# 1. Game Overview

## 1.1 Game Concept

*Space Survivor* is a top-down, action survival game where players take on the role of an astronaut stranded on an uncharted, hostile planet. After a catastrophic crash landing, the player must navigate through treacherous terrains, defend against relentless waves of extraterrestrial monsters, and ultimately find a way to escape the planet. As the game progresses, players encounter increasingly formidable enemies, including a formidable boss at wave 10, adding layers of challenge and strategy.

## 1.2 Genre

- **Action**
- **Survival**
- **Top-Down Shooter**

## 1.3 Target Audience

- **Age Range:** 13+
- **Demographics:** Gamers who enjoy action-packed survival experiences, procedural challenges, and sci-fi themes.
- **Platforms:** PC (Windows, macOS, Linux), potential for future expansion to consoles.

## 1.4 Platform

- **Primary:** Desktop (Windows, macOS, Linux)
- **Engine:** libGDX – A cross-platform Java game development framework.

---

# 2. Story and Lore

## 2.1 Narrative Background

In the distant future, humanity has mastered space travel, venturing into the far reaches of the galaxy. Amidst this era of exploration, a lone astronaut, Captain Alex Mercer, embarks on a mission to explore uncharted planets for potential habitation and resources.

## 2.2 Crash Landing

During a routine expedition, Captain Mercer’s spacecraft, the *Starfire*, encounters a massive cosmic storm near an unknown planet. Despite the crew’s efforts to navigate through the tempest, the ship sustains severe damage, forcing an emergency crash landing on the planet's surface.

### 2.3 The Hostile Planet

The planet, designated as **Xenon-9**, is a harsh and unforgiving environment teeming with bizarre and aggressive alien lifeforms. The landscape is a mix of desolate deserts, dense jungles, and eerie, abandoned ruins hinting at a lost civilization.

### 2.4 Survival and Escape

Stranded with limited resources, Captain Mercer must survive against relentless waves of monsters that emerge from the planet's depths each night. As days pass, Mercer discovers remnants of ancient technology and mysterious artifacts that may hold the key to repairing the ship or signaling for rescue.

### 2.5 The Ultimate Challenge

At wave 10, a colossal and highly intelligent creature, known as the **Golem King**, emerges as the planet's guardian. Defeating the Golem King is crucial for Mercer's survival and the hope of escape from Xenon-9.

---

## 3. Gameplay Mechanics

### 3.1 Player Mechanics

- **Movement:** The player can move in four directions (up, down, left, right) using keyboard inputs.
- **Combat:** The player can shoot projectiles to defend against enemies. Weapon types may vary, each with unique attributes.
- **Health Management:** The player has a health bar that decreases upon taking damage. Health can be restored using healing buffs.
- **Experience and Leveling:** Defeating enemies grants experience points, leading to level-ups that can enhance player abilities.
- **Buffs and Power-ups:** Players can collect various buffs to improve movement speed, fire rate, and other attributes.

### 3.2 Enemy Types

- **Standard Monsters:**
    - **Trouille:** Fast-moving creatures that swarm the player.
    - **Xela:** Heavier, more resilient enemies that deal significant damage.
- **Boss Enemy:**
    - **Golem King:** A massive, highly durable creature with advanced attack patterns and abilities, appearing at wave 5.

### 3.3 Boss Mechanics

- **Appearance:** The Golem King appears at wave 10, marking a significant increase in game difficulty.
- **Abilities:**
    - **Projectile Attacks:** Shoots powerful projectiles at the player.
    - **Area Attacks:** Executes area-of-effect attacks that damage the player within a certain radius.
    - **Defensive Maneuvers:** Uses shields or temporary invincibility phases to avoid damage.
- **Health:** Significantly higher health than standard enemies, requiring sustained effort to defeat.
- **Behavior:** Intelligent movement patterns, targeting player weaknesses, and adapting attacks based on player actions.

### 3.4 Progression System

- **Wave-Based Survival:** Players face waves of enemies that increase in number and difficulty with each subsequent wave.
- **Wave Triggers:**
    - **Standard Waves:** Spawn based on the wave number, scaling the number of enemies.
    - **Boss Wave:** At wave 10, the Golem King is spawned, replacing standard enemies.
- **Rewards:** Defeating waves grants experience points, score increments, and potential buffs or resources.

### 3.5 Health and Buffs

- **Health Bar:** Represents the player's current health. Decreases upon taking damage from enemies or environmental hazards.
- **Healing Buffs:** Restore a portion of the player's health when collected.
- **Move Speed Buffs:** Temporarily increase the player's movement speed.
- **Fire Speed Buffs:** Enhance the player's weapon fire rate for a limited duration.

### 3.6 Collision Detection and Handling

- **Entity Interactions:** Collision detection ensures interactions between the player, enemies, projectiles, and environmental obstacles.
- **Map Boundaries:** Prevents entities from moving outside the playable area.
- **Hitboxes:** Each entity has a hitbox that determines collision points, essential for combat and movement.

---

## 4. Art and Style

### 4.1 Visual Style

*Space Survivor* adopts a **pixel art** aesthetic combined with modern lighting and particle effects. The game features vibrant colors to distinguish different entities and environments, enhancing visual clarity and appeal.

### 4.2 Character Design

- **Player (Captain Mercer):** A detailed pixel art astronaut with customizable outfits and gear that reflect progress and upgrades.
- **Monsters:**
    - **Trouille:** Agile and sleek designs with glowing eyes, indicating their predatory nature.
    - **Xela:** Bulkier forms with armored exteriors, showcasing their resilience.
    - **Golem King:** A towering, menacing figure with intricate patterns and animated parts, emphasizing its boss status.

### 4.3 Environment Design

- **Planetary Landscapes:** Diverse terrains including barren deserts and alien constructions.

### 4.4 Animation

- **Smooth Animations:** Fluid movement animations for some entities to ensure responsive gameplay.
- **Boss Animations:** Complex animation sequences for the Golem King.

### 4.5 Sound and Music

- **Ambient Sounds:** Atmospheric sounds that enhance the sense of isolation and danger on Xenon-9.

---

## 5. Technical Requirements

### 5.1 Engine and Tools

- **Game Engine:** libGDX – Chosen for its cross-platform capabilities and robust feature set.
- **Programming Language:** Java
- **Graphics Tools:** Aseprite or Photoshop for pixel art creation.
- **Audio Tools:** Audacity for sound editing and creation.
- **Version Control:** Git with GitHub for source code management.

### 5.2 Technical Features

- **Cross-Platform Support:** Ensures the game runs seamlessly on Windows, macOS, and Linux.
- **Efficient Rendering:** Optimized rendering pipelines for smooth performance, even with multiple entities on screen.
- **Collision Detection:** Precise hitbox calculations to ensure accurate interactions between entities.
- **Resource Management:** Efficient loading and disposal of textures, sounds, and other assets to prevent memory leaks.
- **Scalable Difficulty:** Wave progression system that adjusts enemy numbers and types based on the current wave.

### 5.3 Performance Targets

- **Frame Rate:** Maintain a stable 60 FPS on average hardware configurations.
- **Load Times:** Minimize loading times between waves and game screens through optimized asset management.
- **Memory Usage:** Keep memory consumption within reasonable limits to prevent crashes or slowdowns.

---

## 6. User Interface (UI) and User Experience (UX)

### 6.1 HUD Elements

- **Health Bar:** Displays the player's current and maximum health, positioned prominently on the screen.
- **Score Counter:** Shows the player's current score, updating in real-time as enemies are defeated.
- **Wave Indicator:** Notifies players of the current wave number and upcoming boss wave.

### 6.2 Menus and Navigation

- **Main Menu:**
  - **Start Game**
  - **Options**
  - **Exit**
- **Pause Menu:**
  - **Resume**
  - **Restart**
  - **Main Menu**
- **Game Over Screen:**
  - **Retry**
  - **Main Menu**
  - **View Score**
- **Options Menu:**

    - **Audio Settings**
    - **Graphics Settings**
    - **Controls Configuration**

---

# 7. Controls

## 7.1 Keyboard Controls

- **Movement:**
    - **W / Up Arrow:** Move Up
    - **A / Left Arrow:** Move Left
    - **S / Down Arrow:** Move Down
    - **D / Right Arrow:** Move Right
- **UI Interaction:**
    - **Escape:** Pause Game
    - **H:** Toggle Hitbox Visibility

---

# 8. Additional Features

## 8.1 Power-Ups and Upgrades

- **Weapon Upgrades:** Players can find or purchase upgrades to enhance weapon damage, fire rate, and special abilities.
- **Skill Trees:** Unlockable skills that provide passive bonuses or active abilities, such as temporary invincibility or area-of-effect attacks.
- **Environmental Interactions:** Utilize the environment to gain advantages, such as activating traps or using cover to avoid enemy attacks.

---

# Conclusion

*Space Survivor* aims to deliver a compelling survival experience set in a richly crafted sci-fi universe. By combining engaging gameplay mechanics, a gripping narrative, and a distinctive art style, the game seeks to captivate players and provide a challenging yet rewarding journey of survival against the odds. With meticulous planning and structured development phases, *Space Survivor* is poised to become a standout title in the action-survival genre.

---

# Appendices

### Appendix A: Asset List

- **Sprites and Textures:**
  - Player Character Sprite Sheets
  - Enemy Sprite Sheets (Trouille, Xela)
  - Boss Sprite Sheet (Golem King)
  - Environmental Textures (Deserts, Jungles, Ruins)
- **Audio Assets:**
  - Background Music Tracks
  - Sound Effects (Shooting, Explosions, Enemy Attacks)
- **UI Assets:**
  - HUD Elements (Health Bar, Score Counter)
  - Menu Screens and Buttons
- **Miscellaneous:**
  - Particle Effects (Explosions, Buff Effects)

### Appendix B: Technical Specifications

- **Programming Language:** Java
- **Framework:** libGDX
- **Version Control:** Git (Hosted on GitHub)
- **Graphics Software:** Aseprite, Photoshop
- **Audio Software:** Audacity
- **Project Management Tools:** Trello, Jira (optional)

---

# Authoritative Notes

This Game Design Document serves as a comprehensive guide for the development team of *Space Survivor*. It outlines the game's vision, mechanics, and technical requirements to ensure a cohesive and efficient development process. Regular updates and revisions to this document are encouraged to reflect changes and improvements throughout the project's lifecycle.