

Run Screaming

Game Design Document

[High-Speed-Elevator Pitch](#)

[Executive Summary](#)

[Diversifier](#)

[Gameplay](#)

[The World \("The city"\)](#)

[Player 1 \("The chosen one"\)](#)

[The Other Players \("The party society"\)](#)

[The \(printed\) Map](#)

[The Exit \("The Party"\)](#)

[Landmarks](#)

[Look and Feel](#)

[Player 1:](#)

[Other players:](#)

[Platform](#)

[Development System](#)

[Required Artwork](#)

[Team](#)

[Appendix 1: Story Ideas](#)

[Appendix 2: Milestones](#)

[Milestone 0: 2015-11-04](#)

[Milestone 1: 2015-11-27](#)

[Milestone 2: 2015-12-..](#)

[Milestone 3: 2016-01-30](#)

[Milestone 4: 2016-02-15](#)

[Milestone 5: 2016-02-27](#)

[Milestone n+1 aka: The Bonus Content](#)

High-Speed-Elevator Pitch

Help! Your party is running out of cold beer! And your friends agree: *You are the chosen one, chosen to save the night*. But now you are on the other side of the city, with the desired crate of beer, but lost in chaos and disorientation. To run fast and not get lost at the same time is impossible. And worst of all: the beer is getting warm! In order to reach the party on time, help is needed! Luckily, the party society is at hand via phone to guide you ... back to the party ... to deliver the (hopefully still cold) beer ... and save the night!

Executive Summary

The game is supposed to be played by multiple (3+) players.

One player ("**Player 1**") is sitting behind a desktop computer, the others ("**The Other Players**") should not be able to see onto the screen.

The main goal of the game is to pilot Player 1 through a labyrinth, which features some landmarks. The other players have access to maps (either via printouts or by accessing some website (on their phone)) which show some of these landmarks. But every player owns a different map! First, the players with the maps have to identify the correct map and location at which the playing player on the computer has been spawned. Second, they have to navigate the player safely to the goal. To make the game harder, there is a time limit for reaching the goal.

Diversifier

"Can I Try?: Any spectator of the game automatically becomes a player as well. "
... by handing out maps to all players "spectating".

Gameplay

The game takes place in a maze. The maze features several landmarks to help the player to find their position and eventually their way out.

The World ("The city")

The world is a traditional maze. Brick walls, grey stone floor and dark night sky with a moon. The map is created by a maze generation algorithm, and revised by level designers. Landmarks, the spawn point and the exit are placed semi-randomly during that process.

Player 1 ("The chosen one")

The player on the screen, controlling the actual actor in the level. In the same (real world) room as *The Other Players*. The actor on the screen is carrying the crate of beer which has to be delivered to *the party*.

The Other Players ("The party society")

1, 2 or more people sitting in the same room as *Player 1*. They don't control any actor in the level. In possession of *The Map* they help *Player 1* to navigate through *the World*. To do so, they first have to pin down the (randomly selected) *World* and *Player 1*'s initial location in *the World* by comparing the surroundings with striking landmarks on *the (printed) Map*. After doing so, they have to help *Player 1* to reach the Exit as fast as possible.

The (printed) Map

The Map (or maps) is a simple 2d blueprint of *The World* printed on paper or shown on a mobile device. The map is created semi-automatically by game-designers at the same time as *The World*. It gives *The Other Players* additional information about *The World* still unknown to *Player 1*. Mainly an overview of *The World*, the location of *The Safe Exit*, and some/all(?) landmarks. It does not tell *The Other Players* the initial or current location of *Player 1* in *the World*, though. This information can be split over multiple versions of *The Map*. Thus, since multiple *other Players* hold different copies of *The Map*, none of them knows everything. The more *other Players* join the game, the easier it gets to keep track of the world around *Player 1*.

The Exit (“The Party”)

The primary game objective is to help *Player 1* to reach the party before the beer gets too warm. *The exit* is a (randomly selected) location somewhere on the edges of *the world*. To allow *the other players* to guide *player 1* to *the exit*, it is also marked on *the map*.

Landmarks

Mailboxes, garbage bins and street lights are spread all over the map to give the *player 1* something to describe to their fellow *other players* to help the pin down their location and direction. Mailboxes are rare, but in contrast to the other landmarks they are uniquely marked (numbered) and therefore allow definite pinning down of the player’s location (if the correct map has been identified).

Look and Feel

First person controlled player in a city-themed 3D maze at night. The maze features several landmarks to help the player to find their position and eventually their way out, for example post boxes and various street signs.

Player 1:

Played in an 3d world in first-person perspective.

Other players:

The other players don’t need a computer. They sit in the same room as player 1, but instead of a computer they are equipped with sheets of paper. On each sheet of paper ...

Platform

The game is developed using Unity 3d. Although it will result in support for various platforms (<https://unity3d.com/unity/multiplatform>), the development & testing is focused on a WebGL version.

Development System

The game is developed using Unity 3d, mainly on Windows. In addition the Linux version of the development tools (although only available as experimental build) was used (<http://blogs.unity3d.com/2015/08/26/unity-comes-to-linux-experimental-build-now-available/>).

The game development process, including design documents, concepts, story, artwork and source code, is carried out on public platforms, for example GitHub (<https://github.com/RunScreaming/>).

Required Artwork

Due to the rather simple nature of the setting, Unity 3D assets are used for the map textures and ground. Furthermore, the skybox is featured by a standard Unity 3D asset. The landmarks are modelled by our inhouse 3D designer. In addition, UI Graphics are created by our senior aMaze master.

Since audio is a risky factor, the game is a party game to be played in a noisy room, and no audio expert was at hand, we simplified our audio setup by using no audio at all.

Resources:

- <http://opengameart.org/>
 - <http://opengameart.org/content/dance-music-1>
 - <http://opengameart.org/content/streetsound-music-futuristic>
- <http://www.freesound.org/>
- <https://github.com/RunScreaming/>
- <http://spektral.at/musikarchiv>
 - dubmood, kick de bucket

Team

Name	Role	Notes
Anja Karl	Head of Unity	unity on Windows
Roland Urbano	Unity Best Boy	unity on Windows
Peter Grassberger	Senior aMaze Master	unity on Linux, 2D GFX
Stefan More	“Researcher”, Story, Docs	unity on Linux, Testing, GIT
Lukas Neugebauer	3d art	Blender Ninja, 3D GFX

Appendix 1: Story Ideas

Codename	Run-Motivation	Enemies
“The obvious one”	Zombies, Bio-Foo	Zombies, Traps
Uni-Life, #1: Party Life	Warm beer, phone battery, late for Party, ...	Alcohol, loud music, ... ?
Uni-Life, #2: Academia	Some Deadline (Paper, Project, ..)	“Distractions”: Friends, Professors, other Deadlines, Parties, Sickness, more exciting lectures, ...
Moon Base	Running out of oxygen	Dying environment, something your base found?
Underwater Base	Running out of oxygen	Dying environment, scary underwater-animals, something your base found
Underground Research base (see: <i>Half-Life</i>)	Experiment gone wrong	Headcraps! Dangerous environment! Mutants! Confused scientists!
Medieval castle	Your prince is getting married!	Crocodiles, Swamp, Drawbridge, ...
Keep running or everybody explodes	Run, or everybody explodes!?	
Open world motor-biking	Win the race against the	-

	clock	
Dream	Dont wake up	something to wake you up
Vulcano (Indiana Jones)	Lava	Traps, Arrows, Villains, ..
Postman	Deliver all the post	Other Post-Persons? ...
Library		
<i>your favorite PC game</i>		

Appendix 2: Milestones

Milestone 0: 2015-11-04

First draft of Game Idea.

https://docs.google.com/document/d/1Av_Bsn_tiRX_xHnQ8h3AeJ_XR4MLINMTdViNGzcQ6KI/pub

Milestone 1: 2015-11-27

Game Design preview.

DOC: First draft of Game Design Document.

STORY: brainstorm ideas

GAME: -

ARTWORKS: -

Milestone 2: 2015-12-30

Technological preview.

DOC: Small adjustments based on prototype.

STORY: narrow down ideas based on prototype.

GAME: First prototype of game. First or second iteration.

ARTWORKS: First prototype artworks. No sounds.

Milestone 3: 2016-01-30

Second prototype. Playable preview.

DOC: adjustments based on prototype.

STORY: 'Stable' story.

GAME: Later iterations. Internal tests. Maybe, first external tests.

ARTWORKS: Asset-list freeze. First sounds.

Milestone 4: 2016-02-29

Beta release.

DOC: 'freeze' (but write down notes for later versions)

STORY: 'freeze' (but write down notes for later versions)

GAME: Feature freeze. Extensive external tests.

ARTWORKS: Final assets. Sound tuning.

Milestone 5: 2016-03-02

"Final" release & presentation.

Milestone n+1 aka: *The Bonus Content*

If there is time, or if there is a version 2.0, some of the following ideas are considered to be implemented. Otherwise those exist to be implemented by those who have time:

- When more players access the game (while it is already running), the game could e.g. generate items which have to be picked up to open the goal. These are only shown on the new maps.
- There could be different modes.
 - e.g. one mode which always takes place in the same HUGE labyrinth. Monsters and Traps, etc. are generated randomly by the computer and at the start of the game, everybody gets a printed empty map, and the camera is zooming out of the labyrinth/moving over it. Now everybody has to note down as fast as possible, where those monsters are located. Nevertheless, there are too many monsters to note everything down by yourself, so you have to work together and split the task.
- There could be special items, which are needed for example to be able to defeat monsters. (E.G. Red sword for defeating zerberus or something like that)

- Some easter eggs hidden in the labyrinth
- Limited view for the computer player (player does not see traps or monsters unless they run into them).
- Possibility to “fight” monsters. Every monster has a special weakness which is only known to the player with the correct map. (Complex weaknesses, like press a - b - jump - c or something like that)
- add other diversifiers, for example via dramaturgy.
- more open level / open world