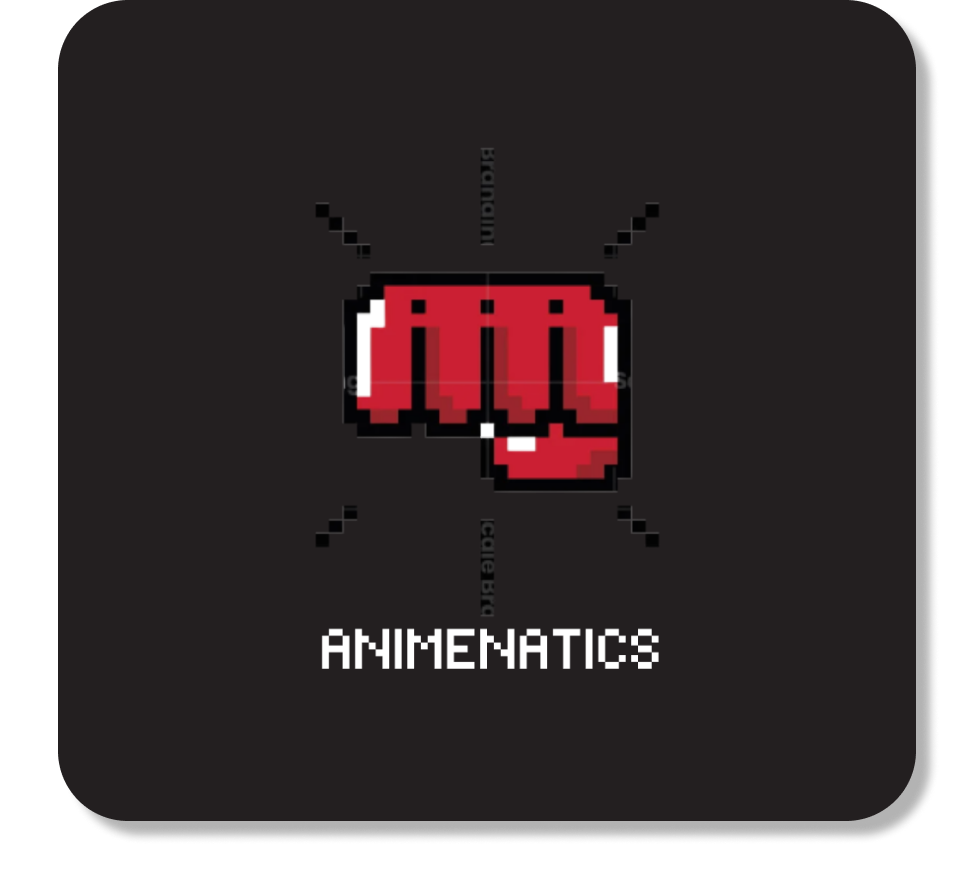
Algorithm & Logic Formulation

Project Idea Proposal

Group 1

* Group members
  + Lumanog Ian
  + Manalo Irish
  + Tomate Alexene
  + Francisco Earl
  + Espinoza Arvin
  + Cunanan Given
* Project outline
  + WEEK 0 Project Flow
  + WEEK 1 Research
  + WEEK 2 Brainstorming
  + WEEK 3 Project Idea Proposal DEC 2 SATURDAY = (?)
  + WEEK 4 Algorithm-Making ( DEC 5 ) SATURDAY = PASS PSEUDO & FL
  + WEEK 5 Development ( DEC 12 )
  + WEEK 6 Development (DEC 19)
  + WEEK 7 Development (DEC 26)
  + WEEK 8 Development ( JAN 2 )
  + WEEK 9 Final Project Presentation
  + WEEK 10 ( ? )
* Instruments / Tools
  + Programming Language: Python
  + Version Control System: Git, Github
  + Data base: MongoDB *removed*
  + API *removed*

**Project Idea 1**

Project Name: Animenatics

About: An originally thought

turn-based anime game that you can play

using the terminal and compete with

the leaderboards to achieve

being the top Anime master.

**Project Design**

* Game Mechanics
  + Animenatics allows its players to pick among a pre-determined pool of anime characters. In this game, players take turns in executing an action based on their chosen character’s skillset.

* + Skillset of the characters will be based on their Anime series.

eg. character: Gojo, skill: domain expansion = Player 2 can’t use any skills for 1 Turn (but can use normal attacks)

* + Each character will have 3 skills, and the cooldown

will be based on turns depending on how powerful that skill is,

(ranked from 1-3) eg. skill 2 = cannot be used in 2 turns.

* + If we can push this project even more, we can add

four different kinds of classes (Warrior, Assassin, Mage, Defender).

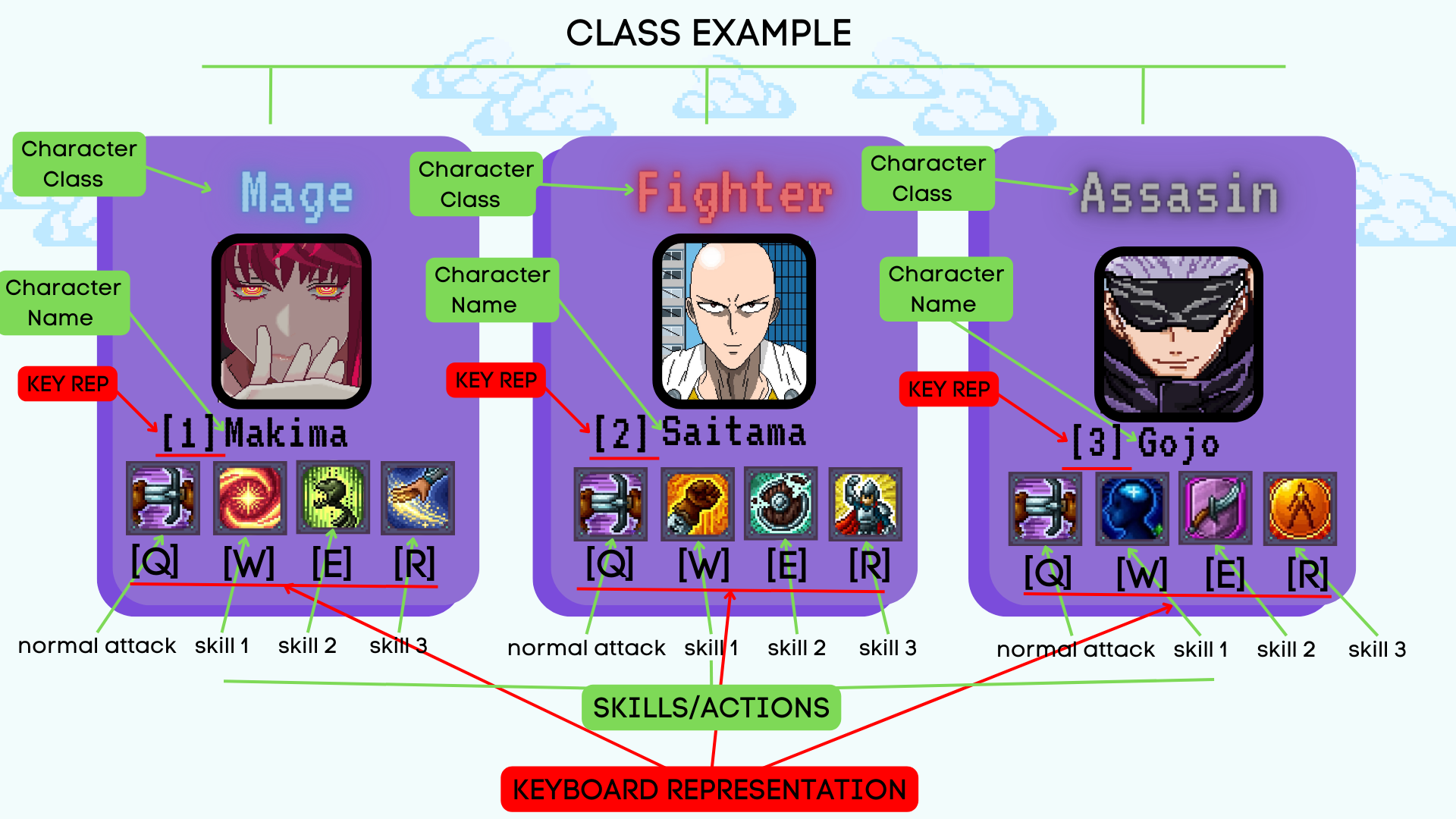
They will have kinds of extra skills based on the class of the anime character (e.g., Defender can taunt, Assassin can %dodge attacks, etc.).

* + Top 20 Arcade inspired point based leaderboard that you can compete with. The total amount of points you will get per win

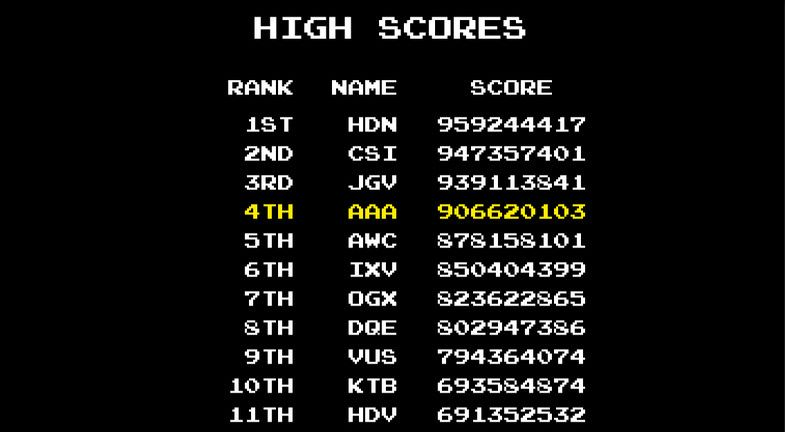
will be based on your characters remaining health times five.

* Visual view of what are we trying to develop

**GAME MECHANICS**



LEADERBOARD



Other information

* Versions
  + version1 1v1 Combat no characters
  + version2 1v1 Combat with 1 character per player
  + version3 1v1 Combat with 4 characters each player
  + version4 1v1 Combat with 4 characters and class
* User Interface
  + Player name
  + Character status
  + Player’s action
  + Last action
  + Character choosing
  + Skill choosing
* Game states
  + Start
  + Random turn
  + Player1 turn
  + Player2 turn
  + WINNER

# dividing the program into game states

* Logic
  + UNIT
    - Characters stats
  + Battle system
    - Algorithm of damages, etc.
    - Game states
    - Cooldown of skills

SAMPLE UI

(will update to more friendly user interface during development)  
+---------------------------------------------------------------+

| ANIME NATICS ANIME NATICS ANIME NATICS ANIME NATICS |  
+---------------------------------------------------------------+

| >>>>>>>>>>>>>>>>>>>>TURN: PLAYER2<<<<<<<<<<<<<<<<<<<<<< |

Player1 name Player2 name

Characters status: Characters status:

[1] Gojo [1] Makima

Health: 1852 Health: 1927

[2] Naruto [2] Denji

Health: 2921 Health: 3259

[3] Goku [3] Saitama

Health: 2425 Health: 2719

[4] Deku [4] Jojo

Health: 1729 Health: 1298

Player1’s action Player2’s action

Characters: [1] [2] [3] [4] Characters: [1] [2] [3] [4]

> (character input) > (character input)

Skills: [Q] [W] [E] [R] Skills: [Q] [W] [E] [R]

> (skills input) > (skills input)

**Project Idea 2**

Project Name: Who wants to be a Singing Bee

About: a Console game where you can test your knowledge about your chosen songs by looking for the correct answer of the missing word/s from its lyrics.

**Project Design**

* Game Mechanics
  + User can choose n songs based on the year: 70’s 80’s 90’s 20’s 2010’s 2020’s. there will be 6 songs per year. = 36 songs
  + The player should guess the missing word on that specific stanza of the chosen song.
  + Player has a 3 hints whole game.
  + Player choice will be: A,B,C,D,H
  + Correct answer without using hint is 1000 points each. With using hint, it’s 500 points each. If your answer is wrong, you get 0 points on that round.
  + After the n songs is done, system will print out the total score of the user, and will ask if the user wants to quit or play again and score should go back again to 0.

**Other information**

* Versions
* version1. Manually adding songs one by one as well as adding a Leaderboard
* version2. Automation of song adding using a Lyric API
* Version3. Implementation of Sound Clips that will play alongside the text as well as a Voice Recognition API to remove the choices altogether
* API’s Used
* Lyrical Construction API - Searches for the song inputted by the user and returns the lyric of the given song which would then be cut into stanzas for use as a lyrical question whilst simultaneously removing a word on the stanza which will serve as the answer for the lyrical question.

Samples of a Lyrical Construction API includes:

* LyricsGenius
* TheAudioDB
* MusixMatch API
* Thesaurus API - Takes in a specified word and returns a list of synonyms and antonyms of the given word. 3 synonyms are taken from the returned list and is added to the 4 choices of possible answers to the lyrical question along with the original specified word that is taken out from the song stanza.

Samples of a Thesaurus API includes:

* WordsAPI
* Lingua Robot
* API Ninja - Thesaurus
* Speech to Text API - Converts audio cues from the user and converts it into a string of text for use in the game. Removes the choices altogether since you only need to say the correct answer and/or say hints to be provided hints.

Samples of a Speech to Text API includes:

* Rev.AI
* AmberScript
* Google Cloud’s Speech-to-Text

SAMPLE UI

(will update to more friendly user interface during development)  
+---------------------------------------------------------------+

| SINGING BEE |  
+---------------------------------------------------------------+

suggestions:

keep ideas back pocket

make it simple

////////////////////Song Playing - Got To Believe In Magic//////////////////////

Take me to your heart

Show me where to start

Let me play the of your first love;

[A] H̶e̶a̶r̶t̶ [C] S̶t̶a̶r̶t̶

[H] Hint

[B] Part [D] Role

+--------------------------------------------------------------------+

| Player Name: uminatics Total score: 0 |

| Player Answer: H Round : 0 |

+--------------------------------------------------------------------+

#output after taking H as input

+---------------------------------------------------------------+

| SINGING BEE |  
+---------------------------------------------------------------

///////////Song Playing - Got To Believe In Magic///////////////////////////////

Got to believe in magic

Tell me how two find each other

In a world that's full of strangers;

[A] Strangers [C] Soulmates

[H] Hint

[B] Eagles [D] People

+--------------------------------------------------------------------+

| Player Name: uminatics Total score: 2 |

| Player Answer: D Round : 0 |

+--------------------------------------------------------------------+

# output after taking D as an input

Fri

sat :

if sir = no classes

meeting 1pm // recap

// recap

if may class

no meeting || final project

end class = 3pm

possible 3pm // recap

algo making monday

// meeting : algo making ( 1-3pm )

// monday 3pm

Monday  
1pm - 4pm locked

9am - 11am algo meet

11am - 12pm ets meet