Game mechanic proposal – template

Student name and surname: Jake Morgan
Student number: 10560948
Name of the game: The Heart of the Soul
Target platform: Steam/DS/PSVita
Controls: Controlled Via UI (with mouse right now, also possibility for touch and directional pad/A button in future).

Tools:
(please provide a list with all the tools you are using – project management, coding, repo, etc)
Trello:
GitHub:
Unity, MonoDevelop, 3DS Max, Photoshop

Story:
(story of your game)
A lowly wizard, laughed out many a Wizarding conference, decides to get his revenge by building the ultimate monster! He has been travelling the world, slaying innocent Beasts and absorbing their souls to power his creation, when he stumbles upon our hero: a new Beast Tamer that has only just begun to realise their potential.

References/Inspiration:
(games or parts of a game that inspired you – add screenshots, descriptions, etc…try to capture the essence of that particular game and why is relevant to your game)

Most of these games will appear in my Moodboard research. Similar games with mechanics similar to mine include:

Stella Glow (3DS) (Placement and timing/range of attacks is the combat focus with stereotypical RPG roles such as fragile mages, tanks, archers, etc. on both player and enemy teams, and includes terrain height as a mechanic and very well designed 'levels' to fight in, linked with the story and world location of the level)

Agarest: Generations of War (PC) (Amazing combat system in which all characters can 'link' based on placement to enable special skills and combos, that the enemy also take advantage of, also features capturing of enemy monsters to use on your team and certain grid tiles that buff or debuff whomever lands on them)

Disgaea (PC/DS/PSP) (Icludes terrain height as a mechanic. Can interact with allies in combat to change flow of battle, i.e. throw teammates to either: land a teammate in a better position so on thier next turn they can make a strategic hit; with certain monsters, throw them as weapons, dealing big damage in an area but killing the monster. Also can throw enemies to same effect and uses a colour based buff/debuff tile system where you can apply a buff/debuff to an entire colour of tiles with a certain object)

Persona (PS2/3/4/PSP/PSVita) (Similar turn based style, not grid based, much more story focused and exploration focused, as such the battle system is quite basic. However, includes a

mechanic where you can attach 'Personas' to the main character to change his usable skills, elemental weaknesses/resistances, and stats.)

Fire Emblem (GBA/DS/3DS/Mobile) (Grid based combat system that allows player characters to 'support' each other with attacks or defend against enemy hits. Also includes a weapon triangle system similar to elemental systems. Uses this support system in unique ways such as pairing two allies to increase stats, or skills to shelter nearby units by pulling them into the same grid space (pairing up with them). Contains a simple attack range system in which most physical attacks can attack directly next to the user, ranged from 1 space away (but cannot attack next to themselves) or magic that attacks both next to and 1 space away, with unique weapons that deal less damage but increase/decrease range.)

Aim:
(describe what you are trying to achieve – please add everything that is relevant - mind maps, sketches, drawings flow diagrams, description of methods, screenshots)
Note: Paragraphs annexed with '******' are important parts that will definitely be included. The rest are ideas that might be implemented, or context for the full mechanic (what I would build if I had much longer to work on this).

My main inspiration is my love of turn based RPGs. I love the way they feel to play and the balance between skill (placement of players, use of skills at the right time, keeping teammates at healthy), and the customisation/personalisation paired with building up strength over time. This combined with high levels of difficulty give a great feeling of reward when completing a difficult mission and when combining this feeling of reward with permanent death mechanics (Fire Emblem style) and customising and building up the strength of a character over time, all helps create a strong bond with the characters.

I'm thinking of a turn based RPG battle system, in which the player relies on pets and weapon upgrades as opposed to personally leveling up. I feel this could make a really interesting mechanic for an RPG, where enemies you beat can be added to your team and the strength of the player is directly related to the strength of the pets.

This is a mindmap to demonstrate the mechanic and a user-story map on the same 'mindmap'.
https://bubbl.us/NDI5ODg5OS84Mzk3ODMwLzg2ZWNiZmQ0ZjQzZjJhMWY3NDk3N2I4YjRkMzJlMzM3-X?utm_source=shared-link&utm_medium=link&s=8397830

******* For the UI, I am thinking a basic menu system will be appropriate. I will create a system with a text box at the bottom for visual feedback, and a reactive menu system, floating to the left hand side of the screen (as left/right floating is custom in RPGs). This will disappear once an option has been selected, and will reappear when the player's turn comes back around.
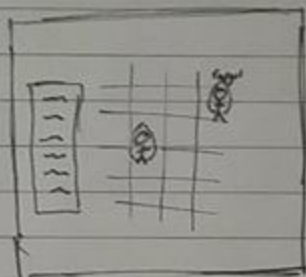
******* I will be using floating text above a characters head to show damage values to that character, important visual feedback that will make players more comfortable with the game (less text to read).

******* I will implement a "glow" around the selected character. If I have time, I might punctuate this with an idle animation to show that character is "active". I will use a blue glow for friendly characters, and a red glow for enemies. I will also apply the red glow to the selected enemy on the player's turn and will likely have the word "Target" float above their head. This, once again, should help the player feel more comfortable with the experience.

*******Whilst the Game Mechanic will be implemented on PC as a base, it would be nice to build for portable systems in the future, which should be taken in account when designing the UI. With Nintendo's N-indie initiative for the Switch, and the RPG-eccentric systems like the 3DS and PSVita having few releases to compete with, and, with the likelihood that a new portable Sony console is in development (due to the history of Playstation's new CEO), developing this for a portable system would be a solid idea. Once built for portables, the game can then be ported to PC with some effort, and can be marketed to the Steam Indie crowd too (with reputation gained from releasing on portable consoles first, the game is less likely to drown in the cascade of Steam Indies). This is all assuming I take the idea further, which I likely will. But this goal will be prevalent in my UI design.

****** Each character will be using certain traits, but will be implemented separately. Therefore, as I am building each unique character (after creating the base scripts) I could create a testing 'master' script for each character that obtains a reference to everything the character needs. For this, I could use an Interface 'Super-class' to ensure that no basic scripts are missed when creating the player/enemy characters. This would also serve as a form of run-time testing, as if anything is missing, it will stop the code from executing and inform me of missing references in the 'master' script.

******* Using the interface to force certain methods on the characters could allow me to call the turn order scripts without needing to use SendMessage, even though they are calling methods with different implementations from multiple different scripts.   To actually do this, I would need to create the Interface in a custom Namespace, and use the Using keyword to allow the character scripts to create the is-a relationship with the interface.

Breakdown

Component : Description (example)

Beast Species : Determines base element and Stats (skelican)

Nickname : Adds individuality and makes friendship increase faster (kiki)

Level : Increasing level boosts stats and can unlock new skills (Lv.4)

Element : Increases/Decreases stats against certain other elements the level affects the magnitude of the change (Fire Lv.2)

Range : The amount of squares the Beast can move (Rng:6)

Weapon : Increases attack and enables skills that use that weapon type (Iron axe)

Armor : Each piece increases defence, and may affect other stats depending on the piece.

Scroll : Temporarily lets the Beast use the skill on the scroll. This May still be limited by weapon (Icicle)

Elemental Stone : Increases the level of the Beasts element if it matches, changes the Element of the beast if it does not match (Ruby increases fire to Lv 2)

Accessories : Can provide additional buffs, elements, or even skills such as allowing skills to be used without the required weapon, or reducing the health consumed by skills. (1. Ring, 2. Bangle)

Each pet would have control of a weapon which gives them additional skills and can boost their stats. The MC will have basic fighting abilities, but will take power from surrounding units. The MC will gain strength from every pet within a certain range, if a pet is killed, it will no longer boost the player in that fight. I might also include a perma-death system with the pets, as this is popular in the current Indie gaming scene and will help the player build an emotional connection with some of their pets and encourage them to keep the pets alive.

Pets might drop weapons or scrolls that can teach skills/enchant weapons on death to reduce the feeling of 'wasted time' when raising pets just for them to fall in combat. I could also add a system in which the player can retire the pets for a much better reward than what is received when the pet dies. This would complement the 'weak player' idea, in which the MC could receive a power boost or new battle skill for raising a pet to the maximum level.

****** The battle system will likely be grid based, with a large battle area broken down into tiles. The player will control their pets and the MC, turn order will be determined on the pet/MC's speed/agility stat and a hidden 'delay' variable attached to the skills themselves (more powerful moves, somewhat slower turns). The pet/MC will be able to move a set amount of tiles, likely based on an A* algorithm for the opponent (if I do not do this 2-player). Moving will also have a hidden 'delay' variable attached to slow turn order.

Each pet will have an element: this will be determined by their species; mammals = earth, birds = air, lizards = fire/water, etc. There will be exceptions, each element will be prescribed on a pet by pet basis. The player will have the ability to change the element of the pet's attack and defence (separate to each other). Attack can be changed temporarily with certain skills and weapons, or buff's applied to self or by other pets to this pet. Defence can be changed with accessories or temporary buffs (in battle).

There could also be an attunement feature that permanently changes the Att/Def element of a pet using certain items (maybe obtained as rare battle drops, or from retiring team pets). Skills can have their own unique elements, these can be more powerful if the skill matches the element of the pet. Elements could be stackable, using buffs or permanent reinforcements can amplify the damage that pet deals with certain elements, but perhaps to balance it could reduce the power of other elements.

****** An idea that could play interestingly could be that using skills requires health as opposed to some kind of stamina/mana bar. This would make battles much more tactical and restrict the use of high-power moves and would encourage players to frequently swap the placement of pets, play to elemental strengths and keep super powerful characters behind to be able to wipe out weakened enemies, while remaining safe and able to use the move. With this mechanic, I could also create a 'Kamikaze' ability for all pets. There could be a hidden friendship stat that monitors this, but the damage of an attack that uses the remainder of a pet's health bar (killing it in the process) deals much more damage, based on proximity to the player, and dependent on the friendliness stat.

Naturally, I will not be able to implement all of these features into this single demo, but this is the overall scope of my project. I will be narrowing this down to manageable chunks, and will see just how much I can make in the given time.