

Project Features list

- Document (as a bullet point list) the major FEATURES of your software application.
- Each features should have a TITLE followed by a brief DESCRIPTION of that feature.

Requirements

- Document the key FUNCTIONAL and NON-FUNCTIONAL requirements for a minimum of SIX features.
- Follow one of the formats/templates provided in class.
- Your application will likely have many more features, and you should create requirements documents for all the features.
- But for this milestone, you need only turn in SIX requirements documents.

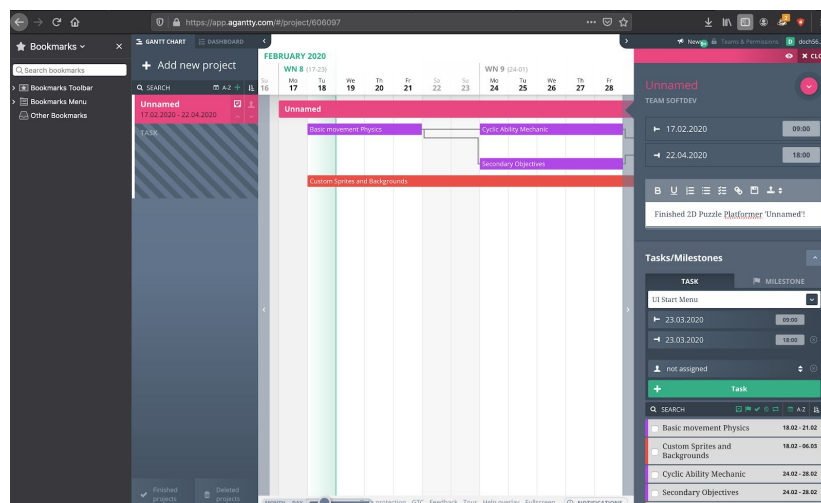
Project Plan

- Create a Project Plan in your chosen Project Management tool. Maybe a GANTT chart, Kanban board, or other suitable format.
- The Project Plan should include each Feature that is to be developed.
- The Project Plan should indicate the SEQUENCE of Sprints. That is, the plan should show in what order the features will be developed (designed, coded, tested, integrated.)
- The Project Plan should show for each feature WHEN (by calendar date) the feature will be worked on.
- The Project Plan should show for each feature WHO will be working on the development of the feature.
- The Project Plan should result in a completed, working application that is ready to present by April 22, 2020

Project Plan

Major Features:

- Basic Movement
 - Move left and right and be able to jump.
- Cyclic ability mechanic
 - Abilities will be cycled through and only one can be equipped at a time
 - Abilities may include:
 - Double jump
 - Air dash
 - Teleport to last point on the ground
 - Teleport to 2nd Player
 - Dash through walls
 - Wall climb
- Local Co-op
 - Up to two players can work together to solve puzzles more easily.
- Puzzle Platformer
 - Must use the changing ability mechanic in the correct order and flow to pass each level.
- Hidden secondary objectives in each level
 - A hidden object that can be collected in every level for extra points
- Custom sprites and backgrounds
- Slide when there is obstacles
- Online Scoreboard



Functional Requirements:

	<u>Functional</u>		<u>Non-Functional</u>
Use Case Name	Communicate	Data Item	<ul style="list-style-type: none">• Resources• Positional• Score• Content/ Category
Title	UX	Type	<ul style="list-style-type: none">• Pointers• Integers• Integer• Text
Description	The user uses the controller to interact with the game.	Description	<ul style="list-style-type: none">• Active ability time• Location of character in level• Number of points User has accumulated• Descriptions of abilities and game flow.
Actors & Interfaces	Controller, Monitor, keyboard	Comment	Encompasses the Users experience.
Initial Status and Preconditions	The controller is connected to the video game and the user has access to it.		
Basic Flow	<ol style="list-style-type: none">1) The user picks up the controller2) User learns the button mapping3) The user plays the game.4) The user enters tag for scoreboard5) User puts down the		

	controller		
Alternative Flows	<p>If the User does not have a controller:</p> <ol style="list-style-type: none"> 1. Use the Keyboard of the computer 2. Learn the key bindings 3. Continue from Step 2 of Basic Flow. 		
Post Condition	<ul style="list-style-type: none"> • Game has ended. • Record score into scoreboard. 		
Exception Paths	<ol style="list-style-type: none"> 1) The ability to pause the game or quit at any time. 		
Other			

	<u>Functional</u>		<u>Non-Functional</u>
Title	Cyclic Ability	Data Item	<ul style="list-style-type: none"> • Selection of ability • Timer • Stat boost
Description	A pop-up menu that the user can choose a specific mechanic to navigate portions of the level's puzzle	Type	<ul style="list-style-type: none"> • Boolean • Integer • Float
Actors & Interfaces	Around 3 mechanic choices, the user's selection, puzzle hints	Description	<ul style="list-style-type: none"> • All abilities are off until one is selected • How long that ability will last • An increase to the player's physics, whether that's adding another jump or increasing the positional velocity
Initial Status and Preconditions	Inside the game loop	Comment	Controls the player's buffs as they choose which abilities to select.
Basic Flow	<ol style="list-style-type: none"> 1. The user holds down a button and a pop-up will appear 2. User selects a direction of which ability they want 3. The user releases the button and the pop-up disappears, empowering the user with the selected 		

	ability.		
Alternative Flows	<p>If ability is disabled</p> <ol style="list-style-type: none"> 1. Continue from 2 2. If the chosen direction's ability is grayed out, then no ability is selected and the audio will buzz. <p>Selecting player difficulty</p> <p>For <i>easier</i> difficulties:</p> <ol style="list-style-type: none"> 1. Continue after step 1 2. Time/Movement slows down 3. Continue at step 2 4. After step 3, Time resumes at its previous pace. 		
Post Condition	<ul style="list-style-type: none"> • User's selection of ability is enabled and decays after a while. • Game is resume its normal mode 		

Exception Paths	User may cancel ability selection by letting go of the button without choosing an ability		

	<u>Functional</u>		<u>Non-Functional</u>
Use Case Name	Puzzle Platformers	Data Item	<ul style="list-style-type: none"> • Platforms • Puzzles
Title	Gameplay Loop	Type	<ul style="list-style-type: none"> • Boolean • Boolean
Description	Each level will have a problem and solution, where the user has to use the cyclic mechanic to navigate the solution.	Description	<ul style="list-style-type: none"> • Type of platform, collision checks. • Checks if player has completed correct objectives
Actors & Interfaces	The character interacting with the platforms and environment <ul style="list-style-type: none"> • Platform interfaces get more challenging as the game proceeds. tel 	Comment	<ul style="list-style-type: none"> • Encompasses the user's goals of how to get from point a to point b in the game.
Initial Status and Preconditions			
Basic Flow	1. The player		

	spawns into the level 2. The player approaches the obstacle 3. The player selects a mechanic to pass the obstacle 4. The approaches the next obstacle 5. Repeat steps 3 and 4 until the level ends		
Alternative Flows	If the player does not follow the flow 1. The character dies 2. The level restarts		
Post Condition	The level is finished and the player may move on to the next level		
Exception Paths	Users have a choice to use the teleportation ability to revive and resume playing.		