

Prototype: Slippery Bash

Team Ice Guys

Omar Ahmed

Tarek Elsherif

Moritz Naser

Akbar Suriaganda

Technische Universität München

Contents

- Goal of the prototype..... 3
- Implementation of rules..... 3
 - Dynamic stage but discrete movements..... 3
 - Difficulties of implementation..... 3
 - Surface and Movements 3
 - Durability 3
 - Round-based Battle Royale 3
- Events 4
 - Titanic 4
- Gameplay..... 4
- Testing and evaluation 5
 - Situation 5
 - Results 5
 - Action Availability..... 5
 - Stage size 5
 - Stage durability..... 5
 - Difficulty of movement..... 5
 - Defeating and recovery 5
- Conclusion 6

Goal of the prototype

The game Slippery Bash is a competitive multiplayer game. Thus, it is very important to balance all its features, or otherwise, the game will quickly evolve to a one-sided massacre. There are a couple things that can be balanced within the game:

- Action availability
- Stage size
- Stage durability
- Difficulty of movement
- Defeating and recovery

Our goal of the prototype is to test these situations and see their strength and weaknesses.

Implementation of rules

The prototype will be physical as required. However, since it's impossible to play a dynamic and simultaneous situation on a board game, we need to convert most of its mechanics.

Dynamic stage but discrete movements

Difficulties of implementation

As in the game, the stage can be broken by jumping or collision with a ship. However, we cannot easily simulate waves and slopes on our board game, which is why we left this feature out. Also, the destruction of the stage is meant to be physically generic and affects the whole connected island. Since that is also difficult to simulate, destruction is only applied locally on a single position. So, the base of our board game is a hexagon patterned surface on which we create the stage with single hexagon tiles. The start positions of the stage tiles are marked on the board and forms a diamond made of 49 hexagon tiles.

Surface and Movements

The tiles also represent the position and distance on the stage. Only one player can be on one tile and actions can bring a player from one tile to another. What is more, the type of surface of the tile also effects on the player's movements which are described later.



Durability

After all these changes, the feature of stage durability can be still tested. Each tile can be damaged and is then marked for cracks. For this prototype, we allow a tile to have 2 cracks before breaking on the third hit. A tile that is surrounded by water (no neighboring stage tiles) sinks on the next round after the last connected tile is gone.

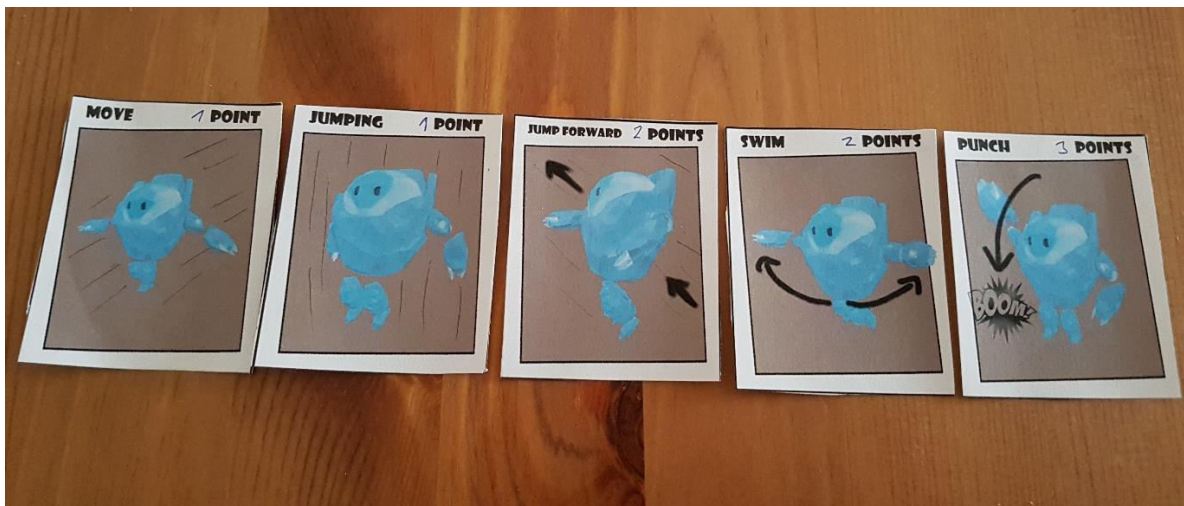


Round-based Battle Royale

A round-based game is the closest we can do to simulate Slippery Bash. Each player can activate action cards only within their turn. The cards should represent the actions the players can do in Slippery Bash. In the actual game, these actions have duration that needs to finish before the player can do something else. So, within a period, a player can only do a certain amount of actions. That is

why the action cards also have a cost and are only available if the player can pay it. For this prototype, we give the players 4 action points to spend them on these actions:

- Walk (1 point): Move to a neighboring tile. If the tile the character lands on has an ice surface, the character automatically walks another tile forward. The player can prevent this by spending another action point.
- Jump (1 point): Jumps on the same place. Mainly used for damaging the tile the character stands on.
- Jump forward (2 points): Move 2 tiles forward, no matter what is in between. Surface of the landing zone affects this action like on Walk. However, the landing zone gets a damage.
- Punch (3 points): Push a neighboring player 2 tiles back. The tile the enemy lands on gets a damage and can break right underneath the enemy. If the landing zone remains and has an icy surface, the enemy slides 1 tile further.
- Swim (2 points): Ice Golems are too lazy to swim. But they must, if they got pushed out from the stage and need to go back. This action card must be used when the character is on water in order to move 1 tile.



Events

For the prototype, we chose to only have the Titanic event because Storm and Snow require complex or random generations. This should not be a big problem of balance testing, since Storm is a threatening event and Snow a recovery one. In summary we don't miss a thing.

Titanic

Throughout the whole game, the ship is repeatedly crashing on the stage. At first, it spawns at one of the 6 start tiles that are marked on the board. To determine where, one player needs to dice the position. Then after every round, it moves a tile towards the stage. When the ship lands on a stage tile, the whole line of connected tiles in front of the ship sinks immediately. The tiles next to the ship also gets one damage. For the rest of the round, the ship sinks before it respawns at the next round.



We need an extra rule for spawning, since we cannot let the ship always spawn on the same 6 positions. Otherwise, the ship will eventually always go through without crashing. That's why, if the line of the start position is free of stage tiles, the start position moves to the closest line with a player on it. If there are multiple lines with this condition set, the players need to dice. The line is chosen which sum of the amount the players on it have diced is lower.



Gameplay

1. Beginning

- 1.1. Lay all stage tiles upside down onto the marked tiles on the board.



- 1.2. Flip the stage tiles to reveal their surface properties.



- 1.3. Place the characters



- 1.4. Determine the first start position of the ship



- 1.5. Determine the order of turns



- 1.6. The highest number starts the main loop

2. Main Loop

- 2.1. Player's turn: Use the action points to activate the action cards (not mandatory)



- 2.2. Repeat with the player that had the second highest dice value

- 2.3. ...

- 2.4. After the last player's turn: Iterate the Titanic behavior described above

3. End condition

- 3.1. Only one player left

Testing and evaluation

Situation

Due to our current situation with COVID-19, we could not test our prototype in a proper manner. So, we tested it on our own and all the characters all played by one person. We hope of getting productive results, though.

Results

The game took about 2-3 hours with about 30 rounds. One reason for that might be that we constantly take notes and pictures of everything. Also, the tester is alone. However, it is still probable to play it at least for one hour in a normal situation. The idea of Slippery Bash is about playing short but multiple rounds together which can not be achieved with a game of that duration. Obviously, the reason for this is the balancing. So, let's go through all our focus points. In order to evaluate the result, we need a reference value that is compared with. We chose the distance a player can walk in one turn, which is regularly 4 tiles.

Action Availability

In a round-based game, every action makes a huge difference. Which is why most turns end up being a skip. It is always better to wait for the opponent to waste action points to come, so I can attack them with full power when they are close enough. In our prototype, the player can attack only opponents that are max 2 tiles away (regularly). Not very much for a quick attack. For the testing, we really needed to force "stupid" movements ourselves, so something happens. For improvements, maybe lower the cost of the punch actions to two, but restrict the player otherwise, like automatically skip after a punch. The other actions felt right in their availability.

Stage size

The stage size affects the balance between action and silent situations. If it is too big, there may be long silent situations before something happens. We aim for an active packed game, so our stage mustn't be too big. However, this was not the case because it took only 2 rounds before 2 players meet. So, the size of our stage is set correctly.

Stage durability

The destruction of the stage felt good. The importance of the choice of actions grew over time with the decreasing amount of time. There was, though, a time in the beginning where it's boring before the action starts. Maybe find a way to make the beginning more interesting. (In other Battle Royale games, for instance, the player is busy with looting in the beginning.)

Difficulty of movement

"Slippery and wet" is the theme we want to realize. We initially wanted to make things difficult with slippery surfaces. The icy tiles of our prototype, however, gives more advantage than disadvantage. While testing, we prefer walking on ice since they bring more distance. Only at the end when the stage is small, it is quite inconvenient because you need to pay twice. Otherwise you might slip into water. But most of the time, these tiles are the player's favorite. Whether it's bad or not is unclear. Maybe we go to another direction with our design and it's good. Let's consider this.

Defeating and recovery

By this, it is meant how hard it is to kill an opponent, and on the other hand, how hard it is to go back to game after you've been punched into water. The result was: It is very hard! For both sides. This is the main reason why the game took so long. These Ice Golems just don't want to die. So, also for this situation, we had to force ourselves to do "stupid" movements to continue the game. But even then, the punched Golem is always able to swim back. Only at the end, when the attacker blocks the only

tile that is close, one Golem finally drowns. Sometimes, we even need to destroy the stage on our own. On the other hand, it is still hard to recover. Even though you swam back, you're not able to evade the next attack. Then you get punched again and the loop continues. Until the perfect ship finally arrives to literally break the loop. So, we need a feature that will prevent the loop. A feature we've not yet found. But it's enough to improve the ability to kill. Thus, the total game duration is shorter, and the fallen players may rejoin.

Conclusion

Despite our doubts, the physical prototype has brought us productive feedback. Even though we had to adjust many things, the results apply also for the actual game. Also, many questions have been asked during our prototype brainstorming that needs to be answered for the actual game as well. So overall a good experience, which would have been better in a non COVID-19 situation.