

| [No changes from previous milestone] | | | | |
|--|--|---|---|---|
| Game Code Name: Survival Horizon | Group Members: Kashan, Dj, Ganesh, Diwakar | | | |
| Target Platform(s): Ue5 / Unity | | | | |
| | | | | |
| 1. Genre <small>(1st person shooter, Puzzle, Sidescroller, etc.)</small> | 3. Player Types <small>(ranked by importance)</small> | 6. Core Pillars <small>form "User can..." (in the</small> | 7. Core System Breakdown <small>(ex: Combat has Melee/Range)</small> | |
| First Person Shooter (FPS) | Action Gamers - Seeking intense combat and quick reflexes challenges | Users can move around with enhanced mobility | Combat System: Engine-specific animation system (UE5: Control Rig / Unity: Mecanim) <i>Physics-based shooting mechanics. Hit Detection & damage system</i> | |
| Survival Horror | Strategy Players - Resource management and tactical decision making | Users can combat zombies in close range | Wave Progression System: AI behavior trees for zombie types, difficulty scales dynamically, Object pooling for efficient zombie spawning | |
| Wave-based Combat | Explorers - Interested in discovering secrets and collecting coins to get upgra | Users can navigate and explore | Resource Economy(Assets): UI system for points/currency display | |
| Roguelite elements | | Users can progress | Saving Systems: Character stat management, resource tracker, use engine serialization to save system | |
| | | | Solo System | |
| 2. IP/World/Environment <small>(high level of environment, characters, style, history, etc.)</small> | 4. Player Experience <small>(in the form "User will feel... when they ...")</small> | and share resources | | |
| Post-apocalyptic modern setting with modular level design | Feel like a skilled survivor against overwhelming odds | | | |
| | Excitement from dynamic gameplay situations and tense moments | | | |
| Dynamic weather system using engine particle systems | | | | |
| Real-time lighting for atmosphere | Satisfaction from mastering wave progression and resource management | | 8. Required Hidden Systems <small>(ex: AI, Save/Load, Navmesh)</small> | |
| | Experience intense moments of both action and horror | | AI: Behavior Trees for complex AI decisions, performance optimization for large hordes | |
| Multiple paths and routes, hidden areas for exploration, | Pride in discovering optimal strategies and secret techniques | | Pause Screen: Async saving to save progress when game crashes and want to reset checkpoint, Auto-save functionality | |
| vertical gameplay elements, interactive environment objects | UI that allows players to change the Frames Per Sec in the game, allowing lower quality pc's to be able to play the game | | NavMesh: Obstacle avoidance, perhaps dynamic updates throughout the game, several different types of navmesh with different behaviors | |
| strategic positions for combat | UI that allows players to mute the music and sfx | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| 5. Inspirations | | | | |
| <small>(Use movies, games, comics, etc. to describe key elements of the game, remember to include what is targeted - ex: UI inspired by LoZ: Breath of the Wild)</small> | | | | |
| Call of Duty: Zombies (Wave-based survival mechanics) | | | | |
| Resident Evil (Resource management and environmental storytelling) | | | | |
| Left 4 Dead (Special zombie types and AI director) | | | | |
| Dying Light (Movement mechanics and day/night cycle) | | | | |
| Dead Space (Horror atmosphere and upgrade systems) | | | | |
| Color Coded Grading Elements | | | | |
| Inspirations (#5) | Genre (#1) | User Types (#3) | Core Pillars (#6) | Core Systems (#7) |
| Inspirations should state what element of the project they are inspiring. <i>Example: Red Dead 2 (UI)</i> | Look for things that don't belong. <i>Example: World descriptions, Features, Systems.</i> | User types should be tied to User Experiences and Core Pillars bu | These should be in the form of "User Can..." The items don't need that phrase but should flow from it. <i>Example: Explore a vast world of differing climates.</i> | Core systems must support a core pillar or key functionality. Example: Load/Save is not a pillar but is a key functionality. |
| Inspirations should have other elements like Genre or general World ele | Look for missing elements: <i>Example: 3rd person but no genre like RPG or Action.</i> | Check the Core Pillars and Experiences to ensure a type is not mi | Each element should be related to a specific user experience and usually fulfill a user type | Think of elements that support key project infrastructure or system that are inferred by Core Pillars. <i>Examples: Travel app that visits various sites in a Country needs a transit system between sites.</i> |
| | Think about Camera and Genre/Class mainly | The Type should be followed by a short description. Example for | Each element must describe a core element of the project, remove it and the project cannot deliver the expected experience. | |
| | | | | |
| | | | The pillars should include the key elements, review the player experience, World Description, and Core systems to see if they describe something that is key but not listed in the Pillars. An example might be multiplayer for a game or activity driven for a travel app. | Remember Core Systems is a means of informing the team what systems need to be developed/architected. |
| | IP/World (#2) | User Experience (#4) | Don't let the core pillars get too low level or you are taking away decision making from the team, keep the pillars high level, giving the game structure without too much detail. | Hidden Systems (#8) |

| | | | | |
|--|---|---|--|--|
| | Each element should describe the background / World, giving context to the project. Review the other areas to ensure the correct elements are present. <i>Example: Genre says RPG and World says Post-Apocalyptic, missing near future or far future to help define the world/tech.</i> | These elements should describe the emotions elicited as users experience the key areas of the project. | | These are your standard systems that don't make the game or the app the app (as in core systems) but they are needed for the project to function properly) |
| | | The Core Pillars should tie to the Experiences, all of the experiences should be there, one experience can support multiple core pillars. | | Examples include animation, AI, voice, cinematics. We can't name all but we should name those key systems that support the gap in general. |
| | | This should only be for experiences, not other areas of the canvas. <i>Example: a core pillar, system or type is listed in this area</i> | | |