Overview:

Goopwar is a falling tromino puzzle game that focuses on simple, fast-paced action instead of intricate combos.  It allows players to remove any cluster of blocks touching the bottom of the stage with the press of a button.  There will be three different colors for the players to sort into as large of a cluster as they can before cashing out.  The game will be primarily competitive, with AI and PvP modes available.  The puzzle mechanics will fuel offensive and defensive abilities between the two players as they race to score as high as possible in short and quick games.  The game will end on either a timer limit, score limit, or the depletion of player HP, or the filling of their game board.  The game will have two-player battles, a vs AI story mode, and a single player arcade mode.

Gameplay:

The puzzle mechanics fit into the game world as fuel for a tank (or similar manned-machine) battle between the two protagonists.  As the player matches the colored gel, it is placed into reservoirs to be used in the battle.  Using the gel places advantages or disadvantages accordingly upon each player, and/or changes their health points.  It will be imperative that the players prioritize which color they collect based on what is happening on the larger scale battle.

The goal of the game’s design will be to keep the puzzle aspects simple and satisfying.  Challenge will come from perfecting the system to get more efficient color chains is less time, or in the battle that is taking place between the two players.

Aesthetics:

The art style of the game will be cartoony, with a focus on machine aesthetics.  The game will feature different avatars displayed as portraits during gameplay.  These will be the pilots of the fighting machines.  The game board should be displayed as a type of engine, with pipes leading goo in and out; The idea being that it flows into the tile board, is sorted, and then flows out to the scoreboard and reservoirs.  Ideally, all menus, GUI, logos, and typefaces will represent this machinist art style.  The goo’s aesthetic will stand out as brightly colored blobs of an oily substance.  They will group together when resting side-by-side, as in puyo-pop.  The actual puzzle game should be very expressive with visual feedback.  Cashed out combos should explode into flames, Identified clusters should grow into larger blobs, and when players accelerate blocks to the bottom they should have a visible and audible ‘splat’ associated velocity of gameplay.

Development:

The driving factor in the development of this game will be to keep it within small scope.  With the gameplay defined, a large factor of the game will fall into audio and visual design.  Executing these sections of the game correctly will help the simplicity of the game create a smooth and friendly experience.  Thus, logical development of the game should be relatively short in regards to man hours.  Assets are created, time will need to be spent on their accurate integration into the game, alongside any additional visual effects to make them pop.  Visual assets will be acquired via online store or by commissioned wherever polished graphics are expected.  Audio assets will be licensed or created personally, depending on value proposed case by case.  The goal of the game is to have the game released on Steam for PC in summer of 2018, with a viable product available by Spring of 2018 if possible.  This will prove capability of creating a marketable product, build presence online, and generate income for future projects.

Marketing:

With knowledge of market saturation taking place on the Steam marketplace, we recognize the challenges of standing out in a flood of products.  To stand out, it is important to develop a unique art style that is capable of creating eye-catching gameplay scenarios.  Every screenshot should be interesting enough that potential players are interested in what is happening.  To hit this mark, art will need to be synchronous and the game will need to be well polished.  The game will be developed with a 5 - 20 CAD price tag in mind based on the perceived value of the final product.  The game will be sold as a low commitment transaction that promises simple and fun mechanics for the price point.  Coupled with simple social media advertising the product should be able to generate moderate sales among a niche market if well developed.

Risks & Challenges:

* Unknown factors of widespread PC development:  With few available PC’s it will be hard to ensure that the game will run with optimum capability on all systems.  Using the unity engine will involve trust that the engine can provide reliable results.  Likewise, as we are developing a controller based game, we will need to consider how to implement a breadth of control options.  Resolution scaling and control customization will be necessities.
* Insufficient Time Allotment:  With this project being predominantly developed on the side, schedules are made with low expectations of commitment in mind.  The game will need to be kept small in order to meet these goals.  Recommend to assess progress against expectation monthly.
* Asset Starvation: Without guaranteed access to required assets, be prepared to make compromises or to allocate a budget based on what is expected.

AI Implementation

AI will be implemented using a state tree traversal of virtual boards using A\* search.  Using a specified depth, the board tiles will be mapped into a 2D array of sub-states.  A set of operations will be defined to emulate both the progression of the game and allowable inputs.  The AI will have an assessment algorithm that values the state of each tile in relation to other states on the board.  Colored states will be valued based on the score that they can provide.  Empty states will be valued based on their potential for each match.  The more adjacent tiles there are of the same color, the higher potential value there is for the empty space.  Empty tiles will also have an implicit value associated to them based on their position on the board.  Values near to middle and the top will have a higher value to prevent tile blockage.

The AI will construct its state tree based on the ability to move left, right, or down.

The AI will have a separate module that decides when it should cash out a combo of tiles.  Based on a cost-benefit evaluation of any each color combo, the combo will be triggered.  As the combos fill the move meters, they will have a similar trigger based on their effects.  This module will be able to evaluate the opposite player’s board to decide between different states of behavior.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 12 | 16 | 24 | 32 | 24 | 16 | 12 |
| 12 | 16 | 24 | 32 | 24 | 16 | 12 |
| 6 | 8 | 12 | 16 | 12 | 8 | 6 |
| 3 | 8 | 12 | 16 | 12 | 8 | 3 |
| 3 | 8 | 12 | 16 | 12 | 8 | 3 |
| 2 | 4 | 6 | 8 | 6 | 4 | 2 |
| 2 | 4 | 6 | 8 | 6 | 4 | 2 |
| 2 | 4 | 6 | 8 | 6 | 4 | 2 |
| 2 | 4 | 6 | 8 | 6 | 4 | 2 |
| 1 | 2 | 3 | 4 | 3 | 2 | 1 |
| 1 | 2 | 3 | 4 | 3 | 2 | 1 |
| 1 | 2 | 3 | 4 | 3 | 2 | 1 |
| 1 | 2 | 3 | 4 | 3 | 2 | 1 |

Fig 1. Intrinsic Value for Empty Tiles

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 |
|  | 1 | 1v | 3 | 4 |
|  | 2v | 3v | 3 | 4 |
|  | 2v | 3v | 1v | 4 |
|  |  |  |  |  |

Fig 2.  Scoring of the bottom corner of the board, holding a cluster of red tiles.  Each red tile evaluates to v, a given value times the number of like tiles bordering the tile.