

**Background**

We have a team of 3 programmers and 5 artists, specializing in various aspects of game design. The programming team and art team have worked together in their own groups respectively but this will be the first time the teams will combine to work together. Each team member has a good work ethic and everyone has a good understanding of what the end goal is and how to achieve it. We’re enthusiastic and passionate about what we do and we have great communication between the team members.   
  
Extra notes:  
Xavier - build master, source control  
Keil - producer

**Premise & High Concept**

You act as a thief in a marketplace trying to get rich quick by stealing items and stashing it away while evading detection and drawing too many eye-witnesses. The player can choose to stealth their way through or blow through the level by running and jumping—running a higher risk of getting caught.

**Game Design**

Core mechanics/concepts – stealing, blending in, “stealth”, running, jumping, wall climbing, sliding

Player experience – as you steal you run higher risk of getting caught but you get a higher reward.

Game reaction – efficient/timely, AI react according to the players actions

Player package/key gameplay – pickup, walking, sprinting, climbing, throwing, sliding, interact, drop, stash

**Art Direction**

Art direction

Themes and concepts inspired by animated film spirited way (2001). Actual assets will be simple and cell-shaded low-poly with details falling onto signage and patterns (ooblets).

Concept > model > UV > texture > check UE4 compatibility > deliver

**Technical Direction**

Technical systmes and tools – UE4, and whatever the artists need

Division of labor – matt will be working on player controller, keil working on AI, behavior trees. Xavier will be backend programming.

Github, tool gitbash.

Agile structure – design > build > test, rapid iteration cycle

**Timeline**

|  |  |  |  |
| --- | --- | --- | --- |
| **Week** | **Art Goals** | **Prog Goals** | **Build** |
| Week 1 | Basic greybox and lighting set up. Test environment. | One functioning AI, player package, interactions and score system. | Player movement and basic controls functioning. |
| Proto | Wireframe for all the assets, large items and a few of the smaller items. Containment environment done. Basic design for AI. | Merchants and wandering AI completed. Basic win/lose state. | Player can interact with “enemies” or other secondary gameplay features. |
| Week 3 | Models for assets complete (majority of textures done). Animations done by end of the week. | Fully functioning alpha, base model for game. | Player “finished.” |
| Alpha | Polishing models and animations, adding on remaining textures. | Enhancing interaction. | Fully textured and animated gameplay with all the necessary feedback. |
| Week 5 | Additional assets for microinteractions and animation variations. Player character (3rd person view) | Variables, juicing. Fluid movements and particle effects (dust, cloth physics, cell-shading). Implementing loss screen. | Improved gameplay tangibility and responsiveness. |
| Media | JUICE! - additional signage, tweaking details | JUICE! | JUICE! |

Appendix

**Risks and Challenges**

Risks/challenges - animation, rigging

Problems, concerns - animation

**Contingencies**

How will your team respond to hiccups in production and failures to meet goals? Do you have a backup plan? Are tasks and goals well prioritized?

**Google**  
Realizing minimum viable product

**Asset List & Art Bible**

Exactly how many assets and of what complexity and type do you expect the project to need? Can you briefly list all of them? Can you rank the importance of each asset in relation to the player experience that your team is authoring?

Create a collection of concept art and works to draw inspiration from to create an art bible. The art bible should help with visually establishing your art direction.

WIP

**Features and Systems**

How many individual coded interactions will be needed to express the gameplay mechanisms? What sort of techniques do you intend to use to solve them? List and briefly describe.  
  
42