slots.**

A screenshot of a computer

Description automatically generated

Figure 24. Swapped pet

However using the arrange menu you can do more than just swapping slots. When you choose to arrange pet of the same kind then it will **merge as long as it is not max level** and can still level up.

A screenshot of a computer

Description automatically generated

Figure 25. Merge pet

As you can see from the example above when I choose slot 3 and slot 5 then the pigeon from slot 3 will merge with the pigeon on slot 5 since they are of the same type and pigeon on slot 5 can still level up. Since the pet merged the stats of both the hp and the **atk will increase by 1**. The pet exp will also increase as you can see **from 0/2 to 1/2**.

* **Sell Page**

**A black text with green and blue letters

Description automatically generated**

Figure 26. Sell page

When the player chooses the sell menu then you must prompt the user **to choose which slot does the user want to sell**. Then after picking the slot to sell then you must **sell the pet on the selected slot** and **display it using a message to the user**. Then the next time the team is printed the **slot that just got sold must be empty** since the pet there has been sold. The **default selling price for every pet is 1 coin**. However this will **increase alongside the pet’s level.**

A white background with black and white clouds

Description automatically generated

Figure 27. Sell Level 2 pet

In the example above we just sold a level 2 pet, and it sold for 2 coins. The **amount of coins** that the player will receive from selling the pet **will be the same as the pet’s level**.

A white background with black dots

Description automatically generated

Figure 28. Sell ability

There are some abilities that **activates by being sold**. In the example above we see that selling a duck will buff all the shop pets for 1 hp. You must make sure that you **implement all** the selling abilities correctly and that it **activates upon being sold.**

* **Battle Page**

When battling the first thing you must do is to **arrange the team**. In the shopping phase the player might not fill every available slots, thus making holes in the team. You need to **move all pet to the front** and leave the **empty slots behind**. Here are some examples:

A screenshot of a computer program

Description automatically generated

Figure 29.Team display before battle

A screenshot of a computer code

Description automatically generated

Figure 30. Team display when battle

As you can see from the example shown above, when the teams enter the battle phase then immedeately **every pet gets arranged to fill the front slots first**. Then you must make sure that when the team goes back to shopping phase the team **will go back to how it was arranged** before and not how it was arranged when in battle phase.

When battling you **can’t generate a random team for the enemy**. The enemy team **must be a team that a player previously played.** For example player A play with a team named “A Team” then when player B plays the game he has a chance to fight player A’s “A Team”. However there is 1 thing that you must make sure of, you must make sure that the “A Team” that you use is appropriate to the current round. For example, when you are fighting in round 1 then you **must get a random team from the database** that is fighting in round 1. This goes for every other round when player fights in round 10 then you must get a random team from round 10 and so on.

A screenshot of a computer code

Description automatically generated

Figure 31. Dsiplay team name

After you **get a random team for the current round from the DB** then you may start the battle. The battle has **3 phase**:

1. **Start**

At the start you must display the team names for the fighting teams just as the example shown above.

1. **Battle**

In this phase the team will fight each other until either one lose or it is a draw.

1. **Conclusion**

You must then conclude the battle either by giving a win, lose, or a draw for the player.

you must **display the team’s name before fighting**. After you display the team name then you will start the battle turn. A turn in a battle will be represented with **2 steps**:

1. **Display team**

In this step you must display both the player and the enemy team fighting both arranged to fill the front slots first.

1. **Display actions**

For this step you must **display every action** that happens on that turn. Wether it is an **ability, attack, damage, etc**. Make sure that every action will be as **detailed** as possible, meaning you must describe **what happens to who and who does it.**

When a pet reaches **0 health it will die**. When a pet dies during battle phase then the next turn you must **display the team without that pet**. Every turn you will **only display living pets** during the battle phase. However they are **not permanently removed** and will be **resetted after every battle**. During battle phase the pet that **attacks will only be the front most pet**, however ability can be activated anywhere disregarding the pet position in the team.

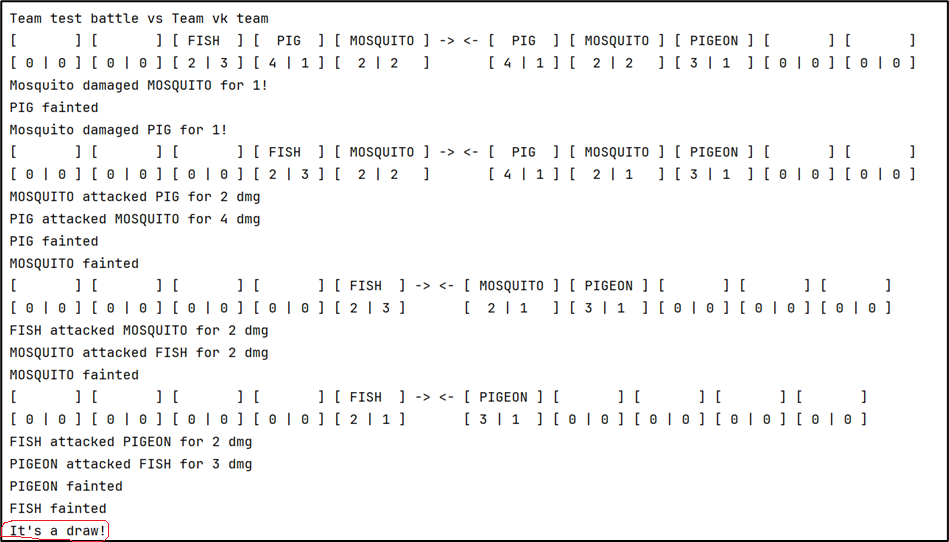


Figure 32. Battle conclusion

After every battle you must **conclude the battle result**. There are **3 types** of conclusion, **draw, win, and lose.** Below will be a more detailed description for every conclusion:

* **Draw**

If both of the team has **no pet left standing** at the end of battle then it will result in a draw. A draw will not cost the player anything and will simply advance the player to the next round.

* **Win**

When the **only pet left standing is on the player’s team** then the player will win. When the player wins it will **increase the number of win for that player in that match**. When the player reaches 10 wins then the match will end.

* **Lose**

The player will lose if the **only pet left standing is the enemy team’s pet**. The player health will **decrease by 1 for everytime the player lose**. When the life reaches 0 then the match will end.

What happens when a match end will be explained in more detail after this on the exit arena section. There is one thing that you must make sure of, for every round you must **save the enemy that is assigned for that round in the database**. That is so that you can **replay** the match and the round later on using the history menu.

* **Exit Arena**

When the player chooses the exit arena menu then the player will be **redirected back to the main menu.** However there is something important that you must do, when a match ends you must **make sure to update the details of that match in the database**.

* **Leaderboard Page**

****

Figure 33. Leaderboatd page

In this page your job is to **display the top 10 players** ordered by their total number of **wins** for all of their match. You must display the leaderboard sorted ordered by the number of wins **from the highest to lowest.**

* **History Page**

**A white background with black and white clouds

Description automatically generated**

Figure 34. History Page

When the user enters this page you must **display the match history** of all the match that the player has played. The match history will be displayed **using the team’s name** and the **number of wins** the match has. The user can then choose to **input -1** to go back or **to input the number of the match** that the user wants to view.

A close up of a computer code

Description automatically generated

Figure 35. Match detail page

When the user **inputs the number of the match** then you must **redirect the user to the match detail page for the chosen match**. In the match detail page you must **display every round** that the match has. After displaying every round for the match you then must prompt the user to either **input -1** to go back to the history page or **to input the round number** that the user wants to replay.

A computer code with many small letters

Description automatically generated with medium confidence

Figure 36. Round replay

When the user inputs the round number that the user wants to replay then you must **replay the battle for the selected round**. Replaying a round **works just like battle round** when you are playing arena, so please **refer to the battle guide** on making the replay section. After you replay the round then **redirect the user back to the match detail page** where the user can then choose to replay another round or just **input -1 to go back** to the history page.

* **Logout Page**

When the user chooses the logout menu then your job is to **logout the currently logged in user** from the application. This will automatically **redirects the user back to the login page** since there is no currently logged in user saved in the application.

Please run the BAT or JAR file to see the sample program.

## Komponen Penilaian

*Scoring Component*

|  |  |  |
| --- | --- | --- |
| No | Component | Weight |
| 1 | OOP Concept | 15% |
| 2 | Design Pattern | 5% |
| 3 | CRUD | 20% |
| 4 | SOLID Principle | 30% |
| 5 | Architectural Design | 8% |
| 6 | Actions | 12% |
| 7 | Validations | 10% |