Movie Ticket Ordering System

Version 1.0

Table of Contents

1. Purpose 3

2. Scope 3

3. Definitions 3

3.1 Actor 3

3.2 Administrator 3

3.3 Attribute 3

3.4 Class 3

3.5 Customer 3

3.6 Database System 3

3.7 Electronic Payment 3

3.8 Interface 3

3.9 Manager 3

3.10 Method 3

3.11 Movie 3

3.12 Object 3

3.13 Operation 3

3.14 Package 4

3.15 Parameter 4

3.16 Property 4

3.17 Scenario 4

3.18 Showing 4

3.19 Theatre 4

3.21 Transaction 4

3.22 Use-case 4

# Purpose

Provide the reader with a clear understanding of the terms used in these documents.

# Scope

This glossary is associated with the **<<name>>** Movie *Ticket Ordering System*. Many of terms are based on the Unified Modeling Language and objects and terms in the theater environment.

# Definitions

## Actor

Entity such as a person system or object that exchanges information with the system. They initiate the scenario and receive the results from the scenario.

## Administrator

*See Manager*

## Attribute

A named value that belongs to a class or object.

## Class

Represents a modeling concept within a system.

## Customer

Someone who uses the system to purchase a ticket.

## Database System

Stores the movie information such as times and location. All data is changeable by a manager.

## Electronic Payment

Processes the payment and retrieves validation information for a customer transaction.

## Interface

A set of operations defined and accessed by one or more classes. An interface is an abstraction with no implementation and no attributes.

## Manager

Someone with administrative capabilities in charge of maintaining the system.

## Method

*See Operation*

## Movie

The identification property of a film. It can be played in multiple theatres at multiple times.

## Object

Container with a unique identity that encapsulates some operations and some attributes. It can also be an instance of a class.

## Operation

Actions that an object in a class can do.

## Package

A folder that refers to another diagram.

## Parameter

A value supplied when generating an instance of a class or package.

## Property

*See Attribute*

## Scenario

A description of how a certain actor achieves a tangible result.

## Showing

A specific movie playing at a specific time.

## Theatre

A cinema room or auditorium.

## Transaction

An attempt by a customer to purchase a ticket from the system.

## Use-case

A unit of functionality created by a system or class. A use-case creates tangible results.