

Sedohr - Technical Design Document  
Team Murfed Toast

1. Introduction

1.1 Overview

The game will be built on the XNA 2.0 framework in C#. Current target platform is the PC. Future release/port on the XBOX360 console will be considered in a later date. The current system requirements will include a video card that supports DirectX 9 with at least 128 megabytes of video ram and a minimum of 1 gigabyte of system ram. It is understood that the game will not utilize all the resources on the system. However, the target machine will be one of the systems in the ET lab. The systems in the ET lab the must have Xbox 360 controller drivers installed by an administrator.

1.2 Technical Features

- Deferred shading via normal mapped sprites
- Displacement mapping
- (possible) Bloom effect
- Parallax scrolling
- Particle systems
- Physics engine

1.3 Technologies

- XNA 2.0 (includes the .NET framework)
- Farseer 2D Physics engine
- XACT
- SVN - Subversion version control

2. Project Plan

1.1 Schedule - 12/21/07 (Milestone 1)

Outline of our initial schedule:

Week 1:

- Initial thoughts of game design and tools consideration.
- Formation of team
- Brainstorm game ideas

Week 2:

- Brainstorm game ideas
- Finalize teams
- Finalize tools
- Role assignment
- Design architecture

Week 3:

- Research & Development - Prototyping
- Finalize game design
- Engine development
- Low level design
- Concept art and asset creation
- Checkpoint 1: Deliverable

Week 4:

- End of engine cycle - Engine completion

- Engine - Post mortem
  - Paper prototype
  - Begin game cycle - Game development
  - (ongoing) Graphics asset creation and development
- Week 5:
- (ongoing) Asset creation and development
  - (ongoing) Content development
- Week 6:
- (ongoing) Asset creation and development
  - (ongoing) Content development
  - Checkpoint 2: Game prototype
- Week 7:
- (ongoing) Asset creation and development
  - (ongoing) Content development
  - Usability testing
- Week 8:
- Game completion
  - Usability testing
  - (ongoing) Tweaking & testing
- Week 9:
- (ongoing) Tweaking & testing
- Week 10:
- (ongoing) Tweaking & testing
  - End of game cycle - Release (submission?)
- Week 11:
- Submission
  - Game - Postmortem
- \* (12/21) Engine almost complete*

## 1.2 Schedule - 1/19/08 (Milestone 2)

- Week 1:
- Initial thoughts of game design and tools consideration: **Done**
  - Formation of team: **Done**
  - Brainstorm game ideas: **Done**
- Week 2:
- Brainstorm game ideas: **Done**
  - Finalize teams: **Done**
  - Finalize tools: **Done**
  - Role assignment: **Done**
  - Design architecture: **Done**
- Week 3:
- Research & Development - Prototyping: **Done**
  - Finalize game design: **Ongoing**
  - Engine development: **Ongoing**
  - Low level design: **Ongoing**
  - Concept art and asset creation: **Done**
  - Checkpoint 1: Deliverable: **Done**
- Week 4:
- Engine - Postmortem: **N/A**

- Paper prototype: **N/A**
  - Begin game cycle - Game development: **Done**
  - (ongoing) Graphics asset creation and development: **Ongoing**
- Week 5:
- (ongoing) Asset creation and development: **Ongoing**
  - (ongoing) Content development: **Ongoing**
  - Checkpoint 2: Deliverable: **Done (Moved from Week 6)**
- Week 6:
- **End of engine cycle - Engine completion (Moved from Week 4)**
  - **Game prototype complete**
  - (ongoing) Asset creation and development
  - (ongoing) Content development
- Week 7:
- (ongoing) Asset creation and development
  - (ongoing) Content development
  - Usability testing
- Week 8:
- Checkpoint 3: Deliverable
  - Asset creation and development completion
  - Game completion
  - Usability testing
  - (ongoing) Tweaking & testing
- Week 9:
- (ongoing) Tweaking & testing
- Week 10:
- (ongoing) Tweaking & testing
  - End of game cycle - Release (submission?)
- Week 11:
- Final Submission: Deliverable
  - Game - Postmortem

### 1.3 Schedule - 2/10/08 (Milestone 3)

- Week 1:
- Initial thoughts of game design and tools consideration: **Done**
  - Formation of team: **Done**
  - Brainstorm game ideas: **Done**
- Week 2:
- Brainstorm game ideas: **Done**
  - Finalize teams: **Done**
  - Finalize tools: **Done**
  - Role assignment: **Done**
  - Design architecture: **Done**
- Week 3:
- Research & Development - Prototyping: **Done**
  - Finalize game design: **Done**
  - Engine development: **Done**
  - Low level design: **Done**
  - Concept art and asset creation: **Done**

- Checkpoint 1: Deliverable: **Done**
- Week 4:
- Engine - Postmortem: **N/A**
  - Paper prototype: **N/A**
  - Begin game cycle - Game development: **Done**
  - (ongoing) Graphics asset creation and development: **Ongoing**
- Week 5:
- (ongoing) Asset creation and development: **Ongoing**
  - (ongoing) Content development: **Ongoing**
  - Checkpoint 2: Deliverable: **Done (Moved from Week 6)**
- Week 6:
- **End of engine cycle: Done**
  - **Game prototype complete: Done**
  - (ongoing) Asset creation and development: **Ongoing**
  - (ongoing) Content development: **Ongoing**
- Week 7:
- (ongoing) Asset creation and development : **Ongoing**
  - (ongoing) Content development: **Ongoing**
  - Usability testing: **Ongoing**
- Week 8:
- Checkpoint 3: Deliverable: **Done**
  - Usability testing: **Ongoing**
  - (ongoing) Tweaking & testing: **Ongoing**
- Week 9:
- Game completion
  - Asset creation and development completion
  - (ongoing) Tweaking & testing
- Week 10:
- (ongoing) Tweaking & testing
  - End of game cycle - Release (submission?)
- Week 11:
- Final Submission: Deliverable
  - Game - Postmortem

### 3. Project Backlog

#### 3.1 High Level Requirement(s)

- Graphics core/engine - rendering, animation, particle engine, post processors, etc
- Scene manager - (part of graphics) basic scene management in 2D game; takes care of the parallax scrolling, culling and entity depth
- XNA content pipeline processor for custom content
- Engine integration
- Scripting - engine wrapping
- Character mechanics - (possible) managed by a state machine
- Weapon system - weapon behavior and properties
- Joint system - destructible environments
- Black/White hole

## 4. Architecture

### 4.1 UML

Please refer to the attached files: *engine\_design.jpeg*, *GameDesign.vsd*

## 5. Development Cycle

With the given the time constraint, the project will be split into two development cycles: engine and game. The game cycle will have mini-cycles that yields quick prototypes which will give us immediate feedback on whether or not the game will be fun. This will mitigate the risk of having a boring game in the end.

### 5.1 Engine

- Cycle duration: 4 weeks
- Process: Waterfall
- Tasks (engine development) will consists of basic prototyping of engine subsystems and engine integration to the current architecture.
  - Graphics - XNA DirectX 9
  - Physics - Farseer 2D
  - Audio - XACT
  - Input
  - Scripting

\* *Tasks have been split up by subsystems.*

### 5.2 Game

- Cycle duration: 6 weeks
- Process: Agile
- Tasks (content development)
  - Player mechanics
  - Weapon system
  - Joint system (destructible environments)
  - Black/White hole
  - Usability testing
  - Testing & Tweaking

Lastly, postmortems will be held at the end of each cycle to review the development that's been done, decisions made, design, and the tasks ahead.

## 6 Revision

### 6.1 Current Revision

- revision 1.2.0 (2.8.2008) - Updated schedule for milestone 3.

### 6.2 Revision History

- revision 1.1.0 (1.18.2008) - Updated schedule for milestone 2.
- revision 1.0.0 (12.21.2007) - Initial project vision.