Kitchen Chaos

Software Requirements Specification

12/10/2023

Mohamed Khaled El-Sayed

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this Software Requirement Specification (SRS) document is to provide a detailed description of the requirements for the development of the “Kitchen Chaos” game, this document serves as a foundation for the design and development to ensure a clear understanding of the desired features and functionalities for those who involved in the game development field.

1.2 Scope

“Kitchen Chaos” is a user-friendly game designed to provide entertainment to players of various ages. The game aims to be across different platforms, including but not limited to desktop and mobile devices. The scope of this document is to discuss the specific features, functionalities, and constraints associated with the development of “Kitchen Chaos.”

1.3 Definitions, Acronyms, and Abbreviations

- **Kitchen Counters:** Interactive elements in the game environment where players perform various recipe-related actions.
- **Controller:** Input device used for playing games, providing a tangible interface for user interaction.
- **Stove Burn Warning:** Visual indicator signaling that the meat on the stove counter is close to burning.
- **Game Timer:** Countdown mechanism indicating the remaining time for a game session.
- **Pause Menu:** Interface accessed during gameplay pause, providing options to resume, access the options menu, or return to the main menu.
- **FPS:** Frames Per Second
- **UI:** User Interface

2. Specific Requirements

2.1 Functional Requirements

2.1.1 User Input and Controls

2.1.1.1 The game must support gameplay using external controllers.
2.1.1.2 The game must support gameplay using the keyboard.
2.1.1.3 Players should have the ability to rebind keys for both the controller and keyboard.

2.1.2 Recipe Dynamics

2.1.2.1 The game shall generate recipes dynamically.
2.1.2.2 Recipes must vary in complexity and ingredients.
2.1.2.3 The spawn rate of recipes should adapt to the player's performance and progress.

2.1.3 Gameplay Mechanics

2.1.3.1 The player must be able to interact with kitchen counters to prepare recipes.
2.1.3.2 Successful recipe completion must be based on accurate and timely player actions.
2.1.3.3 The game should verify whether the recipe delivered by the player is a valid recipe or not.

2.1.4 Audio Feedback

2.1.4.1 Sound effects should accompany player actions, including movement, cutting vegetables, and recipe delivery, with variations for different interactions.
2.1.4.2 Sound volume levels should be adjustable in the options menu to cater to individual player preferences.

2.1.5 User Interface (UI)

2.1.5.1 The game must have a user-friendly interface.
2.1.5.2 The UI in gameplay should display relevant information, such as timer, and all waiting recipes.
2.1.5.3 The UI must be adaptable to various screen resolutions.

2.1.6 Pause and Resume

2.1.6.1 Players should be able to pause the game at any point during gameplay.
2.1.6.2 Resuming the game should return the player to the exact state before pausing.

2.1.8 Player Animations

2.1.8.1 The player character must have some good animations for his different states.
2.1.8.2 Animations should be context-sensitive, adapting to the player's actions and the state of the game.

2.1.9 Various Counter Types

2.1.9.1 Different types of kitchen counters must be available, each dedicated to a specific recipe component or task.
2.1.9.2 Counter types include chopping boards for vegetables, stoves for cooking, and mixing stations for combining ingredients.
2.1.9.3 Each counter should have a unique visual characteristic to distinguish its purpose.

2.1.10 Stove Burn Warning

2.1.10.1 A visual indicator, such as a bar, must appear when the meat on the stove counter is about to burn.
2.1.10.2 The warning system should be accompanied by an audio cue to enhance player awareness.

2.1.11 Game Timer

2.1.11.1 The game must include a timer that counts down during gameplay, and when the timer reaches zero, the game ends.
2.1.11.2 The game must include a timer that counts down from 3 to 0 before starting the game.
2.1.11.3 The timer should be visible on the UI, ensuring players can monitor their remaining time.
2.1.11.4 The end-of-game event should include a summary screen displaying how many recipes the player delivered.

2.2 Non-Functional Requirements

2.2.1 Performance

2.2.1.1 The game must maintain a smooth frame rate of at least 30 frames per second according to recommended system specifications.
2.2.1.2 Loading times for different game elements (e.g., levels, textures) should not exceed 3 seconds.

2.2.2 Reliability

2.2.2.1 The game must be stable and not crash during normal gameplay.

2.2.3 Maintainability

2.2.3.1 Code should be well-documented, following industry-standard coding conventions.

2.2.4 Portability

2.2.4.1 The game should be easily portable to different platforms without extensive modifications.