

Introduction

In order to elicit our requirements we required input from both the product brief and our customer, to ensure that all needs were met. This allowed us to form a "Single Statement of Need", which details what the system will provide, how it will be provided, and what a completed system will provide.

Single Statement of Need

The system shall enable customers to traverse and escape a maze, modelled on features from the University of York. The game should be user friendly and easy to play, limited by the time constraint of 5 minutes, with events in various locations around the maze.

Upon meeting with our customer and conducting an interview, we began to elicit additional functionality required by the system. The system shall enable the customer to use a keyboard and mouse to traverse the maze, and interact with any obstacles that they may encounter. To ensure that the game is easy to play, the maze layout will be fixed, with certain features being recognisable to customers as from the University of York, to help guide the customer. The difficulty will also be fixed, ensuring that anyone can play and complete the maze. Due to the casual nature of the game, the customer will be able to pause the game at any time, and return to the game when they like. Finally, the system shall produce a clear message to show when the game has finished, either from the customer completing the maze, running out of time, or not being able to overcome an obstacle. This will give the system clear start and end points.

Upon completion of the system, the customer will have a portable game that can be played on any platform, which allows them to traverse a maze based on the University of York and escape within 5 minutes. The system will contain obstacles which can both help or hinder the player, adding to gameplay, but maintaining a reasonably low difficulty level to ensure that the game is accessible to all.

Requirement structure

Using the single statement of need, we can form our basic requirements and detail how they can be met. To achieve this, we have split the requirements into two main sections: user requirements and system requirements.

User requirements are those from a customer's perspective, and describes how the system should interact with the users, and how the design brief and customer's wishes should be implemented. These were directly elicited from the customer and the system description in the design brief.

System requirements are used to implement the User requirements, and offer a more detailed and technical description of how the system will be implemented in line with the customer's requirements. There may be many system requirements for a single user requirement, due to the increased complexity.

User requirements

ID	Description	Priority
UR_AUDIENCE	The game shall be directed at a casual audience	Shall
UR_THEME	The game shall be university-themed & family-friendly	Shall
UR_SOUNDTRACK	The game could have a soundtrack	May
UR_SCORE	Players shall be able to score points and see their score	Shall
UR_EVENTS	The game shall have events with player interactions	Shall
UR_POSEVENT	The game shall have a positive event type: a minimum of one visible event which benefit the player if they pass through them	Shall
UR_NEGEVENT	The game shall have a negative event type: a minimum of one visible event which hinder the player from progression	Shall
UR_HIDEVENT	The game shall have a hidden event type: at least one event which is invisible until triggered by the player. These events also have interactions unique to the other two	Shall
UR_PAUSE	The user shall be able to pause the game at any time	Shall
UR_ITEMS	The game could allow the player to pick up items players can use to solve events	May
UR_END	The users shall be clearly informed when the game has ended, either with a win or loss	Shall

UR_PLATFORM	The game shall be playable on all platforms	Shall
UR_INPUT	The game shall be playable using keyboard and mouse	Shall
UR_HARDWARE	The game shall be playable on typical computer hardware	Shall

System Requirements

ID	Description	User Requirements
FR_SCORE	The game shall have a scoring system which updates after player events, and decreases as the player spends more time in the maze. This score shall be shown to players after the maze run ends	UR_SCORE
FR_HUD	The game should display the player's score and remaining time, both being updated in real time during gameplay	UR_SCORE
FR_UNLOCK	The game shall allow the player to remove an obstacle if they select the key in their inventory	UR_ITEMS
FR_EVENTS	The game shall have an event system with three distinct types	UR_EVENTS UR_POSEVENT UR_NEGEVENT UR_HIDEVENT
FR_POSEVENT	The game shall have at least three visible events which buff the player	UR_EVENTS UR_POSEVENT
FR_NEGEVENT	The game shall have at least five visible events which block player progression, or force them to take another route	UR_EVENTS UR_NEGEVENT
FR_HIDEVENT	The game shall have at least three hidden events with unique interactions	UR_EVENTS UR_HIDEVENT

FR_PAUSE	The game shall have a pause screen, triggered by pressing the escape button	UR_PAUSE
FR_ENDSCREEN	The game shall show a success screen if the player escapes, or game over if they fail	UR_END
FR_ITEMS	The game could have an item system and an inventory to store those	UR_ITEMS
FR_HOTBAR	The game could allow the player to select the item they want to use	UR_ITEMS

ID	Description	User Requirements	Fit Criteria
NFR_AUDIENCE	The game shall be playable by those with little experience in gaming	UR_AUDIENCE	New players shall be able to escape the maze within five tries
NFR_THEME	The game shall have a university theme	UR_THEME	The game's assets shall include various items, characters and landmarks which are clearly linked to the university theme
NFR_INPUT	The game shall be controllable with a keyboard and mouse	UR_INPUT UR_AUDIENCE	All of the intended gameplay features shall work with a keyboard and mouse
NFR_HARDWARE	The game shall run smoothly on modern laptops and computers; players are not required to have specialised hardware	UR_HARDWARE UR_INPUT UR_AUDIENCE	Players will have no problem using budget laptops/desktops released within the last five years
NFR_OS	The game shall be compatible with all mainstream OS'	UR_PLATFORM	All game features shall work the same way across Linux, MacOS and Windows 10 & 11

NFR_SOUNDTRACK	The game could have a soundtrack	UR_SOUNDTRACK	Players could be able to hear music suited to most player sequences (e.g. menus, time almost running out)
----------------	----------------------------------	---------------	---