Wydział Matematyki i Nauk Informacyjnych

Politechnika Warszawska

Title	Project #1: Hostel Management
Subject	Enterprise Applications in .NET Framework
Year	2016-2017
Version	1.1
Author	Mario Juez Gil juezm@student.mini.pw.edu.pl

Revision history

Date	Author	Changes	Revision
19/10/2016	Mario Juez	First version of this document.	1.0
20/10/2016	Mario Juez	- Removed general project introduction covering information about three iterations. - Improved business description with detailed information. - Added and modified user histories to fit use cases definition. - Detailed schedule.	1.1 (current)

Table of contents

Table of figures

1 Project #1: Hostel Management

1.1 Business Description

The first project product will be a desktop application for generic hostel management. We understand hostel management as guest check-ins and bedrooms management by a hostel manager.

Guest check-in will be a form that the manager will have to fill up with guest personal data like ID number, name, surname, birth date, and sex. That form will also contain check-in and check-out dates, and room number fields. Depending of bedroom size, the form will allow to the manager register one or more guests into the same check-in.

For bedroom management, the application will have a view with bedroom listing including filtering options by size, price, bathroom type, bed type and availability. Through that view, manager will be able to open the bedroom editor for creating or modifying a bedroom, remove bedrooms, or open the check-in registration view for one specific bedroom.

In order to track operations and avoid usage of non-authenticated users, hostel manager login will be obligatory to use the application.

1.1.1 Technologies

This project is going to use the following technologies:

1. **Windows Presentation Foundation (WPF)**: User Interface.
2. **.NET Core**: Business Logic.
3. **NoSQL Database MongoDB**: Database.

1.2 Functional Requirements

This part of the document, defines project functional requirements by usage of use case diagrams and user stories.

1.2.1 Use Case Diagram

The following figure shows project use case diagram.

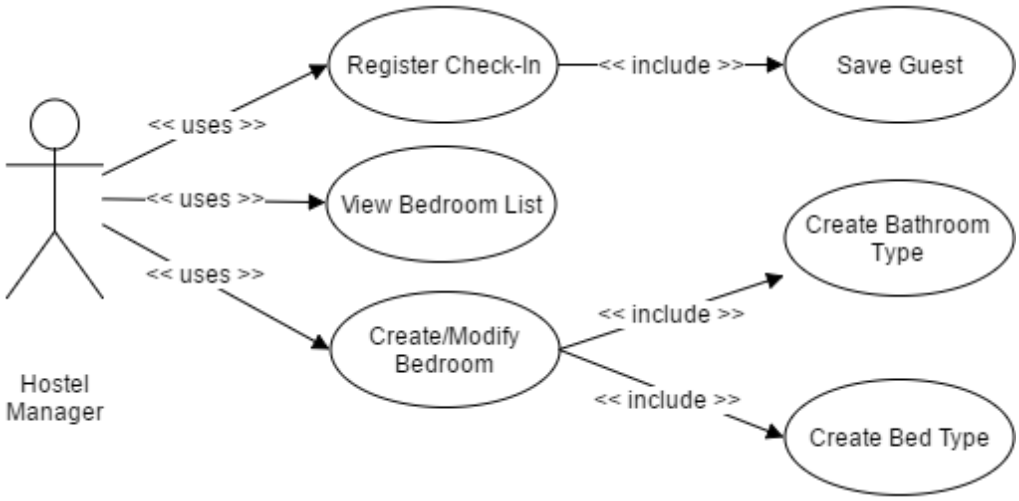

Figure 1: Use Case Diagram.

Actor	Name	Description
Hostel Manager	Register Check-In	The user will enter check-in and check-out date, select the bedroom associated to, and finally the guest information.
	Save Guest	When guest ID is filled, the system will check if that ID was registered before, if so, the form will fill remaining fields automatically. The user is able to modify any field and update it in database by saving the check-in.
	View Bedroom List	The user is able to view the hostel bedroom list, this view provides filtering options to filter the rooms by its features like price, bathroom type… The list will show information about bedroom availability. Inside this view, user will find buttons to create a new room, and modifying or deleting existing ones.
	Create/Modify Bedroom	If bedroom does not exist, user will enter bedroom number, bed number and type, bathroom type, and regular price per night into a blank form. That form will contain bedroom data if the user is modifying it. Bathroom and bed type fields has to have autocomplete feature for those types which already exists.
	Create Bathroom Type	If the introduced bathroom type does not exist in the database, the system has to create it.
	Create Bed Type	If the introduced bed type does not exist in the database, the system has to create it.

Table 1: Functional requirements.

1.2.2 User Stories

1. Check-In Registration:
   As hostel manager, I need an agile method to register guest check-ins, that allows me to enter guest required information as quick as possible.
2. Guest Storage:
   As hostel manager, I need the system to store all hostel guest data for using it if they return in the future, letting me to fill automatically the check-in form by entering guest ID number.
3. Bedroom Listing:
   As hostel manager, I need an easy method to list all bedrooms of the hostel and see its features and availability, in order to offer my clients the bedroom that best fits their needs.
4. Bedroom creation and modification:
   As hostel owner, I have a hostel chain with different hostels placed in different cities. I need flexible solution which allows my hostel managers to configure each specific hostel bedrooms. Besides, bedroom features like price are subject to change along the time, and managers should have the possibility to modify it.
5. Bathroom types:
   As hostel owner, bedrooms in my hostels has many bathroom configurations, like private bathroom + private shower, shared bathroom, private bathroom + shared shower etc.

6. Bed types:
   As hostel owner, bedrooms in my hostels has many beds configurations, like single bed, double bed, two single beds, bunk bed, etc.

1.3 Non-Functional Requirements

Non-functional requirements are the following:

1. Interface:
   a. Desktop application must work properly on screens with a minimum resolution of 1366x768.
2. Portability:
   a. Desktop application must work properly on Microsoft Windows 10 and above.
3. Security:
   a. User must authenticate into the application with valid credentials.

1.4 Schedule

Date Range	Task
14/10/2016 – 21/10/2016	Initial documentation definition.
21/10/2016 – 27/10/2016	Technologies study and architecture design.
27/10/2016 – 29/10/2016	Data model and business logic development.
29/10/2016 – 02/10/2016	Presentation layer.
02/10/2016 – 02/10/2016	Testing.
02/10/2016 – 04/10/2016	Presentation.
04/10/2016 – 11/11/2016	Final documentation and Delivery.

Table 2: Project schedule.