A Game by

James Halliday, Quincy Buchanan, Avery Follet, Evan Goldring and Thomas Macmillan

P A N Z E R P A N I C !

Document Version 1.0

Delivery Platform

Panzer Panic! is being developed for PC, console and arcade. Although we can expand our platform if need be.

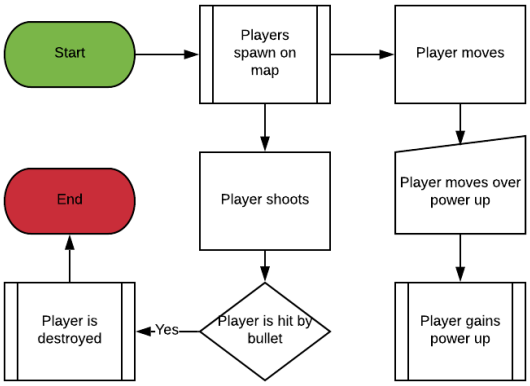
Development Environment

Panzer Panic! is being developed in a 2-dimensional space on the Unity Engine by Unity Technologies. Specifically, Unity 2018.12. We are using this engine because it has a solid foundation for developing 2D games and because it is very good for developing and building for multiple platforms.

To handle joystick controls we are using the Rewired plug-in for Unity as it allows for easy management of joystick inputs across multiple players.

For version control, we are using git-based repository accessed through GitHub.

Logical Flow



Game Mechanics & Systems

Panzer Panic! Is a multiplayer competitive game where players control tanks on opposing sides with one player acting as the driver and another player acting as a gunner on each side. The game reaches an end state when one team destroys the other.

- The driver can use their controller to move the tank around the map and pick up and use powerups that spawn at key locations.
    - *Explosion* – Turret shots explode on impact.
    - *Double Shot* – Turret shots split in two on fire.
    - *Speed Up* – The tank moves faster for a short period of time.
- The gunner can rotate the turret and press the "fire" button to fire a shot from the turret. Shots can be fired in one second intervals.
- If a tank is struck by a bullet, it is destroyed and the opposing team is declared the victor.

Art Pipeline

The art for *Panzer Panic!* will be implemented through the repository we have set up. The artist will place completed assets directly in the project folder. The artist can then commit the assets to the main repository where the programmer(s)/designer(s) will be able to implement the assets into the game space.

Design Pipeline

The designer(s) will be able to put content in the game by having direct access to the project folder through the main repository. While they are within the Unity project, they will have the freedom to add or remove or adjust whatever their heart desires. Most of the important object scripts have public variables attached to directly alter game balance whenever necessary.

Various other important objects are designed by the programmer as prefabs that the designer can use to their whim.

Milestone Updates

- Create playable map prototypes. (done)
- Map tank controls across controllers. (*always iterating*)
- Create in-game level selection. (done)
- Create and optimize power-ups. (*in progress*)