Store Model Service Design Document

Date: 9/25/19

Author: Matthew Thomas

Reviewer(s): Steven Hines, Stephen Thompson

# Introduction

*A high-level description of this document, for example, “This document defines the design for the Canonical Model”.*

This design specifies the implementation of a Store Model Service, one component of the Store24X7 Software System. The Store Model Service is responsible for maintaining the state of the sensors (cameras and microphones), appliances (robots, turnstiles, and speakers), and other domain objects (store, aisle, shelf, inventory, product, customer, basket) within the store.

Overview

*Overview of the problem to be solved. What is the problem and why is it being solved? How will the resulting solution provide business value?*

*Consider adding a diagram that explains how this component fits into the overall System with some descriptive text explaining the diagram.*

# Requirements

*This section provides a summary of the requirements for the <Component Name>.*

*Provide your understanding of the requirements, both functional and nonfunctional. Reference the provided Requirements and System Architecture documents. Do not cut and paste from the requirements document.*

*Product Manager and others can read this to understand what requirements your design will support. There is already a requirements doc, so keep this brief and to the point, highlighting the important requirements that the design is addressing. Structure in a way to provide a requirements checklist for your design.*

# Use Cases

*Enumerate the use cases supported by the design,*

*This design supports the following use cases:*

*Include a Use Case Diagram.*

*Include descriptions of each of the actors and use cases.*

The following Use Case diagram describes the use cases supported by the Store Model Service.

## Actors:

The actors of the Store Model Service include Store, blank, and blank.

### Store

Store’s use case here.

## Use Cases:

### Initialize store

A store is initialized.

# Implementation

*This section of the document will describe the implementation details for ...*

*The implementation section should cover the following topics:*

* *What are the classes, and their properties, associations and methods?*
* *What are the important interfaces and how they will be implemented?*
* *How are the requirements addressed?*

# Class Diagram

The following class diagram defines the Store Model Service implementation classes contained within the package “com.cscie97.store.model”.

*CLASS DIAGRAM GOES HERE*

# Class Dictionary

This section specifies the class dictionary for the Store Model Service. The classes are defined within the package “com.cscie97.store.model”.

## *CLASS 1*

*Class 1 description*

***Methods***

|  |  |  |
| --- | --- | --- |
| **Method Name** | **Signature** | **Description** |
|  |  |  |

***Properties***

|  |  |  |
| --- | --- | --- |
| **Property Name** | **Type** | **Description** |
|  |  |  |

***Associations***

|  |  |  |
| --- | --- | --- |
| **Association Name** | **Type** | **Description** |
|  |  |  |

## *CLASS 2*

...

# Implementation Details

*Explain details of the implementation.*

*How do the various parts fit together or interact?*

*How does the design address the requirements? Justify your design decisions and how they address the requirements.*

*Some implementation details may be addressed in the class dictionary, but for things that are not, describe them here.*

*Remember to reference the requirements from the body of the design document to show how your design is addressing the requirements.*

# Exception Handling

*Provide details on your exception handling. What types of exceptions are expected and how are they handled by the design? Describe your exception classes and their properties.*

# Testing

*Provide a testing strategy for testing the component.*

* *Functional*
* *Performance*
* *Regression*
* *Exception Handling*

# Risks

*Document any risks identified during the design process.*

*Are there parts of the design that may not work or need to be implemented with special care or additional testing?*