

Booger Picker Home	3
Sprint planning	4
Daily Stand-Ups	5
Sprint 1	7
1. May 3	8
2. May 9, 2023	9
3. May 11	11
4. May 13, 2023	13
Sprint 1 Retrospective May 15, 2023	15
Sprint 2	17
1. May 18th, 2023	18
2. May 22nd, 2023	20
3. May 24th, 2023	22
Sprint 2 Retrospective May 26, 2023	24
Game design	26
User Interface (UI)	27
Level Design	28
Level Descriptions	30
Level Blockouts	32
Ice Door Blockout	34
Lava Pit Obstacle	35
Wind Leap Obstacle	36
Spikes on the bottom of the pit, don't kill the player	37
Water Gazers Obstacle	39
Burning Path	41
Puzzle Design	42
Puzzle Implementations	44
Windmill Puzzle	46
Windmill Door Connection Issue	47
Level Layout Concepts	48
Refactor Level Blockouts Scripts	49
The shop interior	51
Library	52
Gameplay	53
Player Agents	55
Player Agent Hurt	56
Player Agent Death	57
Spell Chooser	58
Players money	59
Player inventory	60
The player has a skin	61
Non-player Agents	62
Artificial Intelligence	63
Enemy Healthbar	64
Enemy Agent	65

Enemy Agent Implementation in Action Block Level.....	66
Wind-gifted Ranged Enemy Agent.....	68
Earth-gifted Enemy Agent.....	69
Librarian.....	70
Giving the NPCs skins.....	71
The NPCs react to the player.....	72
Spells /Magic.....	73
Water-based Abilities.....	75
Whip Spell.....	76
Earth-based Abilities.....	77
Earth Fist Spell.....	78
Fire-based Abilities.....	79
Fireball Spell.....	80
Air-Based Abilities.....	81
Dashing Spell.....	82
Levitate.....	83
Tornado Spell.....	84
Spells combinations (Combos) for damage amplification.....	85
Mana Cost System.....	87
Artificial Intelligence (AI).....	88
Enemy AI.....	89
Enemy Patrol.....	90
Ranged Enemy Attack.....	91
Ranged Enemy Retreat.....	92
Assets.....	93
Visuals and Audio (Animations, SFX, VFX).....	98
Implement Sound Effects for Elemental Spells.....	99
A sound for buying an item.....	101
Bugs.....	102
Spell projectile has no movement.....	103
Spell Fireball explodes (Effect only) when shoteed.....	104
Bug Report: All Blockouts Solved with Fire Spell Instead of Appropriate Elemental Spells.....	105
Pulling Procedures for Perforce Version Control in Unreal Engine 5.1.....	106
Team.....	107
Team Members.....	108
Roles and Responsibilities.....	109
Contact Information.....	110
Donovan's Game Ideas for the spells town enemies bosses.....	111

Booger Picker Home

Welcome to the Booger Pickers Confluence Home Page!

Spaces help our team structure, organize, and share work, ensuring every team member has access to the information needed to excel in their work.

Best Practices for Booger Pickers:

1. **Organization and Structure:** Maintain a well-structured space with clear page hierarchy, labelling, and categorization. Use appropriate labels, tags, and page templates to enhance searchability and ease of navigation.
2. **Document Versioning:** Keep track of document versions to ensure everyone is working with the latest information. Clearly indicate the version number, creation date, and any major updates made to the document.
3. **Collaboration and Sharing:** Encourage active collaboration by allowing team members to contribute, comment, and provide feedback on Confluence pages. Use the mentioned feature to notify specific team members about important updates or requests for input.
4. **Regular Updates:** Foster transparency and keep everyone informed by regularly updating relevant Confluence pages, especially during sprint planning, progress updates, and retrospectives. This ensures that everyone is on the same page and aware of project status and changes.
5. **Standup Meetings:** Create a dedicated Confluence page for standup meetings, where team members can share their progress, plan their tasks, and identify any blockers. Use the three-question format to streamline communication and keep the discussion focused.
6. **Agile Workflows:** Utilize Confluence to support Agile workflows, such as creating Jira tickets, tracking sprints, and documenting user stories, tasks, and acceptance criteria. Ensure all team members align with the project's goals, priorities, and timelines.
7. **Knowledge Sharing:** Encourage sharing knowledge and best practices within the team by creating dedicated pages for tutorials, guides, and documentation. Use Confluence's formatting features, such as headings, lists, and tables, to present information in a clear and concise manner.
8. **Visual Assets and Media:** Leverage Confluence's capabilities to embed images, diagrams, videos, and other media to enhance the understanding of complex concepts or demonstrate visual elements of the project. This helps to convey ideas and instructions effectively.
9. **Feedback and Iteration:** Encourage team members to provide constructive feedback on Confluence pages, including suggestions for improvements, clarifications, or additional information. Regularly review and iterate on pages based on feedback to ensure accuracy and relevance.

With these best practices, we can create a collaborative and productive environment within the Booger Pickers team. Confluence is our central platform for documentation, knowledge sharing, and project management. Let's embrace these practices and work together to achieve our goals efficiently and effectively.

If you need more information, check out our Confluence best practices guide or explore our guide for ideas on how to set up your space overview.

Sprint planning

Daily Stand-Ups

Date:

Time:

Duration:

Attendees	Briefly summarize the work you completed the last workday
1.	
2.	
3.	
4.	
5.	
6.	

Attendees	Briefly describe what you plan to work on today.
1.	
1.	
1.	
1.	
1.	
1.	

Attendees	Are you facing any issues or blockers that are hindering your progress?
1.	
1.	
1.	
1.	
1.	
1.	

Attendees	Action Items:
1.	
1.	
1.	
1.	
1.	
1.	

Next Meeting:

Date: [Date of next meeting]

Time: [Time of next meeting]

Duration: [Duration of next meeting]

Sprint 1

1. May 3

Date: May 3 2023

Time:

Duration:

Attendees	Briefly summarize the work you completed the last workday
1. Sean Jamelarin	Looking over the tasks and see if everything seemed valid. Working on fixing on some user stories that don't really fit in the game, and fixing the confluence page on some notes. Progression is halt because the dependency of needing a base character to create an enemy.
2. Malik Carter	Researched the pause the pause menu and player HUD, and planning to make some progression on actually making the UI elements of it
3. Donovan Sellers	Doing research for aesthetics and things that might want to add to the game. Planning to check how difficult some tasks might be, and since he's not in his own home so he can't do as much work as he can.
4. Jairo Manon	Have been working on create a base spell class and working on the fireball spell
5. Miguel Martinez	Designing the possible level blackouts and puzzles the player must get trough by using his/her spells.
6.	

Next Meeting:

Date: [Date of next meeting]

Time: [Time of next meeting]

Duration: [Duration of next meeting]

2. May 9, 2023

Date: Tuesday, May 9, 2023

Time: 9:00 am

Duration: 1 hour

Attendees	Briefly summarize the work you completed the last workday
1. Miguel Martinez Olivares	Imported all the essential assets to begin working and designing the level, as well as the spells.
2. Malik Carter	Research on how to add the playerhud onto the player and how it should be done on the widget
3.	
4.	
5.	
6.	

Attendees	Briefly describe what you plan to work on today.
1. Miguel Martinez Olivares	Today I'll begin with a basic design of the testing level, and create a base Blockout to implement later puzzles
1. Malik Carter	Player Hud with the Health and Mana bar. I also created the equipped spell slots but until there are actual spell images it'll just be white boxes
1.	
1.	
1.	
1.	

Attendees	Are you facing any issues or blockers that are hindering your progress?
1. Miguel Martinez Olivares	No
1. Malik Carter	No rn I just need to keep working on it
1.	
1.	
1.	
1.	

Attendees	Action Items:
1. Miguel Martinez Olivares	<ul style="list-style-type: none">• Level Design• Base blockout porgramming
1. Malik Carter	Player Hud visibility and spell slot inputs
1.	
1.	
1.	
1.	

Next Meeting:

Date: [Thursday, May 11, 2023]

Time: [9:00 AM]

Duration: [1 hour]

3. May 11

Date: Thursday, May 11, 2023

Time: 9:00 am

Duration: 2 hours

Attendees	Briefly summarize the work you completed the last workday
1. Miguel Martinez Olivares	Last workday I've added the dash, as well as dashable obstacles.
2. Sean Jamelarin	Refactored Dash Input
3. Donovan Sellers	Small shop design shop NPC and a little Shop UI
4.	
5.	
6.	

Attendees	Briefly describe what you plan to work on today.
1. Miguel Martinez Olivares	Working on adding the effects component, and depending on what type of spell, react differently.
1. Sean Jamelarin	Adding ability to cast spells with Left Button Click
1. Donovan Sellers	Finishing the shop possibly getting potions done
1.	
1.	
1.	

Attendees	Are you facing any issues or blockers that are hindering your progress?
1. Miguel Martinez Olivares	N/A
1. Sean Jamelarin	Needed to retarget workspace, machine memory storage bottleneck
1. Donovan Sellers	Possibly in game money and possibly HP and Mana
1.	
1.	
1.	

Attendees	Action Items:
1. Miguel Martinez Olivares	<ul style="list-style-type: none">• Effects Component• Interface for receiving effect (IEffectHandler)• Wooden Blockout (Burns with fire)
1. Sean Jamelarin	<ul style="list-style-type: none">• Player Agent• Some type of spell spawning class• Input delegate for player spell-casting
1. Donovan Sellers	<ul style="list-style-type: none">• Shop keeper• Shop keeper background• Shop keeper UI
1.	
1.	
1.	

Next Meeting:

Date: [May 13, 2023]

Time: [12:00pm]

Duration: [1 hour]

4. May 13, 2023

Date: Saturday, May 13, 2023

Time: 4:00 pm

Duration: 1 Hour

Attendees	Briefly summarize the work you completed the last workday
1. Miguel Martinez Olivares	Managed to create the ice wall blockout as well as the geizers puzzle.
2. Adrian Edwards	Finished ai pathing towards player and random patrolling.
3. Jairo Manon	Finished fixing the fireball and burning effect. Also finished working on the earth fist spell as well as the logic for the other spells
4.	
5.	
6.	

Attendees	Briefly describe what you plan to work on today.
1. Miguel Martinez Olivares	Work on the Wind Mill puzzle and on the Lava Pit obstacle
1. Adrian Edwards	Enemy attacks melee/ranged and enemy damage(able to kill enemy)
1. Jairo Manon	Finish the levitate spell, earth spell, and tornado spell
1.	
1.	
1.	

Attendees	Are you facing any issues or blockers that are hindering your progress?
1. Miguel Martinez Olivares	Missing the other spells for manage to work with different blockouts
1. Adrian Edwards	AI differentiating from friend or foe. ai will chase each other currently
1. Jairo Manon	The water whip it's still a bit complicated to figure out but I'm planning to use tomorrow and see what I can do.
1.	
1.	
1.	

Attendees	Action Items:
1. Miguel Martinez Olivares	<ul style="list-style-type: none"> • Wind Mill Puzzle <ul style="list-style-type: none"> ◦ Wind Mill Reacts with air spell ◦ Electricity gets activated ◦ Electric doors • Lava Pit Obstacle <ul style="list-style-type: none"> ◦ Earth platform created on hit
1. Adrian Edwards	Be able to attack or hinder player
1.	
1.	
1.	
1.	

Next Meeting:

Date: [May 15, 2023]

Time: [Asynchronous]

Duration: [Asynchronous]

Sprint 1 Retrospective May 15, 2023

📋 Overview

Reflect back on what you and your team learned and what motivates the group to succeed by following the instructions for the 4Ls Retrospective Play.

Team	Booger Pickers
Team members	@Malik Carter @Sean Jamelarin @Jairo Manon @adrian Michael Edwards @Miguel Martinez Olivares @DRSellers
Date	May 16, 2023
Retrospective period	Sprint 1

💡 4Ls retrospective

Milestones & Owner	Loved	Longed for	Loathed	Learned
Completion on User Interface Malik Carter	<ul style="list-style-type: none"> Designing the playerHud blueprint Finding assets to display the spells, crosshair and etc. 	<ul style="list-style-type: none"> What visual studio is supposed to have while trying to make a c++ class Answers on how to do certain problems that is not in blueprints 	<ul style="list-style-type: none"> While working on blueprints it took a while to find out how to properly connect blueprint classes to c++ classes Accidentally made a c++ writable without checking out but I managed to get it to work 	<ul style="list-style-type: none"> Learned creating references and setting them from blueprint to c++ classes More knowledge on how much blueprints can do
Completion on elemental block-outs Miguel Martinez	<ul style="list-style-type: none"> Successfully implemented a variety of elemental blockouts, showcasing the use of different spells to overcome obstacles. Blockouts provided engaging and challenging gameplay elements. Effective integration of elemental mechanics and puzzle-solving aspects. 	<ul style="list-style-type: none"> Needed the availability of all the different spells to test and experience each blockout fully. Insufficient time was allocated for playtesting and gathering feedback on the blockouts. Limited assets for the lava and windmill, hindering the visual quality and immersion of those elements. 	<ul style="list-style-type: none"> The Lava pit obstacle could use better ways to tell the player where to shoot to make platforms. I don't like the Wind Mill has no way to represent it is connected to the door. The Dash pit obstacle needs the spikes to deal damage to the player. 	<ul style="list-style-type: none"> Learned how to design and implement interactions between different spells and environmental elements. Gained knowledge on treating different damage types and their impact on gameplay. Improved understanding of level design principles and how they relate to gameplay mechanics.
Completion on elemental Spells Jairo Manon	<ul style="list-style-type: none"> The List of the different spells to implement The practice I got with my vector math and physics The problem solving I needed to do to fix some of the spells 	<ul style="list-style-type: none"> Implement the water whip Enough time to focus on this project and my tasks Every model and particle effects already in the project before I started 	<ul style="list-style-type: none"> The design of the solution I came up with for the spells The implementation of the Tornado and Earth fist 	<ul style="list-style-type: none"> How complicated it is to come with a design a solution to a problem Some cool vector math properties Different functions I could've used to launch a character to the direction I want. How to use the Niagara System
Completion on Player Character and Ranged Enemy Agent Sean Jamelarin	<ul style="list-style-type: none"> Successfully implementing spell casting on player character Successfully implementing animations into spell casting 	<ul style="list-style-type: none"> More time to implement casting of all spells Fine tuning of Ranged Enemy More varied Casting animations 	<ul style="list-style-type: none"> Ended up working on Player Character far more than anticipated Many Dependencies to be cleared before working on Ranged Enemy 	<ul style="list-style-type: none"> How to use UE5's new input system How to retarget animations to UE5 How to configure Navmesh generation settings

⚡ Action plan

Action	Owner	Due date	Action items
Finish the blockouts with the implementation of different damage types and spells.	@Miguel Martinez Olivares	May 23, 2023	<ol style="list-style-type: none"> Complete the implementation of the remaining spells to test and experience each blockout fully. Allocate additional time for playtesting and gathering feedback to iterate and refine the blockouts based on player input.

			<ul style="list-style-type: none"> 3. Improve communication and coordination to ensure the availability of required assets for the lava and windmill blockouts. 4. Explore asset acquisition options, such as outsourcing or asset marketplaces, to obtain high-quality assets for the lava and windmill blockouts
Enhance and finish the spell implementation with actual special effects, and visual and audio feedback.	@Jairo Manon	May 23, 2023	<ul style="list-style-type: none"> 1. Make dedicated time in the morning to work on my tasks 2. Fix the problem with collision of the spells to the owner 3. Fix the Tornado logic 4. Add damage type effects to each spell except the fire 5. Make the fireball more explosive 6. Add Sound to spells
	@Malik Carter	May 23, 2023	
Extend current Player Character spell casting to have player-driven choice over multiple spells, Ranged Enemy Agents that cast a different spell each, attach player condition to player HUD.	@Sean Jamelarin	May 23, 2023	<ul style="list-style-type: none"> 1. Refactor Spell Casting to contain information about multiple spell classes. 2. Refactor Spells to house AnimSequence information to their respective animation. 3. Attach Health Bar functionality to Player Character Health Component. 4. Create a Mana Component and attach to the player. 5. Refactor Spells to house Mana cost variable. 6. Attach Mana Bar functionality to Player Character Mana Component 7. Create a ranged enemy for every small school with a ranged magic

Sprint 2

Goal Description:

In Sprint 2, our focus is to further enhance the gameplay experience by implementing advanced features, refining existing mechanics, and introducing fun and innovative elements. We aim to expand the level design, improve spell functionality, balance the game mechanics, and incorporate additional creative mechanics to surprise and entertain players. The goal is to create a more polished and engaging prototype that showcases the game's core elements while providing a balanced and enjoyable experience.

Key Objectives:

1. Level Expansion:

- Design and implement additional puzzles, obstacles, and encounters to expand the gameplay variety and challenge.
- Integrate new blockouts and obstacles that require the use of different elemental spells to progress.

2. Spell Refinement:

- Enhance existing spells' functionality and visual effects, ensuring they are satisfying and visually impressive.
- Implement secondary effects and interactions for spells to add depth and strategic choices for the player.
- Refine and balance existing spells, including the air dash, to ensure a satisfying and fair playing experience.

3. Visual and Audio Enhancements:

- Improve the overall visual quality of the game by updating existing assets and incorporating new high-quality models and effects.
- Integrate atmospheric sound effects and music to enhance immersion and create a captivating audio experience.

4. Innovative and Fantastic Mechanics:

- Introduce three new fantasy-related mechanics, such as magical portal gates, that bring immersive experience and unexpected elements into the gameplay.
- These mechanics can include unique spell effects, unconventional enemy behaviour, or interactive elements that create interesting situations.

Milestone Deliverables:

1. Expanded Level Design:

- Additional puzzles, obstacles, and encounters are integrated into the game level.
- Blockouts and obstacles that require the use of different elemental spells for progression.

2. Refined Spells:

- Enhanced spell functionality, including secondary effects and strategic interactions.
- Visually impressive and satisfying spell effects.

3. Visual and Audio Polish:

- Updated and improved visual assets, including models, textures, and particle effects.
- Atmospheric sound effects and music enhance the immersion and ambiance of the game.

4. Player Character Advancements:

- Character upgrades and unlockable abilities are implemented, providing a sense of progression and customization.
- Advanced movement mechanics and responsive animations for a polished player experience.

We aim to create a more refined and engaging prototype in Sprint 2 by focusing on these key objectives and delivering the milestone deliverables. This will help us gather valuable feedback, iterate on the gameplay mechanics, and further solidify the game's foundation for future development.

1. May 18th, 2023

Date: Thursday, May 18th, 2023

Time: 9:00 am

Duration: 2 hours

Attendees	Briefly summarize the work you completed the last workday
1. Miguel Martinez Olivares	Added ice effect when entity gets damaged. And finished the planning for sprint 2
2. Adrian Edwards	completed switching ai from being child of player to being a child of character base class.
3. Sean Jamelarin	Repaperted the BaseProjectile to BaseSpell instead of Actor class, Added Mana Component to Player Agent
4. Malik Carter	Assets searching for the spells
5. Jairo Manon	Implemented the Fireball enhance
6.	

Attendees	Briefly describe what you plan to work on today.
1. Miguel Martinez Olivares	I'll work on creating a visual link from the WindMill to the electric door.
1. Adrian Edwards	I will work on using animation blueprint and state machines to create a melee animation then also create the melee damage
1. Sean Jamelarin	I will work on implementing a Mana Regeneration mechanic for the Player Agent.
1. Malik Carter	How to properly use the mouse wheel to select which spell to equip
1. Jairo Manon	Try getting started on the earth enhancement
1.	

Attendees	Are you facing any issues or blockers that are hindering your progress?
1. Miguel Martinez Olivares	Ice effect dose not apply movement effect, just attaches to character. Trying to find the solution.
1. Adrian Edwards	I dont know exactly how to do this stuff but i am willing and eager to learn. main point being state machines but the docs will be very useful.
1. Sean Jamelarin	Currently, I must tether a means of communication between the Spell Casting and Mana Component when spells are being used.
1. Malik Carter	Yes I'm currently struggling trying to find a good video or post about what i'm looking for plus I was feeling sick
1. Jairo Manon	Fireball had some bugs because of the change in inheritance
1.	

Attendees	Action Items:
1. Miguel Martinez Olivares	<ul style="list-style-type: none">• Create visual link between WindMill and electric door• Fix Ice effect
1. Adrian Edwards	
1. Sean Jamelarin	<ul style="list-style-type: none">• Mana Component• Mana Cost System
1. Malik Carter	Spells on playerhud player controls
Jairo Manon	<ul style="list-style-type: none">• FireBall• BaseProjectile
1.	

Next Meeting:

Date: [May 20,, 2023]

Time: [1:00pm]

Duration: [1 hour]

2. May 22nd, 2023

Date: Thursday, May 22nd, 2023

Time: 1:00 pm

Duration: 2 hours

Attendees	Briefly summarize the work you completed the last workday
1. Sean Jamelarin	I have successfully implemented Mana Regeneration into the Mana Component, Mana Cost System is complete.
1. Adrian Edwards	Completed melee animation and trigger action when i range of player.
1. Miguel Martinez Olivares	I focused on implementing different elemental effects when an agent gets hit by a spell. I successfully completed the effects for some elements and made progress on others.
1. Donovan Sellers	I made the player inventory mostly, money and shop stuff.
1. Malik Carter	Assets searching for the spells
Jairo Manon	Finished the earth enhancement, and tornado Enhancement.

Attendees	Briefly describe what you plan to work on today.
1. Sean Jamelarin	Start working on the Spell Chooser User Story.
1. Adrian Edwards	blend animation and melee together and cause damage to player when hit.
1. Miguel Martinez Olivares	I plan to continue working on the elemental effects for the remaining elements. Specifically, I will finalize the effects that are required for certain combos to deal extra damage. Additionally, I aim to add damage functionality to the spikes, as they currently do not inflict any damage to the player.
1. Donovan Sellers	I wanna finish both making the interior of both the shop and the library and hopefully find an easy way to put the player in and out of both
1. Malik Carter	Created the enemy ai healthbar and name
Jairo Manon	Getting started on the Boulder Spell and apply spell mana consumption.

Attendees	Are you facing any issues or blockers that are hindering your progress?
1. Sean Jamelarin	Right now, Donovan is directly working on the Player Agent, so until he is finished, my work is limited to the casting actor, and the spells themselves.
1. Adrian Edwards	Dont know how to call melee animation to begin when ai triggers to dela damage against player.
1. Miguel Martinez Olivares	I encountered a minor issue with the timing and duration of the elemental effects. Adjusting the timing to synchronize with the spell impact is challenging but necessary for optimal visual feedback. I will address this issue during today's work session.
1. Donovan Sellers	Yes interior lighting has been a BIG issue but I think I found some stuff that'll work for me
1. Malik Carter	Not at the moment
Jairo Manon	The lack of prepared assets for sounds and meshes, and also having some trouble fixing a weird bug in my tornado logic where it will push anybody out of the area of effect if you try to walk in

Attendees	Action Items:
1. Sean Jamelarin	<ul style="list-style-type: none"> • Spell Chooser
1. Adrian Edwards	<ul style="list-style-type: none"> • Enemy AI
1. Miguel Martinez Olivares	<ul style="list-style-type: none"> • Fine-tune the timing and duration of the elemental effects to align with the spell impact. • Complete the remaining elemental effects required for combo damage amplification. • Implement damage functionality for the spikes to ensure they inflict appropriate damage to the player. • Test and iterate on the implemented features to ensure they function as intended and provide a satisfying gameplay experience.
Jairo Manon	<ul style="list-style-type: none"> • Earth Fist

	<ul style="list-style-type: none">• Tornado_Atk
1. Malik Carter	Enemy hud
-	-

Next Meeting:

Date: [May 24th, 2023]

Time: [1:00pm]

Duration: [1 hour]

3. May 24th, 2023

Date: Thursday, May 24th, 2023

Time: 8:30 pm

Duration: Asynchronous

Attendees	Briefly summarize the work you completed the last workday
1. Sean Jamelarin	EQS Research, final implementations for Ranged Enemy
1. Adrian Edwards	AI now has walking animation. ai can use melee animation but does not do it when in melee range.
1. Malik Carter	Asset searching again for the spells and messed with the playerhud again to fit others criteria
Jairo Manon	Finished with the Boulder class, added to the UI so it casts the boulder spell, added a mana consumption, and fixed a bug with the applied damage
1. Miguel Martinez Olivares	I focused on implementing the spell combo system. Specifically, I successfully implemented the interaction between Fire and Ice spells, which creates a vapour effect and deals extra damage. Additionally, I added sound effects to the spells, enhancing the overall audio experience.
1.	

Attendees	Briefly describe what you plan to work on today.
1. Sean Jamelarin	I will work on the Ranged Enemy retreat, as well as testing the flexibility of my code with other attacking spells.
1. Adrian Edwards	Finish enemy ai melee action meaning both executing animation and dealing damage.
1. Malik Carter	Finished the design of the playerHud and with the help of Jairo I was able to get everything I said I'll do done on the sprint
Jairo Manon	Fix the bug for the radial damage for the fire ball
1. Miguel Martinez Olivares	My plan is to work on implementing all the blockouts and AI into the Action Block testing level. This includes placing the various obstacles and puzzles in the level and ensuring they function as intended. Additionally, I will dedicate time to refactor the code for the Avatar Base Character to manage the effects component better.
1.	

Attendees	Are you facing any issues or blockers that are hindering your progress?
1. Sean Jamelarin	I need to thoroughly walk through the logic of the behavior tree as well as extend the logic of the Perception Update event for Ranged AI Controller.
1. Adrian Edwards	Melee animation not playing when enemy is in melee range.
1. Malik Carter	When I was running the project through a standalone game there was a bit of a resolution error but it's not that big of a deal
Jairo Manon	Projectile gravity doesn't get applied for the boulder for some reason. For now, I enabled physics to the actual projectile but I need to fix something with the collision
1. Miguel Martinez Olivares	Currently, I am not facing any major issues or blockers. However, I anticipate some challenges while implementing AI behaviors, as it may require fine-tuning and adjustments to achieve the desired interactions with the player and environment.
1.	

Attendees	Action Items:
1. Sean Jamelarin	<ul style="list-style-type: none"> • Ranged Enemy Retreat • Ranged Enemy
1. Adrian Edwards	Melee enemy
1. Malik Carter	More spell asset images displaying keybinds
Jairo Manon	<ul style="list-style-type: none"> • Boulder

	<ul style="list-style-type: none"> • Spell Cast • Tornado • Fireball
1. Miguel Martinez Olivares	<ul style="list-style-type: none"> • Implement all the blockouts and puzzles in the Action Block testing level. • Integrate AI behaviours and fine-tune their interactions with the player and environment. • Refactor the code for the Avatar Base Character to improve the management of the effects component. • Test and iterate on the implemented features to ensure they function correctly and provide the intended gameplay experience.
1.	

Sprint 2 Retrospective May 26, 2023

📋 Overview

Reflect back on what you and your team learned and what motivates the group to succeed by following the instructions for the 4Ls Retrospective Play.

Team	Booger Pickers
Team members	@Malik Carter @Sean Jamelarin @Jairo Manon @adrian Michael Edwards @Miguel Martinez Olivares @DRSellers
Date	May 26, 2023
Retrospective period	Sprint 2

💡 4Ls retrospective

Milestones & Owner	Loved	Longed for	Loathed	Learned
	•	•	•	•
Miguel Martinez Olivares Enhancing Elemental Gameplay and Level Design	<ul style="list-style-type: none"> Successful implementation of elemental effects, such as ice and earth, adding visual and gameplay variety to the game. Improved level blockouts, introducing engaging and challenging obstacles for players. Integration of sound effects, enhancing the overall audio experience and immersion. Implementation of spell combinations for damage amplification, providing strategic depth and rewarding player skill. 	<ul style="list-style-type: none"> Availability of more time for thorough playtesting and gathering feedback on the implemented features. Enhanced visual quality and immersion through additional assets for elements like the windmill and lava. Further balancing and refinement of gameplay mechanics, particularly in spell combinations, to ensure a fair and enjoyable experience. 	<ul style="list-style-type: none"> Challenges caused by limited availability of assets, impacting the visual quality and immersion of certain elements. Insufficient time for in-depth playtesting and addressing potential areas for improvement. Inconsistencies with blockouts that were solved using fire spells instead of the appropriate elemental spells. Issues with spikes on the bottom of pits not causing damage to the player. 	<ul style="list-style-type: none"> The importance of comprehensive playtesting and iteration for refining implemented features. Increased knowledge and experience in implementing different elemental effects and their impact on gameplay and level design. The significance of sound effects in creating an immersive game environment. Improved skills in troubleshooting and resolving gameplay issues, ensuring a smooth and coherent player experience. Awareness of the importance of asset availability and quality for enhancing the visual appeal and immersion of the game.
Adrian Edwards Enemy AI and Animations	<ul style="list-style-type: none"> Learning how to use AI in unreal engine was fun Implementing Animation to control how AI acts Using blueprints is a fun and interesting way to code 	<ul style="list-style-type: none"> I wish I had more knowledge with Unreal. It was a real struggle to learn how to do basic things and make something that looked presentable with limited knowledge. 	<ul style="list-style-type: none"> There are a lot of resources however using unreal engine 5.1 was a detriment with fewer resources that when used would be more on the lines of copying code them implementing something new. I disliked spending a lot of time researching a thing that ultimately takes a few minutes to do in the right way. 	<ul style="list-style-type: none"> Documentation on UE5 is very useful try not to overthink how things should be done. ask fellow classmates or instructors for ulterior ways of completing a task their advice is often very helpful
Malik Carter Further developing the User Interface	<ul style="list-style-type: none"> Successfully completing all my tasks 	<ul style="list-style-type: none"> Being able to help with AI since my tasks has a lot of dependencies 	<ul style="list-style-type: none"> Me overthinking on how to do the display of spell switching. It costed me so much time and I was even thinking to duplicate every spell icon to show that it's equipped 	<ul style="list-style-type: none"> I have a better understanding on how blueprints work How to access blueprint and widget classes in different ways
Sean Jamelarin Enemy AI and Player Character	<ul style="list-style-type: none"> Successfully creating a rudimentary Spellcasting system Successfully implementing first iteration of a ranged enemy 	<ul style="list-style-type: none"> More time to redesign basic enemy logic for more intelligent AI. More varied spells for a more rewarding player experience 	<ul style="list-style-type: none"> Not having the time to tear down and redesign current Ranged AI logic once major oversights on original logic were made known, and instead making workarounds. Constant breakdown of blueprint objects due to default values not being saved by perfcore. 	<ul style="list-style-type: none"> Occasional code freezes to ensure all features combine well in development environment, then testing for bugs in the build of the game. EQS for implementation of a more utility-driven AI. More comprehensive understanding of UE5's new input system.
Jairo Manon Spell System	<ul style="list-style-type: none"> Successfully implementing all my tasks 	<ul style="list-style-type: none"> Polish all the spells when we combined our work 		

- Happy with the amount of math that I needed to learn to apply some of my spells
- More

⚡ Action plan

Action	Owner	Due date	Action items
Decide on Project Continuation for Fun and Learning	@Miguel Martinez Olivares	TBD when we decide to continue with the game	<ol style="list-style-type: none"> 1. Hold a team meeting to discuss the interest and enthusiasm among team members for continuing the project purely for fun and learning purposes. 2. Assess the progress and satisfaction of the team with the current development phase, focusing on the enjoyment and knowledge gained during the process. 3. Evaluate the availability of resources, including team members' time and commitment, to ensure the feasibility of continuing the project without external pressures. 4. Review the project roadmap and identify any necessary adjustments or refinements based on the team's goals and interests.
Decide to use Project as base material for Final Project	@Sean Jamelarin	May 31st	<ol style="list-style-type: none"> 1. Discuss with team members and get sentiments for project continuation as the final project, as well as potential alternatives. 2. Once a decision is made, ideate and plan next steps to next milestone, as well as ideal state for next milestone.
Making the PlayerHud look better and more understandable for new players	@Malik Carter	No idea	<ol style="list-style-type: none"> 1. Meeting with team members and discuss our next move

Game design

User Interface (UI)

As a developer, I want a HUD to test game features that would require a visual representation of the player's condition (Current MP, HP, spell, etc.)

Description

Put things in here that would enhance UI that isn't based on our code. References, things to download, you name it.

Completion Criteria

- Have a HUD that displays the intangible elements of the player/mobs that we wish to show (Health, Mana, Friendliness towards player)
- HUD changes to match current condition of player (MP goes down when being used, HP responds to damage/healing, cooldowns of spells.)

Level Design

Overview

The Rizzard's level design will be critical to ensure an engaging and challenging gameplay experience. The game will feature linear levels with puzzles that involve spells, unlockable spells, and boss rooms. The prototype level will have linear simple progression unlocking spells as you will require them for the following puzzles.

Player User Stories

User Story 1: As a Player, I want to travel through a level where I need to use different spells to solve puzzles and think through how to use the spells to solve the path.

Competition Criteria

- Each part of the level has puzzles that require the use of different spells.
- The puzzles are logically designed and progress in difficulty.
- The puzzles are complicated but not to the level of despair.  BOOG-10: Level Traveling (Movement with spells) IN PROGRESS

User Story 2: As a Player, I want to be able to use spells to traverse otherwise unreachable parts of the level.

Competition Criteria:

- The player must have access to at least one spell that allows them to traverse these areas.
- The spell must be easy to use and intuitive.
- The player must be able to use the spell without any special equipment or items.
- The spell must not be overpowered or make the game too easy.



Developer User Stories

Story 1: "As a developer, I want to create a level design that is intuitive for players to navigate through and understand."

Completion Criteria:

- Each level has a clear path for players to follow
- Levels have landmarks and indicators that guide the player through the level
- The level design is intuitive and easy to understand for players



Story 2: "As a developer, I want to design puzzles that challenge players and require them to use different spells."

Completion Criteria:

- Each level has puzzles that require the use of different spells
- Puzzles are designed logically and progress in difficulty
- Puzzles are challenging but not frustrating for players



Level Descriptions

Prototype Level: Action Block Level

Description:

The Action Block Level is a prototype level designed to test and demonstrate the game's core mechanics and features. The level consists of action blocks showcasing combat, spell usage, puzzle-solving, and platforming.

Completion Criteria:

- The level demonstrates the game's core mechanics, including combat, spell usage, puzzle-solving, and platforming.
- The level has clear and intuitive tutorials that teach other developers or testers how to use the game's mechanics and features.
- The level provides a challenging but not frustrating gameplay experience.



Epic Milestone (Maybe for capstone)

Level 1: Hub Town

Description:

Hub Town is the starting level of the game, and it serves as a hub area where players can interact with NPCs, unlock new spells, and access different levels.

Completion Criteria:

- The level has interactive NPCs that provide information to players and unlock new spells.
- Players can access different levels from Hub Town

Level 2: Forest of Enchantment

Description:

The Forest of Enchantment is level with a lush forest environment, and it features puzzles that require the use of water and earth spells.

Completion Criteria:

- The level has a visually stunning forest environment
- The level has puzzles that require the use of water and earth spells
- The puzzles are logically designed and progress in difficulty

Level 3: Fire Temple

Description:

The Fire Temple is level with a fiery environment, and it features puzzles that require the use of fire spells. At the end of the level, the boss room has a challenging fire-based boss that requires strategy and skill to defeat.

Completion Criteria:

- The level has a visually stunning fiery environment
- The level has puzzles that require the use of fire spells

- The puzzles are logically designed and progress in difficulty

Level Blockouts

Overview

Level blockouts are rough layouts of the game levels that help the team visualize and iterate on the level design before adding art assets and fine-tuning gameplay mechanics.

User Stories

Developer Stories

Story 1: "As a developer, I want to create level blockouts that accurately reflect the final level design."

- **Who:** Developers
- **What:** Create level blockouts that accurately reflect the final level design
- **How:** By creating rough layouts of the levels that include core gameplay mechanics, puzzle elements, and boss encounters

Completion Criteria:

- Level blockouts accurately reflect the final level design
- Blockouts include core gameplay mechanics, puzzle elements, and boss encounters
- The blockouts are logical and intuitive to navigate through  BOOG-19: Level Blockouts Implementation DONE

Story 2: "As a developer, I want to iterate on the level blockouts based on playtesting and feedback."

- **Who:** Developers
- **What:** Iterate on the level blockouts based on playtesting and feedback
- **How:** By testing the blockouts with the team and making adjustments based on feedback

Completion Criteria:

- Level blockouts are tested with the team and adjusted based on feedback
- Blockouts are polished and refined based on playtesting results
- The blockouts are ready for art asset implementation  BOOG-20: Playtesting and feedback for level blockouts DONE

Level Blockouts

Elemental Blockouts

Description:

Elemental Blockouts are puzzle elements that require the use of spells of different elements (fire, earth, air, and water) to progress through the level. The blockouts include puzzles that involve moving platforms, activating switches, and unlocking doors.

Completion Criteria:

- The blockouts include Elemental puzzle elements that require the use of spells of different elements.
- The puzzles are intuitive and easy to understand for players.
- Players must use spells of different elements to progress through the level.

Developer Story 1: "As a developer, I want to create puzzles that require the use of different spells and are fun to solve."

Completion Criteria:

- Elemental Blockouts include puzzles that require the use of spells of different elements

- Elemental Blockouts include puzzles that require the use of spells or different elements.
- The puzzles are fun and engaging for players to solve.
- The puzzles are tested and adjusted based on playtesting results. BOOG-22: Elemental Blockouts Design IN PROGRESS

Action Block Level

Description:

The Action Block Level is a prototype level that includes all the game's core mechanics, such as combat, spell usage, puzzle elements, and boss encounters. The level is a linear adventure game that ends with a challenging boss battle.

Completion Criteria:

- The level includes all the game's core mechanics, such as combat, spell usage, puzzle elements, and boss encounters.
- The level is intuitive and easy to understand for players.
- The level ends with a challenging boss battle. <https://2305boogerpicker.atlassian.net/browse/BOOG-23> - Can't find link

User Stories

Player Story 1: "As a developer, I want to experience all the game's core mechanics in one level to get a feel for the game's overall design."

Completion Criteria:

- The Action Block Level includes all the game's core mechanics, such as combat, spell usage, puzzle elements, and boss encounters.
- The level is fun and engaging for players to play through
- The level is tested and adjusted based on playtesting results

Developer Story 1: "As a developer, I want to create a prototype level that showcases all the game's core mechanics to get feedback from the team."

Completion Criteria:

- The Action Block Level includes all the game's core mechanics, such as combat, spell usage, puzzle elements, and boss encounters.
- The level is a linear adventure game that is intuitive and easy to understand for players.
- The level is tested and adjusted based on playtesting results.

Ice Door Blockout

Description:

The Ice Door Blockout is a puzzle element requiring a fire spell. It is a wall of ice that players must melt with the fire spell to progress through the level.

Spells needed to get through:

- Fire Spell

Completion Criteria:

- The blockout includes the Ice Door puzzle element
- The puzzle is intuitive and easy to understand for players
- Players must use the fire spell to melt the ice and progress through the level  BOOG-21: Ice Door Blockout DONE

User Stories

Player Story 1: "As a player, I want to use my spells creatively to solve puzzles and progress through the level."

Completion Criteria:

- The Ice Door Blockout is a creative puzzle element that requires spell usage.
- Players must use the fire spell to melt the ice and progress through the level.
- The puzzle is challenging but not frustrating for players.

Developer Story 1: "As a developer, I want to create puzzle elements that are intuitive and easy to understand for players."

Completion Criteria:

- The Ice Door Blockout is an intuitive puzzle element requiring fire spell use.
- The puzzle is easy to understand and logical for players
- The puzzle is tested and adjusted based on playtesting results

Lava Pit Obstacle

Lava Pit Obstacle

 BOOG-78 

Overview

The Lava Pit Obstacle is a puzzle element that requires the player to use their earth elemental power to create steps to progress over pools of lava in the level. This page will outline our approach to designing and implementing this puzzle element in the game.

Design

The Lava Pit Obstacle is designed to offer a challenging and engaging puzzle element requiring the player to think creatively and use their earth elemental power in unique ways. The obstacle is a section of the level that is filled with pools of lava that the player must cross to progress. The player must use their earth elemental power to create temporary steps over the lava that allow safe passage.

Spells Needed to Get Through

- Earth Spell

User Stories

Player Story 1: "As a player, I want to use my spells creatively to solve puzzles and progress through the level."

Completion Criteria:

- The Lava Pit Obstacle requires creative use of the earth's elemental power to create steps over the lava.
- The puzzle is challenging but not frustrating for players.
- The puzzle is intuitive and easy to understand for players.

Developer Story 1: "As a developer, I want to create puzzle elements that are intuitive and easy to understand for players."

Completion Criteria:

- The Lava Pit Obstacle is an intuitive puzzle element requiring creative use of the earth's elemental power.
- The puzzle is easy to understand and logical for players.
- The puzzle is tested and adjusted based on playtesting results.

Completion Criteria

To ensure that the Lava Pit Obstacle meets our design principles and goals, we will use the following completion criteria:

- The Lava Pit Obstacle is included in the game and is a creative and engaging puzzle element.
- The puzzle requires the player to use their earth elemental power uniquely to overcome the obstacle.
- The puzzle is intuitive and easy to understand for players.
- The puzzle is playtested extensively to ensure it is challenging but not frustrating for players.

Wind Leap Obstacle

Wind Leap Obstacle

 BOOG-79  DONE

Overview

The Wind Leap Obstacle is a puzzle element that requires the player to use their air elemental power to jump and dash to progress through the level. This page will outline our approach to designing and implementing this puzzle element in the game.

Design

The Wind Leap Obstacle is designed to offer a challenging and engaging puzzle element requiring the player to think creatively and use their air elemental power in unique ways. The obstacle is a level section that requires the player to jump across significant gaps and dash through the air to reach higher platforms.

Spells Needed to Get Through

- Air Jump Spell
- Air Dash Spell

User Stories

Player Story 1: "As a player, I want to use my spells creatively to solve puzzles and progress through the level."

Completion Criteria:

- The Wind Leap Obstacle requires creative use of the air elemental power to jump and dash through the air to progress through the level.
- The puzzle is challenging but not frustrating for players.
- The puzzle is intuitive and easy to understand for players.

Developer Story 1: "As a developer, I want to create puzzle elements that are intuitive and easy to understand for players."

Completion Criteria:

- The Wind Leap Obstacle is an intuitive puzzle element requiring creative use of the air elemental power.
- The puzzle is easy to understand and logical for players.
- The puzzle is tested and adjusted based on playtesting results.

Completion Criteria

To ensure that the Wind Leap Obstacle meets our design principles and goals, we will use the following completion criteria:

- The Wind Leap Obstacle is included in the game and is a creative and engaging puzzle element.
- The puzzle requires players to use their air elemental power uniquely to progress through the obstacle.
- The puzzle is intuitive and easy to understand for players.
- The puzzle is playtested extensively to ensure it is challenging but not frustrating for players.

Spikes on the bottom of the pit, don't kill the player.

Spikes on the bottom of the pit don't kill the player

 BOOG-108 DONE

Description:

The bug reported states that the spikes located on the bottom of the pit in the game are not functioning as intended, failing to kill the player upon contact. This Confluence page addresses the issue, providing a detailed analysis of the bug, its impact on gameplay, and steps for resolution.

Bug Details:

- Bug Title: Spikes on Bottom of Pit Do Not Kill Player

Impact:

The bug has the following effects on gameplay:

1. Inconsistent Gameplay: The failure of the spikes to kill the player upon contact creates an inconsistency in the game's mechanics and expected behaviour.
2. Lack of Punishment: Without the threat of instant death from the spikes, the pit loses its intended level of danger and suspense, affecting player engagement.
3. Progression Obstacles: If the player can safely traverse the pit without repercussions, it may diminish the challenge and impact of overcoming this obstacle.

Steps to Reproduce:

1. Position the player character near the pit with spikes on the bottom.
2. Allow the player character to fall into the hole and come into contact with the points.
3. Observe that the player character does not die upon contact with the spikes and can continue to move or escape the pit unharmed.

Expected Result:

Upon contact with the spikes on the bottom of the pit, the player character should suffer an instant death or receive significant damage, resulting in a failure state and potential respawn or reset.

Current Result:

The player character does not die or take any damage upon touching the spikes in the pit. They can continue to move freely without any consequences.

Resolution Steps:

To resolve the issue, the following steps can be taken:

1. First, locate the relevant code responsible for handling player collisions with the spikes in the pit.
2. Review the collision detection logic to ensure it properly detects contact with the player character.
3. Check for any missing or incorrect collision response code that should result in the player character's death or damage upon contact with the spikes.
4. Adjust the code to trigger the appropriate actions, such as player death or damage, when a collision with the spikes occurs.
5. Test the changes thoroughly to verify that the spikes now function as intended, causing the player character to die or receive damage upon contact.

Completion Criteria:

The following criteria will determine the completion of this bug fix:

1. The spikes on the bottom of the pit are functioning correctly, resulting in player death or damage upon contact.

1. THE SPIKES ON THE BOTTOM OF THE PIT ARE FUNCTIONING CORRECTLY, RESULTING IN PLAYER DEATH OR DAMAGE UPON CONTACT.
2. The bug fix does not introduce any new issues or regressions in the gameplay or functionality of the game.
3. Playtesting and verification confirm that the spikes now pose a significant threat to the player, enhancing the overall challenge and immersion of the pit obstacle.

By addressing this bug and ensuring the spikes in the pit function as intended, we aim to restore the intended gameplay experience, challenge, and sense of danger for players navigating the pit.

Water Gazers Obstacle

Water Gazer Obstacle



Overview

The Water Gazer Obstacle is a puzzle element that requires players to use their frozen and fire elemental powers to build and melt ice stairs to progress through the level. This page will outline our approach to designing and implementing this puzzle element in the game.

Design

The Water Gazer Obstacle is designed to offer a challenging and engaging puzzle element requiring the player to think creatively and use their elemental powers in unique ways. The obstacle is a series of water gazers that shoot water upwards, creating a series of platforms that the player must climb to progress through the level.

Spells Needed to Get Through

- Frozen Spell
- Fire Spell

User Stories

Player Story 1: "As a player, I want to use my spells creatively to solve puzzles and progress through the level."

Completion Criteria:

- The Water Gazer Obstacle requires creative use of the frozen and fire elemental powers to build and melt ice stairs to progress through the level.
- The puzzle is challenging but not frustrating for players.
- The puzzle is intuitive and easy to understand for players.

Developer Story 1: "As a developer, I want to create puzzle elements that are intuitive and easy to understand for players."

Completion Criteria:

- The Water Gazer Obstacle is an intuitive puzzle element requiring creative use of the frozen and fire elemental powers.
- The puzzle is easy to understand and logical for players.
- The puzzle is tested and adjusted based on playtesting results.

Completion Criteria

To ensure that the Water Gazer Obstacle meets our design principles and goals, we will use the following completion criteria:

- The Water Gazer Obstacle is included in the game and is a creative and engaging puzzle element.
- The puzzle requires players to use their frozen and fire elemental powers uniquely to overcome the obstacle.
- The puzzle is intuitive and easy to understand for players.
- The puzzle is playtested extensively to ensure it is challenging but not frustrating for players.

Description

The Water Gazer Obstacle is a puzzle element that requires players to use their frozen and fire elemental powers to build and melt ice stairs to progress through the level. The obstacle comprises a series of water gazers that shoot water upwards, creating a series of platforms that the player must climb to progress through the level.

Completion Criteria

To complete the Water Gazer Obstacle, the player must:

- Use their frozen power to freeze the water at different levels to create ice stairs.
- Use their firepower to melt the ice stairs if they are not at the correct height.
- Climb the ice stairs and progress through the level.

Dependencies

The Water Gazer Obstacle requires the following elements to be completed before it can be implemented:

- Implementation of the frozen and fire spells.
- Design and implementation of the water gazer models and animations.
- Design and implementation of the level layout and progression.

Burning Path

Burning Path Blockout

 BOOG-82 DONE

Description: The Burning Path Blockout is a level obstacle requiring players to use their water or freezing spell. The obstacle consists of a fire-filled path that players must traverse to progress through the level. However, the fire is too hot to touch, and players must use their elemental power to extinguish the flames and create a safe path to proceed.

Spells Needed to Get Through: Water Spell or Freezing Spell

User Stories:

Player Story 1: "As a player, I want to use my spells creatively to solve puzzles and progress through the level."

Completion Criteria:

To ensure that the Burning Path Blockout meets our design principles and goals, we will use the following completion criteria:

- The Burning Path Blockout requires creative use of the water or freezing spell to create a safe path.
- The Burning Path Blockout is an intuitive puzzle element requiring the use of water or a freezing spell.
- The puzzle is easy to understand and logical for players.
- The puzzle is tested and adjusted based on playtesting results.
- The Burning Path Blockout is included in the game and is a creative and engaging puzzle element.

Puzzle Design

Overview

Puzzles are an essential part of our game design, requiring players to creatively use their spells and abilities to overcome obstacles and challenges. This page will outline our approach to designing puzzles for the game and the key considerations that go into creating engaging and challenging puzzles.

Design Principles

Our puzzle design approach is based on the following principles:

- **Intuitive Design:** Puzzles should be easy to understand and intuitive for players to solve without the need for extensive tutorials or explanations.
- **Variety and Creativity:** Puzzles should be varied and offer different ways for players to solve them, allowing for creative solutions that use different spells and abilities.
- **Gradual Difficulty:** Puzzles should gradually increase in difficulty, starting with simple puzzles that introduce players to new mechanics and concepts before building up to more complex and challenging puzzles.
- **Feedback and Iteration:** Puzzles should be playtested regularly to gather feedback and make necessary adjustments based on player feedback.

Puzzle Elements

Our puzzles will include a variety of different elements and mechanics, such as:

- **Movement and Platforming:** Puzzles that require players to navigate through the environment using spells to jump, glide, or teleport to different locations.
- **Object Manipulation:** Puzzles that require players to move or manipulate objects in the environment using spells to activate switches or open doors.
- **Combat and Enemies:** Puzzles that require players to defeat enemies or use spells to avoid obstacles and hazards.
- **Spells and Abilities:** Puzzles that require players to use different spells and abilities to solve the puzzle, including elemental spells such as fire, water, earth, and air.

Design Process

Our puzzle design process will follow these general steps:

1. **Brainstorming:** The design team will brainstorm puzzle ideas and concepts, considering the above principles and elements.
2. **Prototyping:** The team will create rough prototypes of the puzzle, testing out different ideas and mechanics to see what works and what doesn't.
3. **Refining and Iteration:** Based on player feedback and playtesting results, the team will refine and iterate on the puzzle, making necessary adjustments to the design and mechanics.
4. **Implementation:** Once the puzzle is refined and finalized, it will be implemented into the game and tested again to ensure it works as intended.

Completion Criteria

To ensure that our puzzles meet our design principles and goals, we will use the following completion criteria:

- Puzzles are intuitive and easy to understand for players
- Puzzles offer variety and creative solutions using different spells and abilities

- Puzzles gradually increase in difficulty, starting with simple puzzles and building up to more complex and challenging puzzles.
- Puzzles are playtested regularly and refined based on player feedback and testing results.

Dependencies

To successfully design puzzles, we will need to clearly understand the game's core mechanics, including combat, spell usage, and level design. We will also need to have access to the necessary development tools and software to create and implement the puzzles in the game.

Puzzle Implementations

Overview

Once puzzle designs are finalized, the team will begin the implementation process. This page will outline our approach to implementing puzzles in the game and the critical considerations for creating engaging and challenging puzzles.

Implementation Process

Our puzzle implementation process will follow these general steps:

1. **Blockout Design:** The puzzle will first be designed as a blockout, allowing us to test and refine the puzzle's mechanics before creating final art assets.
2. **Art Assets:** Once the puzzle's mechanics are finalized, final art assets will be created to give it a unique look and feel. (LEGENDARY MARK, do not focus right now on this)
3. **Programming:** The puzzle will be programmed using the game engine's scripting language, with different elements and mechanics added as necessary.
4. **Playtesting and Refinement:** The puzzle will be playtested extensively to gather feedback and make necessary refinements to the design and mechanics.

Puzzle Examples

Here are some examples of puzzles that we will implement in the game, with elemental powers needed and increasing difficulty over time:

Puzzle 1 - Windmill Puzzle

- Elemental powers needed: Air
- Difficulty: Easy

Description: This puzzle involves a windmill that the player needs to activate using the air spell. The player must navigate through the environment and use their air spell to activate the windmill's blades, which will open doors and progress through the level. The puzzle requires timing and accuracy to activate the windmill's blades correctly.

Puzzle 2 - TBD

- Elemental powers needed: TBD
- Difficulty: Medium

Description: This puzzle involves using spells to navigate through the environment, with platforming challenges and hazards that the player must avoid. The puzzle requires the player to use different elemental powers to solve the puzzle and progress.

Puzzle 3 - TBD

- Elemental powers needed: TBD
- Difficulty: Hard

Description: This puzzle involves combat and enemy encounters, requiring players to use their elemental powers to defeat enemies and navigate the environment. The puzzle includes boss encounters and other obstacles that require creative use of the player's spells and abilities to overcome.

Puzzle 4 - TBD

- Elemental powers needed: TBD
- Difficulty: Expert

Description: This puzzle combines different spells and abilities to solve complex challenges with a high degree of difficulty and multiple paths to victory. The puzzle requires the player to use their knowledge of the game's mechanics and spells to solve the puzzle and progress.

Completion Criteria

To ensure that our puzzles are engaging and challenging, we will use the following completion criteria:

- Puzzles are well-designed and require the player to use their spells and abilities creatively
- Puzzles gradually increase in difficulty, with a clear progression from easy to expert puzzles
- Puzzles are playtested extensively to ensure they are fair and enjoyable for players

Dependencies

To successfully implement puzzles, we must have access to the necessary development tools and software to create and implement the puzzles in the game. We will also need to clearly understand the game's core mechanics, including combat, spell usage, and level design, to ensure that the puzzles are well-integrated into the game's overall design.

Windmill Puzzle

Windmill Puzzle

 BOOG-81  DONE

Description: The Windmill Puzzle is a puzzle element that requires the player to use their air elemental power to activate a windmill's blades to progress through the level. This page will outline our approach to designing and implementing this puzzle element in the game.

Design: The Windmill Puzzle is designed to offer an accessible and engaging puzzle element requiring the player to use their air elemental power creatively. The puzzle is a windmill the player needs to activate using their air spell. The player must navigate the environment and use their air spell to activate the windmill's blades, which will open doors and progress through the level. The puzzle requires timing and accuracy to start the windmill's blades correctly.

Spells Needed to Get Through: Air Spell

User Stories:

Player Story 1: "As a player, I want to use my spells creatively to solve puzzles and progress through the level."

Completion Criteria:

To ensure that the Windmill Puzzle meets our design principles and goals, we will use the following completion criteria:

- The Windmill Puzzle is included in the game and is a creative and engaging puzzle element.
- The puzzle requires players to use their air elemental power uniquely to progress through it.
- The puzzle is intuitive and easy to understand for players.
- The puzzle is playtested extensively to ensure it is challenging but not frustrating for players.

Windmill Door Connection Issue

Windmill Door Connection Issue

BOOG-104 DONE

Description:

The Windmill Door Connection Issue is inconsistent in the visual representation of the windmill's connection to the door. Currently, the windmill does not effectively indicate its linkage to the door, leading to confusion for players. This Confluence page outlines the issue and proposes a solution to address it.

Player Story:

As a player exploring the game world, I expect visual cues to guide me toward interactive elements and their connections. In the case of the windmill and door, I anticipate a clear visual representation of their relationship to understand how they are linked.

Expected Result:

When observing the windmill, I anticipate visual indicators, such as visible gears or rotating mechanisms, that clearly demonstrate its connection to the door. This would provide a clear understanding of the windmill's purpose and its effect on the door's functionality.

Current Result:

The windmill lacks visual cues that explicitly indicate its connection to the door. This absence makes it challenging for players to grasp the purpose and mechanics of the windmill in relation to the door.

To address this issue, we propose the following solution:

1. Enhance the Windmill Model:

- Modify the windmill model to include visible gears, cogs, or other mechanical components.
- Ensure that these additions visually connect the windmill to the door, representing their functional linkage.

2. Animation and Effects:

- Implement an animation that synchronizes the windmill's rotation with the door's opening or closing.
- Apply visual effects, such as magical energy or particle effects, to emphasize the connection between the windmill and the door.

By implementing these changes, we aim to provide players with a clear and intuitive understanding of the windmill's relationship to the door. This visual enhancement will enhance the overall immersion and gameplay experience.

Completion Criteria:

The following criteria will determine the completion of this issue:

1. The windmill model is updated to include visible gears or mechanical components that establish a visual connection with the door.
2. An animation is implemented that synchronizes the windmill's rotation with the door's opening or closing.
3. Visual effects, such as magical energy or particle effects, are applied to reinforce the connection between the windmill and the door.
4. Playtesting and feedback indicate that players now clearly understand the windmill's purpose and impact on the door's functionality.

By resolving the Windmill Door Connection Issue, we aim to enhance the clarity and intuitiveness of the gameplay experience for players.

Level Layout Concepts

Refactor Level Blockouts Scripts

Refactor Level Blockouts Scripts

BOOG-106 DONE

Description:

Refactoring the level blockout scripts is an essential task to improve the codebase's readability, maintainability, and overall quality. This Confluence page outlines the importance of refactoring and the goals of the refactoring process and provides guidelines for efficiently refactoring the level blockout scripts.

Developer Story:

As a Developer, I want to avoid trying to understand difficult-to-read code and manage to edit it when needed.

As a Developer, I want to be able to quickly manage to use and edit the blackouts for different purposes on the prototype level.

Importance of Refactoring:

Refactoring the level blockout scripts offers several benefits:

1. Readability: Refactored code is easier to read, understand, and navigate, making it more maintainable for the development team.
2. Modularity: Refactoring promotes modular code structure, allowing for better organization and reusability of script components.
3. Performance: Optimizations made during refactoring can improve performance and efficiency, resulting in smoother gameplay.
4. Debugging and Troubleshooting: Clean and well-structured code facilitates easier bug identification, isolation, and resolution.
5. Collaboration: Refactoring enhances code collaboration among team members, fostering a more efficient and cohesive development process.

Goals of Refactoring:

The primary goals of refactoring the level blockout scripts are as follows:

1. Code Readability: Enhance the readability of the scripts by applying proper naming conventions, adding comments where necessary, and organizing the code into logical sections.
2. Code Structure: Improve the code structure by breaking down complex logic into smaller, manageable functions or methods.
3. Optimization: Identify opportunities for optimizing the performance of the scripts, reducing unnecessary computations or iterations.
4. Modularity: Promote modularity by extracting reusable code segments into separate functions or classes, enabling easier maintenance and updates.
5. Consistency: Ensure consistency in coding style, variable naming, and script architecture throughout the level blockout scripts.

Refactoring Guidelines:

To efficiently refactor the level blockout scripts, follow these guidelines:

1. Review the existing codebase and identify areas that require refactoring based on code complexity, performance issues, or maintainability concerns.
2. Break down complex scripts into smaller functions or methods with clear responsibilities.
3. Following industry-standard practices, apply consistent naming conventions for variables, functions, and classes.
4. Add comments and documentation where necessary to improve code understanding for future maintenance or collaboration.
5. Remove redundant code or unused variables to enhance script efficiency and reduce potential bugs.
6. Test the refactored scripts to ensure that the functionality remains intact and that performance improvements are achieved.
7. Collaborate with other team members to gather feedback and suggestions for further code improvements.

Completion Criteria:

The completion of the level blockout script refactoring will be determined by the following criteria:

1. The codebase is refactored to improve code readability and maintainability.
2. Scripts are organized into smaller, modular functions or classes with clear responsibilities.
3. Performance optimizations are implemented, resulting in improved script efficiency and gameplay performance.
4. Code consistency is achieved following naming conventions and coding standards throughout the level blockout scripts.
5. Playtesting and feedback indicate that the refactored scripts maintain the desired functionality without adding regressions or bugs.

By successfully refactoring the level blockout scripts, we aim to enhance the codebase's overall quality, performance, and maintainability, facilitating easier collaboration and development for the entire team.

The shop interior

Currently the shop is just on the side of a building and really should have its own place to have the player feel like it is more like a shop

Completion criteria

The player walks into a door and it puts them in a different location

the location looks like a shop

The player can leave the shop and be on the outside of the building they entered

Library

The place where the librarian will be at and the player can tell it is a library

Completion criteria

It is a library it and looks like it.

The player can enter and leave

The player can see the librarian

Gameplay

UHealthComponent

Overview

The `UHealthComponent` is a component that can be attached to any actor to provide it with health functionality. This includes taking damage, healing, and triggering death-related behaviour.

Properties

- `MaxHealth` : The maximum health value an actor can have. The default is 100.
- `Health` : The current health of the actor. It is initialized with `MaxHealth`.

Methods

`UHealthComponent()`

The constructor of `UHealthComponent`. Initializes `MaxHealth` to 100 and `Health` to `MaxHealth`.

`BeginPlay()`

This method is called when the game starts. It binds `HandleDamageDel` to the `OnTakeAnyDamage` event of the owner actor.

`SetHealth(float NewHealth)`

Sets the `Health` to the given `NewHealth`.

`TickComponent(float DeltaTime, ELevelTick TickType, FActorComponentTickFunction* ThisTickFunction)`

I called every frame. For `UHealthComponent`, currently does not have any specific behaviour.

`HandleDamageDel(AActor* DamagedActor, float Damage, const UDamageType* DamageType, AController* InstigatedBy, AActor* DamageCauser)`

This method is used to handle the `OnTakeAnyDamage` event of the owner actor. It reduces the `Health` by `Damage`, checks if the actor has died (i.e., `Health <= 0`), and triggers the corresponding behaviours.

Event Handling

The `UHealthComponent` handles the `OnTakeAnyDamage` event by calling the `HandleDamageDel` method, which decreases the actor's health by the amount of damage received. If the health falls below or equals zero, it removes `HandleDamageDel` from the `OnTakeAnyDamage` event and sets the `Health` to 0.

Future Work

- Broadcasting health ratio.
- Broadcasting character death.
- Implementing damage and heal broadcasts.

- Adding support for different damage types and triggering corresponding effects.

Player Agents

User Story

As a player, I want to exist and be able to interact with the level, so I need a player-controlled entity that can carry out the intents of the player.

Description

This entity should have access to sight and hearing, and the ability to move, and the ability to manipulate the camera to see the level.

Dependencies

Avatar

Completion Criteria

The Player Agent should be able to move at the player's input.

The Player Agent should be able to change their direction with the player's mouse input.

 BOOG-52 DONE

Player Agent Hurt

User Story

As the player, I want to be able to know when I am damaged, so I will need to visually (and optionally hear) when I get hurt.

Description

Showing damage through both an animation and the health bar decrease would be ideal.

Dependencies

- Health Bar

Completion Criteria

- An injury animation can be seen when hurt.
- A visible change to the health bar can be seen.

 BOOG-68 DONE

Player Agent Death

User Story

As a Developer, I want to be able to punish a player's inadequate skill through the death of the player-controlled agent they are using.

Description

The Player Agents' death should prevent any further movement and world interaction input from the controlling player, as well as multiple options (through a HUD) to either restart the level, go back to the lobby hub, or to return to the main menu.

Dependencies

- Player Agent
- HUD

Completion Criteria

- The player should not be able to move in the level after dying, nor should they be able to use any spells or items.
- After the player dies, a viewport screen should be shown to the player giving his after-life options.



Spell Chooser

User Story

As a Player Agent, I want to be able to cast multiple types of spells throughout the level, so I will use the mouse wheel to choose what type of spell I want to cast.

Description

The player should be able to use the mouse wheel to cycle through a range of different spells to cast.

Completion Criteria

- The player has a visual cue on which spell is currently chosen
- The player can see currently chosen spell changing upon correct input
- The player can cast all spells available to them



Players money

As the player explores the world they will gather money they can use at a shop.

Completion criteria

Player knows how much money they have

Spending money removes their money portions

The item is set in their inventory

Player inventory

The player will need an inventory to hold their belongings that they have gathered through the game whether it be potions or other items

Completion criteria

Player can access their bag

The bag has usable items within it that are subtracted when used

the player can close the bag

The player has a skin

The Player is currently just an unreal maniquine. They should have their own skin that animations match with

Completion criteria

The player has a skin

The animations work with the skin

It looks accurate to the original mesh

Non-player Agents

User Story

As a developer, I want the player to have a more dynamic experience, so I will use non-player agents for the player to interact with through the player agent.

Dependencies

Avatar

Completion Criteria

A tangible Actor that exists in the level.



Artificial Intelligence

User Story

As a non-player agent, I want to be able to respond and interact with the player, so I can help or hinder the player.

Description

We require a behavior tree that compasses the actions that a non-player agent can take towards the player, environment, and its current condition.

Dependencies

- Avatar
- Non-player agent

Completion Criteria

- Non-player agents can react to a player's actions (perceive, interaction)
- Non-player agents can react to the environment (run away if there's fire, avoid falling off of cliff)
- Non-player agents can react to its current state (look for water if on fire, look for healing if hurt)

BOOG-75

Enemy Healthbar

User Story

As a player, I want to have an accurate gauge of what an enemy mob's current health is, in order to reduce the number of unknown variables in a combat situation.

Description

A small health-bar hovering directly above a hostile non-boss enemy's head for the player to see.

Dependencies

- Enemy Agent

Completion Criteria

- An indication of an Enemy's Health Bar above the head.
- Must be able to show the health bars of multiple enemies.
- Must be accurate and up-to-date with variable indicating the health of enemy agent.

Enemy Agent

User Story

As a Developer, I want a dynamic challenge to the player, so I will create an agent that will attempt to hinder the player's progression through the game.

Description

This agent should have the tools to recognize, approach, and antagonize the player, either through melee or magic attacks.

Dependencies

- Non-player Agent
- Magic Spell of any type
- Artificial Intelligence

Completion Criteria

- Be able to recognize the player as a hostile entity
- Be able to attack or hinder the player in some way
- Be able to directly or indirectly cause a player's lose condition to be triggered.



Enemy Agent Implementation in Action Block Level

Enemy Agent Implementation in Action Block Level



Description:

Implementing enemy agents in the Action Block Level is a significant step in enhancing the gameplay experience and providing engaging combat encounters for players. This Confluence page outlines the objectives, considerations, and efforts to effectively implement enemy agents in the Action Block Level.

Player User Story:

As a player, I want to encounter challenging and diverse enemy agents in the Action Block Level to engage in exciting combat encounters and test my skills.

Developer User Story:

As a developer, I want to effectively implement enemy agents in the Action Block Level to provide engaging combat encounters and ensure a balanced, fun gameplay experience.

Objectives:

The primary objectives of implementing enemy agents at the Action Block Level are as follows:

1. Combat Encounters: Create challenging and dynamic combat encounters requiring players to utilize their spells and abilities effectively.
(Managing to test if different mechanics together are fun)
2. Enemy AI Behavior: Develop intelligent and responsive enemy AI behaviour, including movement, attacks, and defensive actions.
3. Enemy Variety: Introduce various enemy types, each with unique characteristics, abilities, and attack patterns.
4. Balance and Difficulty: Ensure the enemies' difficulty and encounters are appropriately balanced to provide a satisfying and enjoyable gameplay experience.
5. Integration with Level Design: Seamlessly integrate enemy agents into the Action Block Level, considering the level layout, obstacles, and player progression.

Considerations:

When implementing enemy agents at the Action Block Level, it is essential to consider the following aspects:

1. Enemy Types: Determine the types of enemies to include, such as melee-based, ranged, or spellcasting enemies, each requiring distinct AI behaviours and attack patterns.
2. AI Behavior: Define the enemy AI behaviour, including movement, decision-making, and response to player actions, ensuring challenging and engaging combat encounters.
3. Attack Patterns: Design varied attack patterns for different enemy types, considering factors such as range, damage, timing, and telegraphing to provide tactical gameplay opportunities for players.
4. Health and Damage: Establish enemy health and damage values to achieve the desired difficulty level and ensure fair and balanced combat encounters.
5. Enemy Spawning: Determine the enemy spawning mechanics, including spawn locations, frequency, and wave progression, to create dynamic and immersive combat scenarios.

Implementation Steps:

The implementation of enemy agents in the Action Block Level involves the following steps:

1. Enemy Character Design: Create enemy character assets, including 3D models, animations, and visual effects, reflecting their unique traits and abilities.

2. AI Behavior Development: Implement enemy AI behaviour using a combination of behaviour trees, state machines, or other AI techniques, focusing on movement, attack, and defence strategies.
3. Attack and Damage Systems: Integrate enemy attack systems, including attack animations, hit detection, and damage calculations, ensuring accurate and satisfying combat interactions.
4. Enemy Spawning and Wave Management: Implement enemy spawning logic, considering spawn points, wave progression, and enemy group compositions.
5. Balancing and Playtesting: Fine-tune enemy parameters, such as health, damage, and AI behavior, based on playtesting and feedback to achieve desired difficulty and engagement levels.

Completion Criteria:

The following criteria will determine the completion of the enemy agent implementation at the Action Block Level:

1. The Action Block Level includes a variety of enemy agents, each with unique characteristics, abilities, and attack patterns.
2. Enemy AI behaviour is responsive and intelligent and provides challenging combat encounters for players.
3. Enemy attacks, hit detection, and damage calculations are accurately implemented, ensuring satisfying combat interactions.
4. Enemy spawning mechanics and wave management create dynamic and immersive combat scenarios.
5. Playtesting and feedback indicate that the enemy agent implementation enhances the overall gameplay experience at the Action Block Level.

By effectively implementing enemy agents in the Action Block Level, we aim to provide players with thrilling combat encounters, strategic gameplay opportunities, and an immersive gameplay experience throughout the level.

Wind-gifted Ranged Enemy Agent

User Story

As this type of Enemy Agent, I want to use my specialty to hurt the player, so I will patrol the level for the player, and harm them with the Tornado spell.

Description

This Enemy Agent will look for the player, and once found, will try to kill the player with the tornado spell, they do not retreat, since their magic forces them to stay still.

Dependencies

- Tornado Spell
- Enemy Patrol

Completion Criteria

- Can be seen patrolling for the player
- Can attack the player upon finding player

Earth-gifted Enemy Agent

User Story

As this Enemy Agent, I want to use my specialty to hurt the player, so I will patrol for the player and attack them using the Earth Fist spell.

Description

This enemy agent will look for the player, and once found, will chase after and attack them through a magic-enhanced melee attack.

Dependencies

- Melee Enemy
- Earth Fist
- Enemy Patrol

Completion Criteria

- Can be seen patrolling for the player
- Can hurt the player with Earth Fist spell
- Can chase after the player



Librarian

The Librarian will tell the player how certain spells work and some applications for the spell

Completion criteria

The spells are all listed

When you click a spell it tells you how it works

You can go back to the spell list and choose a new spell to view

The spell list is split up by sections

Giving the NPCs skins

When the player is in the city they would enjoy seeing different people within the city and not just one lifeless maniquine

Completion criteria

There is a set number of skins that the NPCs around town can have

The NPCs animations work with their skins

The NPCs react to the player

Rather than the NPCs being fine with the player shooting spells all willy nilly they will run away and be scared of the player. Possibly even scream and yell showing fear. On the other hand places like the shop and the librian will kick the player out

Completion criteria

NPCs run away from the players spells

Shop and Librian will kick player out of their environment

NPCs will yell

Spells /Magic

Effects Component

Make an effects component to manage the actors that can detect which spell is currently hitting them BOOG-88: Effects Component (Detection) DONE.

Task Details

1. Design the Spell Detection Interface:
 - Define the necessary methods and properties to enable actors to detect spells affecting them.
 - Consider the requirements for identifying spell names, types, durations, and other relevant information.
2. Implement Spell Detection Functionality:
 - Integrate the spell detection interface into the existing Effects Component.
 - Modify the component to track the spells affecting actors and provide the necessary information for detection.

Add different elements effects to affected actors

 BOOG-109 IN PROGRESS

Description:

This user story focuses on adding different elemental effects to affected actors in the game. It involves implementing the effects of fire, ice, earth, and air spells on enemies, providing players with unique gameplay mechanics and strategic options. This Confluence page outlines the user story, its goals, and considerations for implementation.

User Story:

As a player, I want to see different elemental effects applied to enemies when casting fire, ice, earth, and air spells. For example, I expect the fire effect to deal damage over time, the ice effect to slow or freeze enemies, the earth effect to stun or lock enemies in place, and the air effect to disarm enemies.

Goals:

The primary goals of this user story are as follows:

1. Elemental Variety: Add diverse elemental effects to affected actors, enhancing players' visual and gameplay experience.
2. Strategic Options: Provide players with strategic choices by leveraging different elemental effects to gain advantages in combat.
3. Immersive Gameplay: Create a more immersive and dynamic gameplay experience by simulating the impact of elemental spells on enemies.
4. Balancing: Ensure each elemental effect offers a balanced, meaningful gameplay impact, promoting variety and player choice.

Considerations:

When implementing the different elemental effects on affected actors, consider the following aspects:

1. Visual Representation: Develop distinct visual effects for each elemental effect, clearly communicating the type of effect applied to enemies.
2. Gameplay Mechanics: Design appropriate gameplay mechanics for each elemental effect, ensuring they align with their respective elements and provide strategic advantages for players.

3. Enemy Behavior: Determine how enemies react and adapt to different elemental effects, such as resistance, vulnerability, or altered attack patterns.
4. Balance and Fairness: Fine-tune the strength and duration of each elemental effect to ensure fair gameplay and avoid overpowered or underpowered outcomes.
5. Player Feedback: Implement visual and audio feedback to inform players when an enemy is affected by an elemental effect, reinforcing the impact of their spells.

Implementation Steps:

To implement different elemental effects on affected actors, follow these steps:

1. Define the visual effects for each elemental effect, such as fire particle systems, icy overlays, earthy cracks, and wind gusts.
2. Develop the underlying gameplay mechanics for each elemental effect, including damage-over-time for fire, slowing or freezing for ice, stunning or locking for the earth, and disarming for air.
3. Integrate the visual effects and gameplay mechanics into the affected actors, ensuring they are appropriately triggered when the corresponding elemental spells hit them.
4. Test the implemented elemental effects extensively to ensure they function as intended, provide strategic options, and maintain game balance.
5. Gather feedback from playtesting and iterate on the effects as necessary, making adjustments to improve visual clarity, gameplay impact, and player satisfaction.

Completion Criteria:

The following criteria will determine the completion of this user story:

1. When hit by corresponding spells, different elemental effects (fire, ice, earth, and air) are successfully applied to affected actors.
2. The fire effect deals damage over time, the ice effect slows or freezes enemies, the earth effect stuns or locks enemies, and the air effect disarms enemies.
3. Visual effects and audio feedback communicate the elemental effects applied to enemies, enhancing immersion and understanding for players.
4. Playtesting and feedback confirm that the elemental effects provide strategic options, balance, and an immersive gameplay experience.

By implementing different elemental effects on affected actors, we aim to enrich the gameplay experience, offer strategic choices to players, and enhance the immersion and variety of combat encounters within the game.

Water-based Abilities

References:

[▶ Katara vs. Pakku ▶ Full Scene | Avatar: The Last Airbender](#)

notes:

water enters player's hand allowing a mid-range whip attack that cleaves

Overview

The Rizzard will have access to Water like spells that cause a cleaving effect against enemies who are clutched together. This is a mid-range spells that would give the player the opportunity to attack multiple enemies at the same time at the cost of damage.

Player User Stories

User Story: As a player, I want an option to be able to hit multiple enemies with one spell to save some mana consumption and still do a decent amount of damage.

Competition Criteria

- Make Visual effects.
- When the spell collides with an enemy, the spell would go until the end of the attack animation, hitting anything on its path.

 BOOG-26 TO DO

Whip Spell

Overview

This spell will cause a cleaving effect to a resemblance of a water whip.

Player User Stories

User Story: As a spell, it needs to provide the player the option to cause a mana efficient spell to cut down multiple enemies.

Competition Criteria

- A Whip like visual effect and collision.
- The Whip would not disappear until the animation is finished.
- The Whip is able to apply damage to whatever it hits once.

 BOOG-29 TO DO

Earth-based Abilities

References:

(4) Wizard of Legend - Crit Drill Builds ft. my curse addiction - YouTube

Overview

The Rizzard will have access to an earth like spells that would allow for a close-range combat against the enemies. At the cost of more mana, earth spells would become a more damaging spell than others.

Player User Stories

User Story: As a Player, I want to be able to use earth like spells towards enemies who might be weak against that element and provide a more damaging option for situations when the enemy is a bit too close.

Competition Criteria

- Visual Effects
- The spell deteriorates after some time has passed.
- When spell collides with a hittable object, it would apply massive damage to whatever it hit.

 BOOG-25 TO DO

Earth Fist Spell

Overview

The Rizzard will punch the ground causing it's hands becoming hard as stone.

Player User Stories

User Story: As a spell, it needs to provide the player an option on when the enemy gets too close and outputting as much damage as possible to stay out of trouble.

Competition Criteria

- Visual Effects of the hands of the player becoming stone
- The spell disappears when enough time has passed.
- When Collided with an enemy, it causes blunt damage while also pushing them back.



Fire-based Abilities

reference:

(4) Skyrim Special Edition Builds – BEST Spells Location (3000+ DAMAGE! Secret Mage Build)! - YouTube

Overview

The Rizzard will have access to a fire like spells that would cause some kind of a fire/burning effect to the enemies. This is a long-range spell that would shoot some variant of a fire ball but with different properties between each fire spell.

Player User Stories

User Story: As a Player, I want to be able to use a variety of different fire spells to cause a fire/burning effect to my enemies.

Competition Criteria

- The fire spell has visual effects.
- The Projectile would continue until it hits a object or when the allotted time for it to hit something has been reached.
- When the spell collides with an enemy, it would cause damage, and make the enemy burn over a time of 2 seconds.  BOOG-24: Fire
[Spells' description](#) DONE
- FireType Damage Type class to handle fire effects

Fire Effect

 BOOG-90 IN PROGRESS

Overview:

The fire effect feature aims to provide players with an immersive and visually stunning experience when they cast fire spells in the game. The effect will dynamically attach to the targeted actor, creating the illusion of flames engulfing the character. This will enhance the overall gameplay experience and add a sense of danger and excitement.

User Stories

As a game developer, I want to create a realistic fire effect that can be attached to the actor receiving the fire spell to enhance the visual impact and immersion of the game.

As a Player, I want to feel the environment reacting to the fire spells I throw.

Fireball Spell

Overview

The Rizzard will concentrate their mana to the palm of their hands, firing a fireball projectile when released.

Player User Stories

User Story: As a spell, it needs to provide the player a base damaging spell so that it can protect itself.

Competition Criteria

- Visual Effects of a fireball
- The spell disappears when collided with an object or when enough time has passed.
- When Collided with an enemy, it causes a burning effect that stays damages it over a period of time.

 BOOG-31 TO DO

Air-Based Abilities

references:

 [Jett now has 2 Dashes \(Double Dash\)](#)

notes:

dash in a direction and levitate for a duration allowing mid-air spell casts

tornado attack ability

Overview

The Rizzard has access to air spells to be able to add more into their mobility, in terms of dodging or climbing up objects. It also provides with a tornado spell that can be used to gather up enemies.

Player User Stories

User Story: As a Player, I want to enhance my mobility for dodging different kinds of enemies and traversing around the world. But as well, I want to have a way to gather up enemies and attack them all at the same time.

Competition Criteria

- Create a Dashing Spell
- Create a levitating spell.
 - Be able to cast spells while levitating.
- Spell is destroyed after some time has passed
- The spell has the main Property of pulling enemies together when collided with an enemy, or an object.

 BOOG-27 [TO DO](#)

Dashing Spell

Overview

The Rizzard will have the option to dodge enemies through dashing with its air abilities.

Player User Stories

User Story: As a spell, it needs to provide the player some mobility to make the player escape dangerous scenarios when needed.

Competition Criteria

- Visual effects of the spell taking effect
- The spell has a limited range.
- Is Collision disabled until the end of the animation?



Levitate

Overview

The Rizzard will have the ability to slow down its fall while still having the ability to launch spells at the cost of more mana.

Player User Stories

User Story: As a Player, I want a way to traverse through the world on long distanced platforms, as well as having the option to cast spells while midair.

Competition Criteria

- Visual Effects of air under the player
- The spell has an effect of making spells that are being cast while airborne to cost 1.5 more than usual.
- Spell disappears when the caster has no mana left or has landed.

 BOOG-33 TO DO

Tornado Spell

Overview

The Rizzard will concentrate their mana to the palm of their hands, firing a disc like object that expands on collision of an object. This would damage as well as pulling enemies together towards its center.

Player User Stories

User Story: As a player, I want a spell that allows me to pull a group of enemies together, giving me a chance to defeat them all at once.

Competition Criteria

- Visual Effects of a Tornado
- The spell disappears when a certain amount of time elapsed; either on collision or not.
- The spell expands when it collides with an object, making it pull enemies together from a certain radius.

 BOOG-34 TO DO

Spells combinations (Combos) for damage amplification

Spells Combinations (Combos) for Damage Amplification - Story Task



Overview

This Confluence page details the "Spells Combinations (Combos) for Damage Amplification" story task in Jira. It outlines the objectives and requirements for implementing a combo system that allows players to combine spells for increased damage output. Below are two player stories and one developer story associated with this task.

Player Stories

As a player, I want to discover and utilize powerful spell combinations to amplify damage and defeat enemies more efficiently. By executing successful combos, I can strategically plan my moves and experience the satisfaction of dealing significant damage.

Acceptance Criteria:

- Clear visual and audio cues to indicate successful combo execution.
- The combo system is explained in the tutorial or game instructions.
- Variety of available spell combinations to encourage experimentation.
- Noticeable increase in damage output when executing successful combos.
- Combo effects and amplification are displayed in the user interface for feedback.

As a player, I want to strategically plan my spell combinations to maximize damage output and overcome challenging encounters. By understanding the effects and properties of each spell, I can create powerful combos that give me an advantage in battles.

Acceptance Criteria:

- A clear understanding of each spell's effects and properties facilitates combo planning.
- Accessible information about spell combinations and their potential damage amplification.
- Visual and audio feedback to indicate successful execution of planned combos.
- Balanced gameplay that rewards strategic combo usage without making it overpowered.
- In-game feedback or indicators to show the current active combo and remaining combo duration.

Developer Story

As a developer, I want to implement a robust combo system that enhances the player's combat experience and encourages strategic spell usage. This system should allow players to perform various spell combinations for damage amplification.

Acceptance Criteria:

- Design and implement a comprehensive combo system that allows for various spell combinations.
- Create a database or system to define and store valid spell combinations and their associated effects.
- Integrate the combo system with combat mechanics and damage calculation algorithms.
- Test and balance the combo system to ensure it provides a meaningful and rewarding gameplay experience.
- Document the combo system, including available spell combinations and their effects, for reference and future updates.

Conclusion

Implementing a spell combo system will provide players with an exciting and strategic element in the game. By combining spells, they can achieve higher damage output and overcome challenging encounters. This Confluence page references the "Spells Combinations (Combos) for Damage Amplification" story task, clearly understanding the player and developer's expectations. Let's collaborate and work together to bring this engaging feature to life!

Mana Cost System

User Story

As a developer, I want to discourage the player from reckless use of magic spells, so I will give all spells a mana cost to cast them, preventing the player from casting if they do not have sufficient mana.

Description

The player has a maximum amount of mana they can have at one time, using their reservoir when casting spells, and will regenerate mana when not casting spells after a brief period of time.

Dependencies

- Magic Spells
- Player Agent Spell Casting

Completion Criteria

- Player has a visual cue to how much mana they have
- Player's mana can be seen decreasing when using spells
- Player's mana can be seen refilling when not at max mana and not casting spells after a period of time
- Player cannot cast spells if mana is too depleted



Artificial Intelligence (AI)

User Story

As a non-player agent, I want to be able to respond and interact with the player, so I can help or hinder the player.

Description

We require a behavior tree that compasses the actions that a non-player agent can take towards the player, environment, and its current condition.

Dependencies

- Avatar
- Non-player agent

Completion Criteria

- Non-player agents can react to a player's actions (perceive, interaction)
- Non-player agents can react to the environment (run away if there's fire, avoid falling off of cliff)
- Non-player agents can react to its current state (look for water if on fire, look for healing if hurt)

BOOG-46

Enemy AI

This is a subsection of AI that encompasses behaviors that will hinder the player's progression through the game.

Enemy Patrol

User Story

As an enemy agent, I want to be able to make sure that the player is not currently an immediate threat, so I will move around my surroundings to check for the player's presence.

Description

The enemy should be able to look within a confined area for the player, and once spotting the player, either retreat or attack depending on the conditions of the encounter and type of enemy agent.

Dependencies

Enemy Agent

Completion Criteria

- Enemy can be seen moving in a defined area of patrol
- Enemy can perceive the player
- Enemy can switch to appropriate behavior once player is perceived



Ranged Enemy Attack

User Story

As a Ranged Enemy Agent, I want to hinder the player's progression by means of threatening a lose condition (AKA death), through damaging spells, encouraging either evasion or combat.

Description

The enemy agent should be able to challenge the player through a ranged combat ability.

Dependencies

- Enemy Agent

Completion Criteria

- Ranged Enemy Agent's attacks should be able to damage the player.
- Ranged Enemy Agent's attacks should be able to kill the player.  BOOG-85: Ranged Enemy Attack DONE

Ranged Enemy Retreat

User Story

As a ranged enemy agent, I want to preserve my own existence while attempting to successfully hinder the player's progress, so I will attempt to distance myself from the player while the player is too close to me.

Description

Ranged Enemy Agent will attempt to create a bare minimum amount of distance from the player after said bare minimum distance has been breached.

Dependencies

- Enemy Agent

Completion Criteria

- Player should be able to see ranged enemy agent moving away from player after getting close enough to enemy agent.
- Ranged Enemy Agent should be able to start attacking again once player exits fleeing radius. [BOOG-86: Ranged Enemy Retreat](#)

DONE

Assets

Overview:

The Assets page is where we will document all the assets required for the game. This includes particle effects, elemental assets, sound effects, characters, enemies, and environment models. We will reference other games to gather inspiration and ideas for our own assets.

Assets Needed:

BOOG-83 DONE

Particle Effects:

- Fire and smoke effects for fire spells
- Water and mist effects for water spells
- Earth and stone effects for Earth spells
- Wind and air effects for air spells
- Spell-casting effects for each element
- Spell impact effects for each element

Elemental Assets:

- Fire-themed props and objects
- Water-themed props and objects
- Earth-themed props and objects
- Air-themed props and objects

Sound Effects:

- Spell casting sounds for each element
- Spell impact sounds for each element
- Ambient sounds for different environments
- Enemy sounds and voice lines

Characters and Enemies:

- Middle Earth-inspired player character models
- Middle Earth-inspired enemy models

Environment Models:

- Nature models such as trees, bushes, and rocks
- Medieval-themed environment models such as buildings, bridges, and walls

References:

- Middle Earth games such as Shadow of Mordor and Shadow of War for character and enemy inspiration
- Other games such as Skyrim and The Witcher for the environment and asset inspiration

Completion Criteria:

- All necessary assets are included in the game
- All assets meet the required level of quality and polish
- All assets are tested and adjusted as necessary to ensure they fit with the game's design and mechanics

Dependencies:

- Completion of the Level Blockouts page to ensure all necessary environment models are included
- Completion of the Spell/Abilities page to ensure all necessary elemental assets and particle effects are included
- Completion of the AI page to ensure all necessary enemy models and sounds are included.

References:

(826) Complete MAGIC Overhaul For Skyrim 2023 - Best Magic Mods - YouTube

(826) Walking the WHOLE MAP in ONE GO! 7 Days! ASMR! RAW Video File in Zelda Breath of the Wild - YouTube

Possible Assets:

The screenshot shows a product page for 'Luos's Eight elements' on a digital marketplace. The top navigation bar includes 'CONTENT DETAIL', 'Home', 'Browse', 'Industries', 'Free', 'On Sale', 'Mega Sale', 'Submit Content', 'Help', and a search bar. The main image displays four large screenshots of a character performing various elemental effects: water, earth, fire, and air. Below these are smaller thumbnail previews of other assets. To the right, the product title 'Luos's Eight elements' is shown, along with the subtitle 'Luos - Visual Effects'. It has a 4.5-star rating from 17 reviews and 11 of 11 questions answered. A description states it's a combined version of 'Four Elements' and 'Four More Elements' packages. On the right side, there are buttons for 'Add To Project' and 'Write a Review', and sections for 'Supported Platforms' (Android, PC, Mac, iOS, Windows) and 'Supported Engine Versions' (4.16 - 4.27, 5.0 - 5.1).

Blink and Dash VFX

pelengami - Visual Effects

★★★★★ ✓ 7
7 reviews written | 11 of 11 questions answered

Blink and Dash (mesh-based) effects

Add To Project OR Write a Review

Supported Platforms:

Supported Engine Versions: 4.25 - 4.27, 5.0 - 5.1

Graveyard and Nature Set

NatureManufacture - Environments

★★★★★ ✓ 44 6 reviews written | 5 of 5 questions answered

This pack contains huge amount of modular PBR sculpted cemetery and environment nature assets that are high-quality models with LODs. Textures atlases are 2048x2048 up to 4096x4096. Pack contains 75 assets. Interiors are included now!

Add To Project OR Write a Review

Supported Platforms:

Supported Engine Versions: 4.15 - 4.27, 5.0 - 5.1

Four Pack - Male Dwarfs - Fantasy Dwarf Collection

Yarrawah Interactive - Characters

★★★★★ ✓ 1 1 review written | 3 of 3 questions answered

Add a collection of fantasy dwarves to your game in this combined mega pack!

Add To Project OR Write a Review

Supported Platforms:

Supported Engine Versions: 5.0 - 5.1



Civilian - Male Elf - Fantasy Elves Collection

Yarrawah Interactive - Characters
★☆☆☆☆ 1 review written | 1 of 1 question answered
Add a civilian fantasy elf to your game!

Add To Project Supported Platforms

DWCH PEGI 12



Goblins - Fantasy RPG

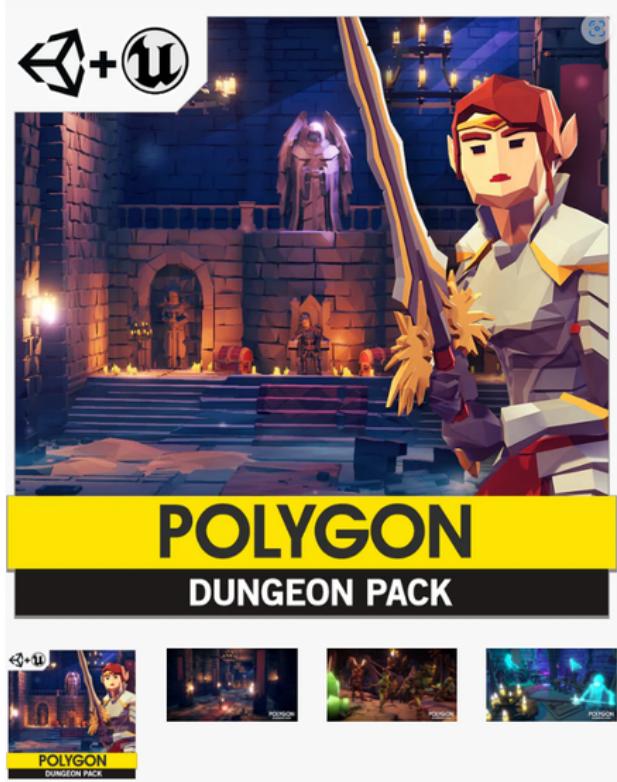
Infinity PBR - Characters
Not Yet Rated 1 of 2 questions answered
Male & female goblin bodies with modular wardrobe, text animations

Add To Project Supported Platforms

OR Supported Engine Version

Write a Review 4.25 - 4.27, 5.0 - 5.2

>



Synty Store
POLYGON - Dungeon Pack

★★★★★ 13 reviews
\$39.99 USD \$79.99 USD

Add to Cart

Add to wishlist

Description Change Log Reviews

Synty Studios Presents -



An Epic Low Poly asset pack of characters, props, weapons and environment assets to create a Fantasy themed polygonal style game.

(826) POLYGON Dungeon Pack - (Trailer) 3D Low Poly Art for Games by #SyntyStudios - YouTube

Visuals and Audio (Animations, SFX, VFX)

Implement Sound Effects for Elemental Spells

Implement Sound Effects for Elemental Spells

 BOOG-105 DONE

Description:

Implementing sound effects for elemental spells enhances the audio experience and immerses players in the game world. This Confluence page overviews the process and considerations for implementing sound effects that complement each elemental spell's visual and gameplay aspects.

User story:

As a player, I want my different senses to receive feedback to fully immerse myself in the game world and enhance my gameplay experience.

Benefits:

By implementing sound effects for elemental spells, we aim to achieve the following benefits:

1. Immersive Gameplay: Sound effects add depth and realism to the gameplay, creating a more immersive experience for players.
2. Feedback and Engagement: Audio cues provide feedback and reinforce the impact of each spell, making the gameplay more engaging.
3. Audio Signaling: Sound effects can serve as audio signals to indicate spell activation, cooldowns, and other important events.
4. Emotional Impact: Well-designed sound effects enhance the emotional impact of spells, making them feel more powerful and satisfying.
5. Accessibility: Sound cues can assist players with auditory preferences or hearing impairments, ensuring an inclusive gameplay experience.

Implementation Process:

1. Sound Design and Selection:

- Collaborate with the sound design team to create a library of elemental spell sound effects.
- Ensure that each spell's sound effects align with its visual and thematic elements.
- Consider the intensity, pitch, and duration of the sound effects to match the desired impact of each spell.

2. Spell Activation:

- Identify the appropriate sound effect for each spell's activation, capturing the essence and characteristics of the element.
- Implement the sound effect to play when the player casts the spell, creating a sense of impact and power.

3. Spell Impact and Interaction:

- Assign distinct sound effects for the impact or collision of each elemental spell.
- Consider variations in sound effects based on the environment or the type of target affected by the spell.

4. Cooldown and Recharge:

- Implement sound cues to indicate when a spell is on cooldown or recharging.
- Use sound effects that convey the concept of the spell gathering energy or returning to full power.

5. Ambient and Environmental Sounds:

- Incorporate ambient and environmental sounds that enhance the atmosphere and reflect the elemental theme of the game.
- Consider sounds like crackling fire, flowing water, rustling leaves, or howling wind to enrich the player's immersion.

Completion Criteria:

The following criteria will determine the completion of implementing sound effects for elemental spells:

1. Each elemental spell has appropriate activation, impact, and interaction sound effects.
2. Sound effects align with each spell's visual and thematic elements, enhancing their impact and immersion.
3. Cooldown and recharge sound cues effectively convey the state of the spells to the player.

4. Ambient and environmental sounds are integrated to enrich the atmosphere and reinforce the elemental theme of the game.
5. Playtesting and feedback confirm that the implemented sound effects enhance the overall audio experience and gameplay engagement.

By implementing sound effects for elemental spells, we strive to create an immersive and captivating audio experience that complements the visual and gameplay aspects of the game, enhancing player enjoyment and immersion.

A sound for buying an item

To acknowledge the player has bought an item it should have a noise associated with it. The same goes for when they don't have enough to buy an item

Completion criteria

On click shop item it plays noise.

The correct noise is played correctly

Bugs

Spell projectile has no movement

Bug Report: Spell Projectile Has No Movement

 BOOG-93 DONE

Issue Description

When casting a spell projectile, the projectile does not exhibit any movement and remains stationary at the point of origin. This bug prevents the player from effectively using the spell to hit targets or interact with the environment.

Steps to Reproduce:

1. Equip the spell that launches a projectile.
2. Aim at a target or an object in the environment.
3. Cast the spell and observe the projectile's behaviour.

Expected Result

The spell projectile should move in the direction aimed and travel toward the target or object, interacting with it upon contact.

Actual Result

The spell projectile remains stationary at the origin point and exhibits no movement.

Reproducibility

This issue occurs consistently and can be reproduced whenever the spell is cast.

Potential Causes

- Scripting or coding error related to the spell projectile's movement properties.

Resolution Steps

1. Investigate the spell projectile's movement script or code to identify any errors or discrepancies.
2. Test and debug the movement functionality of the spell projectile to ensure it moves properly toward the intended target or object.
3. Verify that the bug is fixed by testing the spell projectile's movement in different scenarios and confirming its proper behaviour.

Completion Criteria:

- The spell projectile correctly moves toward the target or object upon casting.

Spell Fireball explodes (Effect only) when shoted

Bug Report: Fireball Explosion Effect Triggered on Player [BOOG-94: Bug Report: Fireball Explosion Effect Triggered on Player](#) DONE

Issue Description: When a player casts a Fireball spell, the explosion effect is triggered immediately upon launch, causing damage to the player who launched it. This bug results in unintended self-inflicted damage and disrupts the intended gameplay mechanics.

Steps to Reproduce:

1. Select the Fireball spell from the available spells.
2. Aim at a target or direction away from the player.
3. Cast the Fireball spell.

Expected Result:

The Fireball spell should travel in the aimed direction and explode upon impact with a target or object in the environment. The explosion effect should only occur when the Fireball collides with a valid target or object.

Actual Result:

Upon casting the Fireball spell, the explosion effect triggers immediately on the player who launched it regardless of impact with a target or object.

Bug Report: All Blockouts Solved with Fire Spell Instead of Appropriate Elemental Spells

All Blockouts Solved with Fire Spell Instead of Appropriate Elemental Spells



Issue Description:

Currently, when interacting with blockouts in the level, all blockouts are incorrectly solved using the Fire spell, regardless of the required elemental spell. This bug prevents players from utilizing the intended mechanics of using specific elemental spells (wind, earth, water, or ice) to overcome the corresponding blockouts, leading to an inconsistent and unintended gameplay experience.

Steps to Reproduce:

1. Encounter a blackout that requires an elemental spell (e.g., wind, earth, water, or ice).
2. Equip the Fire spell.
3. Attempt to interact with the blackout.

Expected Result:

The player should only be able to solve the blackout by using the appropriate elemental spell (wind, earth, water, or ice) corresponding to its requirements.

Actual Result:

All blockouts in the level are erroneously solved by using the Fire spell, irrespective of the intended elemental spell requirement.

Pulling Procedures for Perforce Version Control in Unreal Engine 5.1

Pulling Procedures:

When pulling changes from the Perforce depot for version control in Unreal Engine 5.1, follow these steps to ensure a smooth and efficient workflow:

1. Pull from Source:

- Start by pulling changes from the Source folder in the Perforce depot.
- If any changes exist in the Source folder, open the solution folder associated with your project.
- Compile the solution by selecting a Development Editor Solution Configuration and a Win64 Solution Platform. These options are typically located to the left of the play button in your development environment.
- Compiling the solution ensures that any code changes are properly integrated into your project.

2. Pull from Config:

- After pulling changes from Source, proceed to pull changes from the Config folder in the Perforce depot.
- If there are any changes in the Config folder, close the Unreal Engine editor if it's currently open.
- Reopen the Unreal Engine editor to allow it to recognize and apply any edits, additions, or deletions made to the configuration files.
- This step ensures that your project is using the most up-to-date configuration settings.

3. Pull from Content:

- Finally, pull changes from the Content folder in the Perforce depot.
- This step involves pulling any changes made to the content assets of your project, such as textures, meshes, materials, and blueprints.
- Ensure that you pull these changes after pulling from Source and Config, as they rely on the compiled code and updated configuration settings to function correctly.

By following these pulling procedures in the specified order, you can maintain a consistent and reliable workflow when integrating changes from the Perforce depot into your Unreal Engine 5.1 project.

Team

Team Members

Miguel Martinez Olivares

Malik Carter

Adrian Edwards

Jairo Manon

Donovan Sellers

Sean Jamelarin

Roles and Responsibilities

Miguel Martinez Olivares: Level Design (Obstacles, level blocks, puzzles)

Malik Carter: User Interface (UI)

Adrian Edwards: Animation & Spells

Jairo Manon: Spell/Abilities

Donovan Sellers: Gameplay

Sean Jamelarin: Artificial Intelligence (AI)

Contact Information

Name	Miguel Martinez Olivares
E-Mail	MMartinezOlivares@student.fullsail.edu
Discord	MikeMtz#5037
Phone	+1-321-260-4194
Other	Miguel Martinez Olivares – Game developer portfolio

Name	Sean Jamelarin
E-Mail	srjamelarin@student.fullsail.edu
Discord	ShonenRage#4528
Phone	(850)-598-3830
Other	

Name	Malik Carter
E-Mail	mjcarter3@student.fullsail.edu
Discord	Asura#2884
Phone	984-269-6528
Other	

Name	Adrian Edwards
E-Mail	amedwards@student.fullsail.edu
Discord	Redtheredistofreds#5434
Phone	254-444-8326
Other	

Name	
E-Mail	
Discord	
Phone	
Other	

Name	
E-Mail	
Discord	
Phone	
Other	

Donovan's Game Ideas for the spells town enemies bosses

Ideas on what to add

I had another one that got deleted with only a few more videos and stuff but I felt like those are redundant

Water spell

Movement based kinda like this at 6:04

[YouTube Katara's Waterbending Evolution | Avatar: The Last Airbender](#)

Or here as Todoroki uses ice to travel 0:54

[YouTube My Hero One's Justice: Shoto Todoroki Basic Moves & Combos](#)

Fire Spell

Burns the ground and leaves a pillar of fire on the ground 1:03

[YouTube Skyrim - All Destruction Spells](#)

Wind Spell

Will Pull objects towards you or push them away pull effect like here at 14:01

[YouTube The Legend of Zelda: Twilight Princess - The Gale Boomerang - Episode 9](#)

Levitation Spell

The ideal is like Breath of the wild Magesis but the minimum is lego harry potter

Town Aesthetic

I am thinking European houses like Bright Hoof from tiny tinas wonderland or Attack on titan cities

[YouTube Tiny Tina's Wonderlands - Brighthoof All Collectibles Lucky Dice Scrolls Poetry Pages Marbles](#)

Fire Spell puzzles

Could burn foliage like in this video at 0:53

[YouTube How to Destroy Thorns Zelda Breath of the Wild \(HD\)](#)

Lighitng torches like here at 3:07

[YouTube Legend of Zelda Twilight Princess Walkthrough 04 \(1/5\) "Forest Temple: West"](#)

Water Spell Puzzles

Could use the moment to fly across gaps and stuff like link does with Pegasus boots 10:27

[YouTube How to get the Pegasus Boots & beating Slime Eye in Key Cavern - Link's Awakening](#)

Doors close quickly so you have to use speed like here at 0:25

[YouTube Skyrim Ustengrav Stones Puzzle](#)

Or open up waterfalls or lakes like here in Ocarina of time 5:40

[YouTube Legend of Zelda Ocarina of Time Walkthrough 05 \(2/7\) "Zora's Domain"](#)

Wind Puzzle

It can be to clear things out the way like here at 3:20

[YouTube Zelda Twilight Princess - Hyrule Castle - Part 3 | WikiGameGuides](#)

Or push enemies off or objects

Enemies

Lets keep it simple 2 main types one that is pure Melee and wants to get closer to the player and the other is Pure Range that wants to get away from the player excellent example is Twilight Princess Bulblins

Bosses

Fire spell boss- He's made of wood there is water everywhere and you are trying to just deal damage as he puts himself out and swing his large fist

Water- No boss just an escape sequence? Possibly

Levitation- The Boss charges at you and you have to lift objects from the ground removing armor until you lift him and throw him away

Wind- Like the bullies from Mario 64