**Counting Sheep**

Amir Sadiev, Imran Ahmed, Maclean Frazier , Walt Brady

Table of Contents

**1. Project Definition (**100 - 200 words**)** –*Group responsibility*

The purpose of this project is to make a hotel management system that allows both front and back end users to access and use the system. Since this is an application that will allow users to book hotel rooms, it will need friendly and easy to navigate GUI so that booking a room, requesting services and other accommodations is not an issue. The application will also support back end needs to employees whom follow up on their customer’s requests in order to make sure that all necessities are in place and issues are taken care of, as a key feature of this application allows users to directly make requests to the employees of the hotel they are staying in. All of this will be achieved by having a good application that is easy to use on both front and back ends, will allow easy modification, an up to date database that will make sure that there are no miscommunication anywhere in the application.

**2. Project Requirements** – *Group responsibility*

* Functional

The application will serve as a way for clients to book different hotels and rooms to try to get the best prices, deals, request services. It will serve as the gap between the hotel and the user and be the middleman where the customer will try to get the best deal. On the back send, it will allow operators of different hotels to manage their task all in one place. Customer could request service, cleaning and other responsibilities and the staff would get on it to get them done while they can also see other aspects of the business on the back side.

* Usability
  + User interface

There will be a GUI application that will allow both users and hotel employees to retrieve hotel information and make requests…….

* + Performance
* System
  + Hardware

The system will be a desktop application and can be converted to a mobile application so users can use it anytime they want.

* + Software

This will be a desktop application connected to a SQL database

* + Database

This system will use SQL database that updates in real time in response to user inputs.

* Security

Each user will have their own login credentials so that no one else can log in their account. Once the user logs in, there will be a security question the user will need to answer before they can access their entire account. They can reset their password using the question or another form of sensitive information like their social security number.

**3. Project Specification** – *Group responsibility*

* Focus / Domain / Area: Hotel Booking/Management for both front and back end users
* Libraries / Frameworks / Development Environment: Java FX, SQL
* Platform (Mobile, Desktop, Gaming, Etc): Desktop
* Genre (Game, Application, etc): Travel Application

**4. System – Design Perspective** – *Group responsibility*

* Identify subsystems – design point of view
  + Illustrate with class, use-case, UML, sequence ..... diagrams
  + Design choices (Optional)
* Sub-System Communication (Diagram and Description)
  + Controls
  + I/O
  + DataFlow
* Entity Relationship Model (E-R Model)
  + Example - <https://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model>
* Overall operation - System Model
  + Simplified Sub-system to System interaction

**5. System – Analysis Perspective** – *Group responsibility*

* Identify subsystems – analysis point of view
* System (Tables and Description)
  + Data analysis
    - Data dictionary (Table - Name, Data Type, Description)
  + Process models
* Algorithm Analysis
  + Big - O analysis of overall System and Sub-Systems

**6. Project Scrum Report -** *Group Responsibility*

* Product Backlog (Table / Diagram)
* Sprint Backlog (Table / Diagram)
* Burndown Chart

**7. Subsystems**

**7.1 Subsystem 1** – Name 1 - GUI

Initial design and model

* + Illustrate with class, use-case, UML, sequence ..... diagrams
  + Design choices
* Data dictionary
* If refined (changed over the course of project)
  + Reason for refinement (Pro versus Con)
  + Changes from initial model
  + Refined model analysis
  + Refined design (Diagram and Description)
* Scrum Backlog (Product and Sprint - Link to Section 6)
* Coding
  + Approach (Functional, OOP)
  + Language
* User training
  + Training / User manual (needed for final report)
* Testing

**7.2 Subsystem 2** – Name 2 - *Individual responsibility*

* Initial design and model
  + Illustrate with class, use-case, UML, sequence ..... diagrams
  + Design choices
* Data dictionary
* If refined (changed over the course of project)
  + Reason for refinement (Pro versus Con)
  + Changes from initial model
  + Refined model analysis
  + Refined design (Diagram and Description)
* Scrum Backlog (Product and Sprint - Link to Section 6)
* Coding
  + Approach (Functional, OOP)
  + Language
* User training
  + Training / User manual (needed for final report)
* Testing

**7.3 Subsystem 3** – Name 3 - *Individual responsibility*

* Initial design and model
  + Illustrate with class, use-case, UML, sequence ..... diagrams
  + Design choices
* Data dictionary
* If refined (changed over the course of project)
  + Reason for refinement (Pro versus Con)
  + Changes from initial model
  + Refined model analysis
  + Refined design (Diagram and Description)
* Scrum Backlog (Product and Sprint - Link to Section 6)
* Coding
  + Approach (Functional, OOP)
  + Language
* User training
  + Training / User manual (needed for final report)
* Testing

**7.4 Subsystem 4** – Name 4 - *Individual responsibility*

* Initial design and model
  + Illustrate with class, use-case, UML, sequence ..... diagrams
  + Design choices
* Data dictionary
* If refined (changed over the course of project)
  + Reason for refinement (Pro versus Con)
  + Changes from initial model
  + Refined model analysis
  + Refined design (Diagram and Description)
* Scrum Backlog (Product and Sprint - Link to Section 6)
* Coding
  + Approach (Functional, OOP)
  + Language
* User training
  + Training / User manual (needed for final report)
* Testing

**8. Complete System** – *Group responsibility*

* Final software/hardware product
* Source code and user manual – screenshots as needed - Technical report
  + Github Link
* Evaluation by client and instructor
* Team Member Descriptions

***This is just a guide, and use it to create/improve your report. Feel free to add sections. You are responsible for your own subsystem/s, not other members. You have to contribute to the team’s goals and objectives, and develop your subsystem/s, write your documents and slides.***