REVISION HISTORY

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <dd/mm/yy> | <x.x> | <details> | <name> |
| 09.11.2016 | 0.01 | Software Design Description added for Model component. (3.1) | Mehmet Melih Arıcı |
| 9.11.2016 | 0.02 | Software Design Description added for View component. (3.2) | Eren Balatkan |
| 09.11.2016 | 0.03 | Software Design Description added for View component. (3.3) | Arda Mutlu  Berkay Bayram |

**TABLE OF CONTENTS**

**1** **Introduction**

***1.1*** ***References***

1.1.1 Project References

**2** **Software Architecture overview**

**3** **Software design description**

***3.1*** ***Component 1***

3.1.1 Component interfaces

3.1.2 Component design description

3.1.3 Workflows and algorithms

3.1.4 Software requirements mapping

***3.2*** ***Component 2***

3.2.1 Component interfaces

3.2.2 Component design description

3.2.3 Workflows and algorithms

3.2.4 Software requirements mapping

***3.3*** ***Component 3***

3.3.1 Component interfaces

3.3.2 Component design description

3.3.3 Workflows and algorithms

3.3.4 Software requirements mapping

**4** **COTS Identification**

# Introduction

Brief description of the software system and the purpose of the document.

This document describes the design of the XXX software system.

## References

### Project References

|  |  |  |
| --- | --- | --- |
| # | Document Identifier | Document Title |
| [R1] | ID | Add your documents references.  One line per document |

# Software Architecture overview

Describe here the top level software components and their interactions/relationships.

Use UML diagrams.

# Software design description

Describe each top level package/component of your software in the following subsections. The number of sub-sections (3.1, 3.2, … etc.) depends on the number of packages/components in your architecture diagram.

## Component 1

Model

### Component interfaces

Public methods;

GetBuildingID

GetRoomID

GetUserID

GetReservationID

getCapacity

### Component design description

### Workflows and algorithms

### Software requirements mapping

SRS-SQ-RMS-020.0 – Database connection

SRS-SQ-RMS-030.0 - Database connection

SRS-SQ-RMS-040.0 – Room Reservation

SRS-SQ-RMS-050.0 – Room Reservation Cancellation

## Component 2

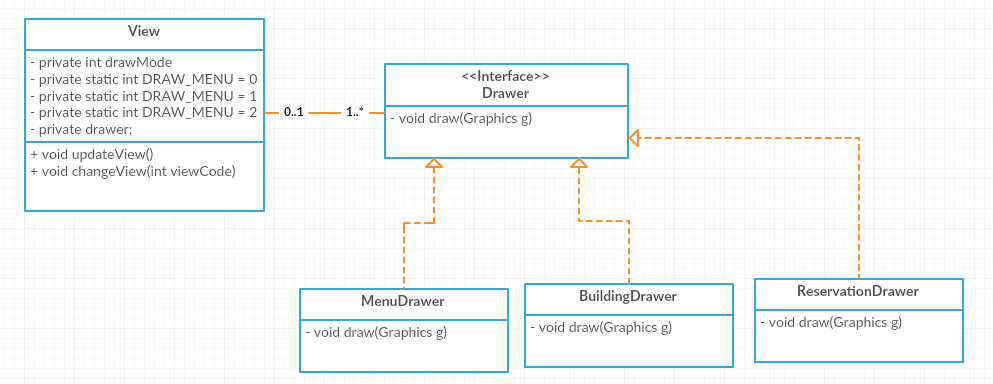
View

### Component interfaces

void updateView()

void changeView(int viewCode)

### Component design description



### Workflows and algorithms

Use sequence diagrams to show the workflows of components/packages/classes inside the component.

**updateView:** Requests fresh data from controller and re-draws the frame according to the new data

**changeView:**  This method is called by the controller to tell the view what it is going to draw ( Menu, Building , etc..)

### Software requirements mapping

SRS-SQ-RMS-040.0 – Room Reservation

SRS-SQ-RMS-050.0 – Room Reservation Cancellation

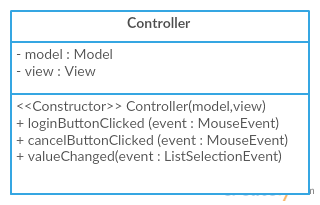
## Component 3

Controller

### Component interfaces

Describe the interfaces of the component and input output data

### Component design description

**

### Workflows and algorithms

### Software requirements mapping

SRS-SQ-RMS-010.0 – Login system

SRS-SQ-RMS-040.0 – Room Reservation

SRS-SQ-RMS-050.0 – Room Reservation Cancellation

# COTS Identification

List external software components/libraries that your system relies on, if there are any.

Example:

COTS (commercial of the shelf) libraries used in XXX are the following:

* foo.jar, version id, download URL, License type,
* bar.jar, version id, download URL, License type,