A FULL EXPERIENCE

IN A SMALL AND INTIMATE HOTEL

WITH HISTORIC FLAVOR

RED TEAM

Hotel Management Software Development Project

[Deliverable 6: Database design]

OCTOBER 31ˢᵗ , 2022

Client Information :
Hôtel Manoir Ramezay – Vivian (Proprietor)
492, rue Claude de Ramezay (Route 227),
Marieville (Québec) J3M 1J6
Telephone : (450) 460-3251 | E-mail: info@manoirramezay.com

<u>CERTIFICATIONS</u>:

I, Red Team MEMBER, **Chi-Tao Li # 9730157**, certify that I have contributed to this deliverable.

Signature: _____ Date: <u>2022-10-31</u>

I, Red Team LEADER, **Li Yu # 2295012**, certify that I have contributed to this deliverable.

Signature: _____ Date: <u>2022-10-31</u>

I, Red Team MEMBER, **Patrick Larocque # 0879202**, certify that I have contributed to this deliverable.

Signature: _____ Date: <u>2022-10-31</u>

I, Red Team MEMBER, **Murat Osmonov # 219501**, certify that I have contributed to this deliverable.

Signature: _____ Date: <u>2022-10-31</u>

I, Red Team member, **Gerasimos Vlassopoulos # 2295049**, certify that I have contributed to this deliverable.

Signature: _____ Date: <u>2022-10-31</u>

Table of Contents

I. Statement of Prior Work

The work herein is possible in part due cumulative learning assignments and projects undertaken in the past, as part of the team's Software Development curriculum. As such, some of the ideas or technical skills used in this project originate partially from prior work. The table below lists past projects undertaken by members of the Red Team, which may contribute in part to elements found in the scope of this project.

Past Projects	Tools Used	Contributor(s)
Mock-ups for personal websites, Application Dev 1	Adobe Photoshop, Cava, Figma	All membres
Simulation Program	Java and Java Swing	Chi-Tao Li
Car Rental System	MS SQL Server	All members
Inventory Management System	Apache Derby Database, Java, and Java Swing	Patrick Larocque
Issue Tacker	C#, Google API, .NET, Firebase	Patrick Larocque
Added Prior Works used for this deliverable	Tools Used	Contributor(s)
Deliverable 1 Report	MS Word, Instagantt.com (for Gantt Chart)	All members
Deliverable 2 Report	MS Word	All members
Database course	Draw.io, Lucidchart	All members
Deliverable 3 Report	MS Word, Draw.io	All members
Deliverable 4 Report	Miro	All members
Deliverable 5 Report	MS Word, Miro	All members

II. Executive Overview

The following document outlines the red team's efforts and activities aimed at collecting system requirements for a new information system, designed to solve the business problems that is identified throughout the scope of this project. These requirements were gathered through a series of interviews with Manoir Ramezay.

Using the existing User Stories from the previous deliverable, we modified it to develop a revised version of the User Story Map based on our future information system. We were always in contact with the client as we revised the User Stories, Acceptance Tests, and User Story Map. From there, we developed hand-drawn prototypes in front of the client and later developed a digital version of the prototype based on the client's comments. Throughout this deliverable, the process used to interact with the client will be mentioned. Overall, many of the tasks, actions and features built into the system will be derived from these User Stories that were gathered over the course of this deliverable.

As with previous reports, the reader will find a statement of their business problems. Next, the reader will find a summary narrative description of the project, indicating what has changed or has been added since the last deliverable. In addition, we've applied 10 usability guidelines and their explanations for the prototypes that were designed for the client. A copy of these prototype interfaces is included in hand-drawn and digital format.

III. Business Problem

Hotel Manoir Ramezay is a 3-star hotel located in Marieville with 14 rooms in total. Clients can make a reservation through hotel website, Expeida, booking.com, telephone and walking in.

Right now, Hotel Manoir Ramezey doesn't have an efficient system in place to keep track of room availability across available booking channels. To keep a running total of past reservations, the front desk employees must update a physical ledger or print out a confirmation from an email issued by a third-party platform whenever a room is booked via one of their platforms. To prevent duplicate reservations and to reflect actual availability, front desk employees must manually alter room availability across all other platforms. This is a very inefficient process, especially when the hotel is busy, during the summer months. This problem will be solved by the new system, which will enable hotel staff to get up-to-date room availabilities through an API. The new system will also enable hotel staff to fulfil reservation requests from clients who make direct hotel reservations, whether they do so over the phone, by email, in person, or online.

Another problem of Hotel Manoir Ramezey is writing down all guests' information on paper is a slow and error prone process. Booking and customer information is dispersed over several tools and platforms, all of which must be cross-referenced each time a booking is made. However, front-desk staff needs to be able to quick verify the guests' personal information for the check-in and check-out process. Additionally, the front-desk staff must consult the physical

ledger or papers to locate a repeat a client's personal information or preferences if he returns to the hotel to make a reservation (if any were noted). This often results in the client being required to repeat a lot of the information that was provided during their previous appointments. The new system will provide a quick searching, modifying and deletion of guests' information and reservations. This will improve the experience for repeat guests especially, and if it will allow the hotel staff to work in a more efficient and organized manner.

Furthermore, considering the computer skills of the hotel owners and staff, we decide to design a system which is easy to use, which will save time to learn and teach for new staff.

IV. Narrative description of the Database Design

We create two primary roles relevant to the system. The first role is the receptionist, who will function as a generic user. The receptionist needs basic authorization, sufficient to create, modify and delete reservations, as well as browse, and update availabilities. The second role is a manger role. The manager will have admin privileges within the system. In addition to being able create, modify, and delete reservations, as well as read and update availabilities, the administrator will be able to create, read, update, and delete users on the system. The administrator will be able to manage user privileges, and the scope of their access to the system.

Both the user and administrator will begin at a login screen, where they will enter their usernames and password. Should they enter valid credentials, they will be logged in, and taken to

the home screen with account privileges matching their account credentials. Should they enter invalid credentials, they will be shown an error message. Upon a successful login, the system will initiate API calls to retrieve up-to-date availabilities from third-party booking platforms. As a note, different screens will be used for the administrator as compared to the receptionist, and this is seen after the log in.

Once at the home screen, the user and the admin both will see some statistics amount the day's availabilities. The user will have the option to navigate to a reservation's menu, allowing the user to create a new reservation. Initiating a new reservation will also necessitate payment functionality. They may navigate to a calendar view of the month's current availabilities. Another view option would be in a form of a list that shows either past, current, or future reservations. From here, the user may be able to search, modify, or delete reservations. There will be a button to refresh availabilities, which will make a series of new API calls to retrieve the most up-to-date booking history. The administrator will have access to the same functionality, however, they will additionally be able to navigate to a user's tab, where they will be able to create, read, update, or delete users. At any moment, the user and the administrator both should be able to exit the application by clicking a button.

V.   Block Diagram

VI. Appendix 1 – Data Dictionary

VII. Appendix 2 – ER Diagram

VIII.   Appendix 3 – Descriptions and Explanations

• (2 marks) Indexes and the database architecture of your design. What indexes are you going to be using in which tables, and why?

• (2 marks) Query optimization in your design. There are going to be many queries. Do you need to optimize them? If so, why and how? If not, why not?

IX. Appendix 4 – The Project size of the Database

The projected size of the database (in MB or GB), now and for the next three to five years. For each table, determine the maximum size of each record. Then, estimate the maximum number of records per table, and then come up with a value for the maximum size of the table. This will give you the maximum size of the database now. Make assumptions about how the number of entries in the database will grow, and then use that number to determine the future size of the database. Show the detailed calculations and assumptions made to arrive at your estimate.

X. Appendix 5 – The Access Speed Required

Explain the access speed required, and how your design will permit this. How often will the database be accessed? How much data will need to be stored or retrieved? What kind of response time will be necessary?

XI. References

9 Principles of Good Web Design. (2021, November 4). Feelingpeaky - Creative Design Agency, London. Retrieved October 8, 2022, from

Adobe. (2021, July 10). *Adaptive vs Responsive Design & Key Considerations | Adobe XD*. Ideas. Retrieved October 12, 2022, from https://xd.adobe.com/ideas/process/ui-design/adaptive-design-vs-responsive-design/

Agile, E. (2021, June 8). Use Cases vs. User Stories: How They Differ and When to Use Them. Easy Agile. Retrieved September 21, 2022, from https://www.easyagile.com/blog/usecases-vs-user-stories/

Barraclough, D. (2021, November 18). *11 Worst Web Design Mistakes to Avoid in 2022*. Expert Market. Retrieved October 8, 2022, from https://www.expertmarket.co.uk/web-design/web-design-mistakes-to-avoid

Brandenburg, L. (2018, May 25). 3 Situations that are Absolutely Perfect for Use Cases!Bridging the Gap | We'll Help You Start Your Business Analyst Career. Retrieved September 27, 2022, from https://www.bridging-the-gap.com/when-would-you-write-a-use-case/

Dublino, J. (2022, September 30). *12 Tips for Building an Effective Business Website*. Business News Daily. Retrieved October 8, 2022, from https://www.businessnewsdaily.com/9811-effective-business-website-tips.html

Jiminez, D. (2022, April 1). *10 Tips That Can Drastically Improve Your Website's User*

   Experience. Retrieved October 12, 2022, from

   https://blog.hubspot.com/marketing/improve-your-websites-user-experience

Nickerson, B. (2022, April 28). *12 Web Design Best Practices for 2022*. Tiller. Retrieved

   October 9, 2022, from https://tillerdigital.com/blog/12-web-design-best-practices-for-
   2022/

Pop, S. A. (2022, May 31). *Essential UI Design Tips for Creating a Good User Interface*.

   TeleportHQ. Retrieved October 9, 2022, from https://teleporthq.io/blog/design-tips-for-
   creating-a-good-user-interface

Requirements 101: User Stories vs. Use Cases. (2017, October 25). Building Better Software.

   Retrieved September 21, 2022, from

   https://www.stellmangreene.com/2009/05/03/requirements-101-user-stories-vs-use-cases/

Top 10 Web-Design Mistakes. (2021, October 29). [Video]. Nielsen Norman Group. Retrieved

   October 12, 2022, from https://www.nngroup.com/videos/top-10-web-design-mistakes

Varga, D. (2020, November 24). User Story vs Use Case: Everything You Need to

   Know.DigitalNatives. Retrieved September 21, 2022, from

   https://www.digitalnatives.hu/blog/user-story-vs-use-case/