ZED

Software Requirements Specification

Open Game Developers

[Version: 0.0.0.0] [Date: 01|10|2012]

ZED

Change Log

Version	Author	Changes
0.0.0.0	Rico	Initial revision Created document structure

ZED

Table of Contents

ZED

Preface

This document will describe the ZED engine in terms of what the engine will be implementing in terms of requirements.

Early class names shall be described here.

API-specific information will not be described.

ZED

Introduction

Purpose

ZED is intended to provide a variety of components to aid in game development. Components which are supported by ZED are as follows:

- 2D and 3D Mathematics
- 2D and 3D Graphics Rendering
- Storage
- System Services
- Physics
- Artificial Intelligence
- Audio
- Input
- Networking
- Scripting

While these functions are implemented in ZED, they are intended to be extended by the developer for their specific development needs.

ZED will also be implemented on a wide variety of hardware. As not all hardware will be capable of accomplishing the same task, the API will need to accommodate these lesser hardware platforms. In addition to hardware limitations, some hardware will be more esoteric and will have to be handled appropriately.

Abstracting away all known APIs from the developer is the ultimate goal of ZED. As the engine is intended to be used by programmers who are not working on lower-level details such as which graphics API to use or how to correctly apply a texture to a mesh.

Scope

As there are a wide variety of components involved in the creation of the ZED engine, the scope of each will expand as needed. In the initial version of ZED, the API will offer a minimal set of functionality for developers to extend as they see fit. ZED will be available for Linux, Windows, Xbox, Pandora, and BlackBerry PlayBook initially.

Definitions, Acronyms, Initialisms, and Abbreviations

Definitions

Acronyms
ZED – The engine which this document is intended to put forth the requirement effort involved

Initialisms
API – Application Programming Interface

Abbreviations
Sockets – Windows or Berkeley Sockets API
WinSock – Windows Sockets API

System Overview

ZED is intended to be as lightweight as possible and as optimised for each platform it supports. Modularity is important for the future expansion of the engine, thus there will be a heavy focus on keeping the system well documented and open for developers.

Following is a narrative on each system provided by ZED:

System Services
Providing memory management, multi-threading, and platform information.

2D and 3D Mathematics
Vectors, Matrices, Quaternions, Rays, Planes, Polygons, Axis-Aligned Bounding Boxes, and Orientated Bounding Boxes.

2D and 3D Graphics Rendering
Direct3D, OpenGL, applying shaders to primitives, Textures, and Fonts.

Storage
Reading/Writing files, and directories.

Input
Keyboard, Mouse, Gamepad, Motion Controller.

Audio
Multi-channel audio.

Networking
Modem, Ethernet, Dead Reckoning, Multicasting.

Artificial Intelligence
Finite State Machines, Behaviour Trees, A* Pathfinding, NavMesh, NavPoints, Steering.

Physics
Rag Dolls, Collision.

Scripting
ZED Scripting Language.

References

ZED

Overall Description

Product Perspective
Product Functions
User Characteristics
Constraints
Assumptions and Dependencies

Specific Requirements

External Interface Requirements
User Interfaces
Hardware Interfaces
Software Interfaces
Communication Interfaces

Functional Requirements

Performance Requirements
Standards
Hardware Limitations

Design Constraints
Availability
Security
Maintainability

Other Requirements