



PROJECT PLAN

GROUP 3 – PROTOTYPE 3 – SUSAN SAUSAGE ROLL



Group 3
WITS | WSOA3000

1. Project Overview

This project aims to clone the game *Stephen Sausage Roll* to gain a deeper understanding of the development process of the game. The team is given a week to complete the task.

The game created, *Susan Sausage Roll*, includes base level mechanical aspects of the original. This includes character movement, rolling and cooking sausages and selecting levels. This is to develop the mechanical understanding of the game and the division of the development process of the mechanics. The inconsistent yet acceptable mix of art styles from the original were duplicated to understand how to replicate the aesthetic.

Levels were replicated as the difficulty of designing the levels could not be achieved within the time frame. However, in cloning the levels, an understanding of the layout and reasoning for puzzle solving was developed. As well as a method to easily create the levels.

Daily testing, check-ups and development was applied to the project. The deliverables include a Project Plan, Game Design Document and a Build of the game. Deadlines and schedules are laid out with leeway for mishaps.

2. Technologies Used

Table 1: Technology and Tools used

Name	Use	Specification
	Software	
Unity Version 2019.1.11f1	Development Environment	The cross-platform game engine being used as team members have the most experience using this engine.
Visual Studio	Text Editor	Used to code in C#; currently comes with the Unity game engine.
Krita	Art Tool	Used for 2D art assets
Paint.net	Art Tool	Use of grid system to create the map texture to easily create the layout of the ground in Unity.
Blender	Art Tool	Used for 3D modelling, texturing and animations.
GitHub and SourceTree	Source Control, Backup and Storage	Provides access and source control and several collaboration features such as bug tracking, feature requests, task management, and documentation for ease of use.
WhatsApp	Communication Media	Quick access communication means between individuals to deliver information to each other.
Discord	Communication Media	It enables you to centralize all your notifications into one searchable place where your team can discuss and act on each.
In-Design and MS Word	Documentation	Free (for students) and easy to use online and offline (respectively) document creator and editor. Used to create the Design Document and Project Plan, respectively.
Gmail	Communication Media	Communication means when GitHub fails a single team member (contingency plan to get assets in the development environment).

	Hardware	
DELL Inspiron 15 70000 Gaming Laptop	Work Environment	Basic hardware setup needed for the team to develop the prototype.

3. Group Management

The project manager is to set up the repository (on GitHub) and ensure that every team member is a collaborator (ensuring that everyone has a GitHub account); thereafter, each team member is to clone the repository on their device. Thereafter, the means of communication is setup—a WhatsApp group is made, and a Discord Group Chat is started. This is to ensure communication channels are open and clear. The team must also ensure that they are using the same version of Unity (2019.1.11f1). A to-do list for each day must be laid out immediately. This setup must be done upon forming the group.

The project manager does daily progress check-ups on team members and adjusts the schedule accordingly. An audio list and asset list are developed after considering the styles that the Designer, Audio designer and Artist agree upon. The GitHub repository is managed by the team. Each team member must state when they have pushed an update, on WhatsApp—and thereafter, everyone should pull to avoid Merge Conflicts and out-of-date models.

The Programmer has daily code submissions to complete core mechanics. These submissions are noted on GitHub. The Designer and the Programmer are to collaborate on Level Design aspects due to the methodology used to create the levels (mapping colours to Gameobjects).

The Artist and Sound designer are to communicate with the Designer and vice versa to obtain assessment of art assets and a list of adjustments to be made. The Project Manager must always be aware of any completed task or progress done.

The Project Manager is to pin the overall to-do list on Discord. Each member is to pin their respective to-do list on Discord (such as asset lists and audio lists) and then the Project Manager will confirm if this list is complete or requires adjustments. Each to-do list requires an estimated date of completion. This list is referred to daily.

The team meets up on Monday, Tuesday, Wednesday and Friday for an hour to review progress and set the tasks for the following day. Thursday, Saturday and Sunday meetings are held online (for an hour with specific individuals).

Table 2: Team and Duty division

Team Member	Core Duty	Sub Duties
Dhannya Mathew	Project Manager	3D Modelling, Quality Assurance, Main Menu Programmer
Blake Denham	Programmer	Particle Effects
Nkosinathi Mviko	Artist (3D Modelling and 2D Art)	Animation
Marco Prinsloo	Sound	Quality Assurance
Victoria Bench	Designer	3D Modelling, Quality Assurance

The reasoning behind splitting up the 3D Models is outlined in the Game Design Document.

Table 3: Duty Definitions

Duty	Includes
Project Manager	Creating the Project Plan Creating the Repository Ensuring the team communicates with each other and stays up to date on tasks Scheduling and re-scheduling after progress evaluation Checking up on and managing team members and progress

	Managing repository processes and problems
Programmer	Creating code for the game Creating a technical outline to include in the Design Document Testing the game during the development process Implements Assets
Art (3D Modelling and 2D Art)	Creating 3D models and 2D art assets
Sound	Creating and/or locates audio for the game Creating an audio development outline to include in the Design Document
Designer	Creating the Design Document Creating level/map design Dealing with adjustments and final decision on art assets
Quality Assurance	Testing the build Finding and reporting bugs
Particle Effects	Creating Particle effects for the game
Animation	Animating 3D models and 2D art assets

4. Asset Lists

Particles Effects

- Cooking
- Burning
- Sausages Disappearing

3D Art Assets

- Character
- Fork
- Flowers
- Ground Tiles (7 different types: Grill (on and off), Sand, 3 Grass and Dirt)
- Sausage pieces (4 separate pieces)

2D Art Assets

- Cooked Sausage for when hovering over buttons
- Burnt Sausage for when buttons are clicked

Audio Assets

Royalty Free sounds:

- Sizzle
- Character Turn Sound
- Shove or push with the pitchfork
- Button hover and button click
- Sausage in water plop sound
- Walking sound
- Ouch (for when the player walks onto the fire)

Composed Audio:

- Background-Game
- Background-Main Menu
- Lose/Sad loss
- Win/ Puzzle Complete SFX

5. Schedule

The team is given a week to complete. Tasks were designated for each day. The colour coding is according to the Team Member who completed the task (in accordance to the key in Table 2). No colour means that the team completed the task together. Red Highlights are late events and Green Highlights are early events.

Table 4: Estimated Schedule

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Check-In with Team Members (during meeting time)	Check-In with Team Members (during meeting time)	Check-In with Team Members (during meeting time)	Check-In with Team Members (via WhatsApp)	Check-In with Team Members (during meeting time)	Check-In with Team Members (via WhatsApp)	Check-In with Team Members (via WhatsApp)
Update Discord Pins	Update Discord Pins	Sausage Mechanics (rolling sausages, level selection and moving sausages)	Sausage Mechanics (undo movement, cooking, burning)	Update Discord Pins	Update Discord Pins	Complete Design Document
Choose Game to clone	Core Mechanics (character movement and undo, map generation)	Design and implement map	Model and texture all tile assets	Core Mechanics (winning levels, end game if sausage burnt, escape button menu)	Particle Effects	Edit and implement audio and remaining art assets
Assign Duties	Asset List discussion	Model and texture fork	Find royalty free audio assets	Test and modify audio tracks tested	Complete Project Plan	Quality Assurance Testing
Make a list of assets required and due dates	Complete Concept Art	Model and texture sausage	Model, texture and animate character and flowers	Implement tiles	Complete composed audio	Debugging according to quality assurance testing
Setup GitHub Repository, Discord and WhatsApp				Create art for main menu buttons (2D)	Add main menu System and do Quality Assurance Testing	Create Water Shader

Table 5: Actual Schedule

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Check-In with Team Members (during meeting time)	Check-In with Team Members (during meeting time)	Check-In with Team Members (during meeting time)	Check-In with Team Members (via WhatsApp)	Check-In with Team Members (during meeting time)	Check-In with Team Members (via WhatsApp)	Check-In with Team Members (via WhatsApp)
Update Discord Pins	Update Discord Pins	Update Discord Pins	Sausage Mechanics (undo movement)	Update Discord Pins	Update Discord Pins	Complete Design Document
Choose Game to clone	Core Mechanics (character movement and undo, map generation)	Sausage Mechanics (rolling sausages, level selection and moving sausages)	Model and texture all tile assets	Sausage Mechanics (cooking, burning, winning levels)	Model, texture and animate character. Create main menu button art (2D).	Implement audio and remaining art assets
Assign Duties	Rescheduling according to design changes (Art Asset Division)	Design and implement map	Royalty free audio assets found	Add main menu system	Particle Effects and end game conditions	Water shader and fixing undo
Make the list of assets required and due dates	Model and texture Sausage	Model and texture fork		Test and modify audio tracks tested	Project Plan completed	Escape button options
Concept Art	Map Designing			Rescheduling according to new art asset deadlines	Composed Audio completed	Quality Assurance Testing
Setup GitHub Repository, Discord and WhatsApp chat				Implement tiles	Quality Assurance	Edit and evaluate audio
					Remodel and texture flowers	Debugging according to quality assurance testing

Table 6: Gantt Chart for overall schedule

Activity	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Design and Planning							
Brainstorming game							
Task creation and delegation							
Design adjustments							
Development							
Art Asset development							
Audio development							
Mechanical Development							
Menus							
Integrating assets							
Tweaks and fixes							
Testing and Quality Assurance							
Play test							
Build test							

6. Difficulties Occurred

There was a time crunch for art assets. This was due to the inexperience with Blender for animation. Deadlines for the art assets were pushed by 3 days due to this.

A team member did not inform the team that they had not set up the repository until Thursday; by then it was deemed pointless to set it up as emailing the assets would be far less time consuming than explaining the setup and risking unadvised actions to be done on SourceTree.

The lack of access to a stable Internet connection forced one team member to stay at university throughout the week.

7. Testing and Quality Assurance

Upon testing before the build phase, the mechanics and aesthetics are tested in Unity by the programmer and designer. Changes are implemented after each test. Tests are done at the end of each day; after each mechanical implementation for the day.

After building the project, the game is tested for bugs and gauging the game's comparison to the original. Upon locating a bug, the Assurance Tester notifies the Programmer via WhatsApp stating the bug and what triggered it. If it is difficult to explain via text, the individuals discuss it on Discord.

8. Risk and Conflict Management

During the week, risks are anticipated to occur. Plans were made to bypass these risks.

Table 7: Risks and Contingencies

Possible Risks	Contingencies plans in place
Limited Internet Access	A meeting is held every day except for Thursday, Saturday and Sunday. During these meetings, deliverables are scheduled to be due to pass them via portable devices to the responsible individuals. Individuals who are most likely to be without internet access have earlier deliverable dates to avoid stress over the weekend as contact is limited then.
GitHub Errors	Errors are made to be prevented by stating when a push is made via communication means setup. However, if an error does occur, the project manager keeps an up to date version of the project (constantly pulls from the repository) and can recreate the latest version of the repository at any time.

	<p>The error that the team member has must be shown to the group to see if it can be resolved. If so, the team communicates via Discord to solve the error. Otherwise, the individual emails the assets implemented and re-clones the repo to save time.</p>
Miscommunication	<p>Any errors via WhatsApp are to be clarified via a Discord voice call and or in person during meetings. To ensure that the team knows their respective deliverables and duties, each team member themselves writes their to-do list, after being debriefed on it on the first day, and posts it on Discord to be pinned.</p> <p>The project manager then evaluates the list and notifies individuals if things are missing and how they can fit back in.</p>
Requested Changes	<p>The schedules are made with a (max) two-day leeway. Individuals are allowed 1-2-day delay in deliverables (depending on the deliverable).</p>
Over scoping and/ or Insufficient training/ knowledge	<p>Although the game has been chosen according to a thorough evaluation of the team's skills; it is still very possible to over scope the game.</p> <p>Online tutorials for insufficient training are to be found on the first day and is scheduled in to 'train' the individual.</p> <p>The team is versatile enough to take on tasks for other members if necessary. Team members are on standby for such situations.</p>