

Can Tho, February 25th 2020

Software Engineering Major

Subject: PRJ311 DESKTOP JAVA APPLICATIONS

Topic

CYBER GAMING HOUSE MANAGEMENT SOFTWARE

Member:

|  |  |
| --- | --- |
| (Leader) Duong Tri Tin: | (CE140401) |
| Tat Huynh Anh Khoi: | (CE140273) |
| Ha Thuc Huynh Duy: | (CE140033) |
| Tran Ngoc Man Huy: | (CE140289) |
| Luu Tien Minh: | (CE140300) |

**THANKS**

With the assignment of Mr. Luong Hoang Huong, we have implemented the project "Cyber ​​Gaming House". In order to successfully complete the project, we would first like to thank Mr. Luong Hoang Huong for facilitating, orienting and providing necessary things during the working time.

Due to limited knowledge and time, the software will have shortcomings. I hope that the teachers and friends will give more comments to help the group have a better understanding of the software.

Thank you sincerely!

Can Tho, February 25, 2020

Tat Huynh Anh Khoi

Duong Tri Tin

Ha Thuc Huynh Duy

Tran Ngoc Man Huy

Luu Tien Minh

**Comments**

**………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………**

**………………………………………………………………………………………………………………………………………………………………………………………………………**

**………………………………………………………………………………………………………………………………………………………………………………………………………**

**………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………**

**………………………………………………………………………………………………………………………………………………………………………………………………………**

**………………………………………………………………………………………………………………………………………………………………………………………………………**

**………………………………………………………………………………………………………………………………………………………………………………………………………**

**Table of contents**

Contents

[Terminology and acronyms 4](#_Toc36415206)

[List of pictures 4](#_Toc36415207)

[Abstract 4](#_Toc36415208)

[I. Introduction 5](#_Toc36415209)

[I.1 Problem statement 5](#_Toc36415210)

[I.2 Topic goals & Research subjects 6](#_Toc36415211)

[I.2 Research content 6](#_Toc36415212)

[I.3 The main contributions of the topic 6](#_Toc36415213)

[THEORETICALLY 7](#_Toc36415214)

[PRODUCT 7](#_Toc36415215)

[I.4 Dissertation layout 7](#_Toc36415216)

[II. Content part 7](#_Toc36415217)

[II.1 Java 7](#_Toc36415218)

[II.2 Description 9](#_Toc36415219)

[II.2.1 Introduction 9](#_Toc36415220)

[II.3 Overview description 9](#_Toc36415221)

[III. Design and installation of solutions 10](#_Toc36415222)

[III.1 System design 10](#_Toc36415223)

[Introduction 10](#_Toc36415224)

[System overview 11](#_Toc36415225)

[System architecture 11](#_Toc36415226)

[Data design 14](#_Toc36415227)

[Install solutions 20](#_Toc36415228)

[III. Testing and evaluation 24](#_Toc36415229)

[III.1 Porpose 24](#_Toc36415230)

[III.2 Scenario testing 24](#_Toc36415231)

[IV. Conclusion 25](#_Toc36415232)

[IV.1 Achievements 25](#_Toc36415233)

[IV.2 limit 25](#_Toc36415234)

[V. References 25](#_Toc36415235)

# Terminology and acronyms

|  |  |
| --- | --- |
| Terminology / acronyms | Means |
| CPU | Central Processing Unit |
| CMD | Canonical data model |
| PC | Personal Computer |
| PS | Play Station |
| JFC | Java Foundation Classes |
| JVM | Java virtual machine |
| RAM | Random access memory |
| Lan | Local Area Network |
| PCI | Peripheral Component Interconnect |
| PSU | power supply unit |

# List of pictures

[Figure 1. Architectural design 12](#_Toc36415767)

[Figure 2. Use-case diagram 13](#_Toc36415768)

[Figure 3. User case 14](#_Toc36415769)

[Figure 4. Admin case 15](#_Toc36415770)

[Figure 5. Device management 16](#_Toc36415771)

[Figure 6. Staff Management 17](#_Toc36415772)

[Figure 7. Room management 18](#_Toc36415773)

[Figure 8. User login 19](#_Toc36415774)

[Figure 9. Food & Drink 20](#_Toc36415775)

[Figure 10. Customer interface 20](#_Toc36415776)

[Figure 11. Device management interface 21](#_Toc36415777)

[Figure 12. Staff management interface 21](#_Toc36415778)

[Figure 13. Room management interface 22](#_Toc36415779)

[Figure 14. Customer menu interface 23](#_Toc36415780)

[Figure 15. Orders management interface 23](#_Toc36415781)

[Figure 16. ER diagram 24](#_Toc36415782)

# Abstract

Cyber ​​Gaming House software is a software to manage and organize activities in the Net, including activities such as customer management, management of rooms, equipment, food, drinks and staff. At the same time, the software provides expenditures to make development planning more accurate.

Net shop is now an expanded model of services such as entertainment services such as gaming including devices such as PC desktop, PlayStation, ... wifi support services, and guest rooms including travel and overnight. Besides, the restaurant also serves foods and drinks to customers. Thereby, the software helps to manage and organize the activities in the Net shop according to the plan and produce more effectively, minimizing errors during the operation of the Net shop.

# I. Introduction

## I.1 Problem statement

There are many reasons why investors choose Cyber ​​Game model. The main reason is that this model brings a tremendous profit. Unsaturated Cyber ​​Game Market: Cyber ​​Game is completely different from regular net shops. If the typical pub can own a steady profit, then when investing in Cyber ​​Game that number will increase a lot of times.

This is simply because investing in Cyber ​​Game requires more capital than usual. But also need a management level, aesthetics and methodical strategy. Cyber ​​Game has a lot of money services: Net shops will usually rely on the money earned from playing by the hour, eating, and loading cards. Meanwhile Cyber ​​Game owns a series of catering services with a large turnover. For example: Catering, coffee, milk tea at the shop. Service of leasing places to organize tournaments, offline, fan meeting, soft drinks service for those who do not want to play games but love technology.

This stage is now considered a golden time to invest in Cyber ​​Game. Therefore, anyone who is fast will gain the upper hand in the market.Cyber ​​software includes:

room manager

device management

Employee management

admin and customer management

Food & drink order system

chat system

If management software, project management software or financial management systems provide management information for an entire enterprise, with computer room management software, users not only monitor workstation status but can also calculate revenue and other information on the net room system. It can automatically manage access time and charge usage of workstations in the game room. This includes both internet access and associated services. In addition, it also manages the use of membership cards net. Balance daily income and financial statements used. Provides the ability to monitor, control and protect workstation systems.

Even the private room service for youtuber and professional gamers generates huge revenue.

## I.2 Topic goals & Research subjects

fluent in java language, It was initially developed for interactive television to let application developers write code once and run it on any platform without the need to decompile it, thanks to JVM. For the obvious reason, such flexibility and multiplatform usability can be applied anywhere.

fluent in using swing in java, Swing is a set of program component s for Java programmers that provide the ability to create graphical user interface ( GUI ) components, such as buttons and scroll bars, that are independent of the windowing system for specific operating system . Swingcomponents are used with the JFC.

Know more about the Apache poi, JDBC library.

## I.2 Research content

- Connect to sql server jdbc database management system

- How to program with java language

- Export data to excel file with Apache poi

- Java swing interface support tool

## I.3 The main contributions of the topic

### THEORETICALLY

- Java programming knowledge using netbeans tool

- Knowledge using sql server jdbc database

- Programming knowledge with Apache poi library.

### PRODUCT

Is a Cyber Game management, which helps control the information related to the client such as time to use the service, accompanying services, especially the system that provides the ability to store user account information. to manage the account every time you log in and use it.

## I.4 Dissertation layout

- The layout of the text consists of 3 parts

- The introduction presents the problem, the target of the topic, the object of the study, the content of the research and the main contributions of the topic.

- The content consists of 3 chapters:

1. describe the problem.

2. Design and installation of solutions.

3. Testing and evaluation.

- The conclusions section presents the results of the constraint and the development direction of the problem.

- Finally there are references and appendices.

# II. Content part

## II.1 Java

1. Introduction

Java is a general-purpose programming language that is class-based, object-oriented, and designed to have as few implementation dependencies as possible. It is intended to let application developers write once, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture.

Java was originally developed by James Gosling at Sun Microsystems (which has since been acquired by Oracle) and released in 1995 as a core component of Sun Microsystems' Java platform. The original and reference implementation Java compilers, virtual machines, and class libraries were originally released by Sun under proprietary licenses.

Some advantages of using Java

* Simple
* Object-Oriented
* Platform-Independent
* Distributed
* Secure
* Rich APIs
* Powerful Opensource Rapid Development Tools
* Robust
* Resource Availability

The Java syntax is heavily borrowed from C & C ++ but has a simpler object-oriented syntax and fewer lower-level processing features. Therefore, writing a program in Java is easier, simpler, and less time consuming to fix errors. But in terms of object-oriented programming, Java is more complex.

In Java, memory leaks are unlikely to occur because the memory is managed by the Java Virtual Machine (JVM) by automatically "clearing junk". Programmers do not have to care about allocating and deleting memory like C and C ++. However, when using network resources, IO files, databases (outside of JVM control) and programmers do not close (close) streams, data leakage can still occur.

1. Open-sources JDBC

JDBC (Java Database Connectivity) is a standard API used to interact with relational database types. JDBC has a collection of classes and interfaces for Java applications that can talk to databases.

1. Import to Excel (Use Apache Poi)

Many a time, a software application is required to generate reports in Microsoft Excel file format. Sometimes, an application is even expected to receive Excel files as input data. For example, an application developed for the Finance department of a company will be required to generate all their outputs in Excel.

Any Java programmer who wants to produce MS Office files as output must use a predefined and read-only API to do so.

Apache POI is a popular API that allows programmers to create, modify, and display MS Office files using Java programs. It is an open source library developed and distributed by Apache Software Foundation to design or modify Microsoft Office files using Java program. This is a project run by the Apache Software Foundation, and previously a sub-project of the Jakarta Project, provides pure Java libraries for reading and writing files in Microsoft Office formats, such as Word, PowerPoint and Excel.

## II.2 Description

### II.2.1 Introduction

1. Purpose

The purpose of this section is to provide users with software for managing activities in the net shop.

Document for later analysis and reconciliation for subsequent design, development, and maintenance.

This part of the document is aimed at a group of readers who are designers, programmers, testers, maintainers and all those who want to learn in depth about how the product works.

1. -Ambit

Products include desktop software. Allows users to log in to the software and perform management functions and activities in the Net shop.

1. Arrangement

The layout of this document includes overview descriptions, external communication requirements, product functions and non-functions, solution options, and solution analysis.

## II.3 Overview description

1. Context

Net management software today is essential for Net shops because they expand the scale including services for players and catering services and about machinery.

Products include desktop software for managing rooms, equipment, ... servers, databases of entities and Net service user interface.

1. Function

* Management software
* Room management
* Food and Drink management
* Device management
* Order management
* Customer management
* Staff management
* Customer interface
* Login
* Register
* Menu Customer
* Order of Customer
* Server RMI
* Call Room

1. Characteristic of user

All user can use this software

1. Operation environment

* Software runs on computer with CPU core or upper
* Operation system Windows 10

1. Binding implementation and design

* Entity binding
* Software is bond to Database
* Software process quickly requests of user
* Interface binding
* Interface is easily to using of people
* Font is clearly

# III. Design and installation of solutions

## III.1 System design

### Introduction

a. Purpose

The program content provides an overview of system management, system architecture, data and applied analytical solutions.

The intended audience of this program's content designers, programmers, testers.

b. Range

The scope of management of the system includes control software, management system, for individuals or cyber games who need to use the system for the purpose of managing and developing business on a large scale and can Further research on the features of the product.

c. Overview of this chapter

This chapter consists of 5 parts:

- Part 1: Introduction

- Part 2: System overview

- Part 3: System architecture

- Part 4: Data design

- Part 5: Install solutions

### System overview

- System function:

+ Ensure the requirements of accuracy, consistency, ease of use, and maintenance.

+ Functions that meet user requirements.

- Context:

+ Meet the needs of using and managing users' systems.

- Design:

+ Friendly interface, easy to use, with reasonable layout buttons.

+ Icons that are evocative, brief notifications easy to understand.

+ The program is easy to understand for maintenance and upgrade phase.

### System architecture

a. Architectural design

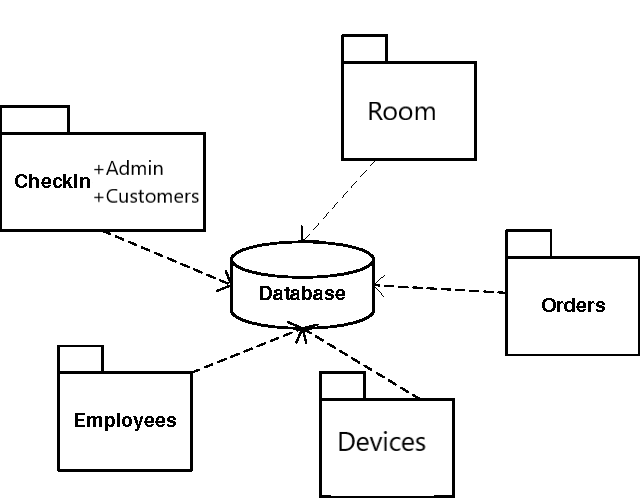


Figure 1. Architectural design

System requirements:

- Hardware & Software:

Main: Intel S1200 Motherboard Socket 1151 | 6 x SATA 3 6Gb / s

CPU: Intel® Core i3 8100 Socket 1151 Processor

Ram: 32Gb ECC (08Gb x 4) cDDR4 Ecc for server

SSD: 1 x 128Gb dedicated for bootrom with Sata 3 speed of 6Gb / s.

SSD: 3 x 256Gb dedicated for bootrom with Sata 3 speed of 6Gb / s.

HDD: 3TB WD Black SATA 6Gb / s 64MB Cache (Full Game)

Lan Card: 3 x Removable PCI cards 04 PORT + 02 PORT Onboard = 14 PORT

(Recommend the 20Gb optical system, contact for advice)

PSU: Source JETEK for server 600W (80 Plus Gold)

Case: SERVER BOOTROM Performance - BEST CASE

Windows 7 operating system or higher

.Net framework 4.0

b. Use-case diagram

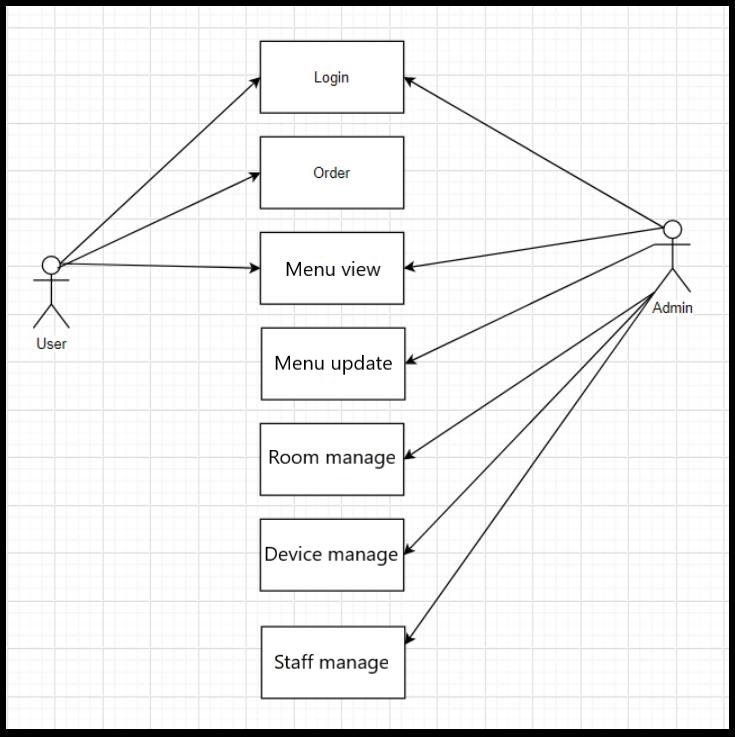


Figure 2. Use-case diagram

c. Facility design

- Design the system architecture according to each module individually and in the order of dependencies among modules, modules dependent on other modules will be designed after the module on which it depends. This type of design has a stratified form, easily checks for components when errors occur in a bottom-up fashion, and due to the simple system, it does not take much time to check.

- The design is divided into function groups, each group may have smaller functions.

### Data design

a. Data description

The problem requires storage of large amounts of data. The system's data will be uploaded to the database continuously, as well as be accessed regularly, requiring management and analysis of data through admin (mostly) and user tasks.

b. Data model UML

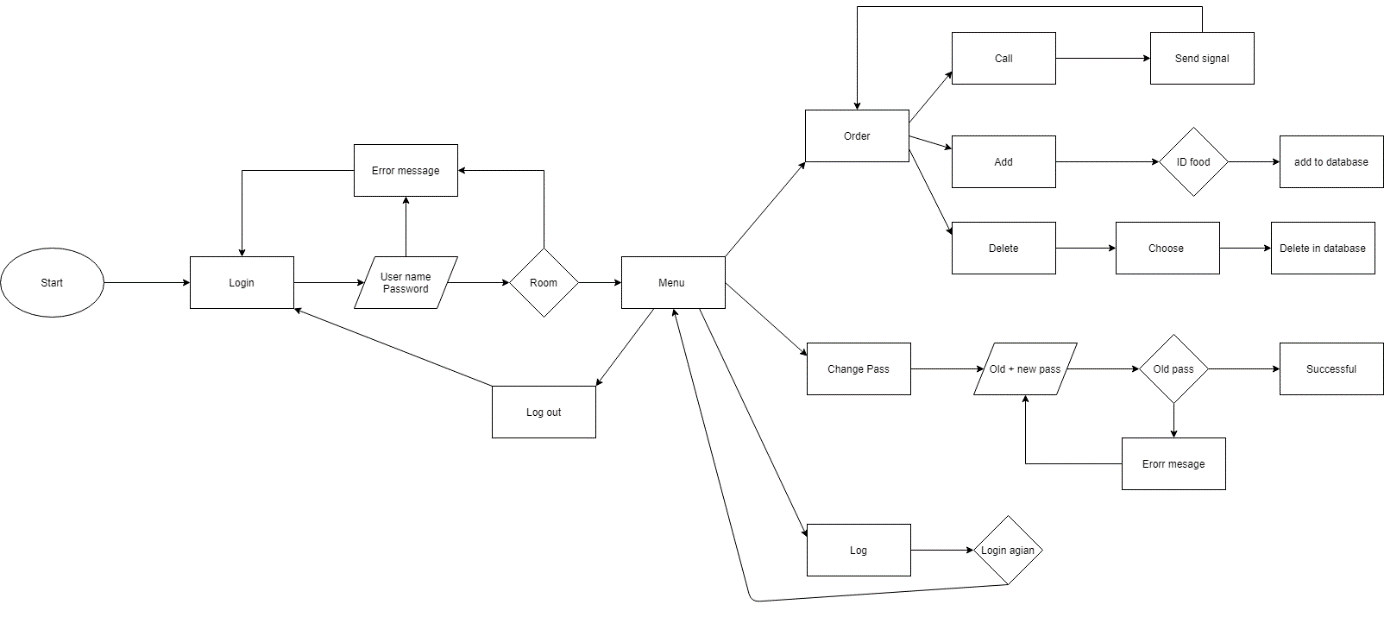


Figure 3. User case

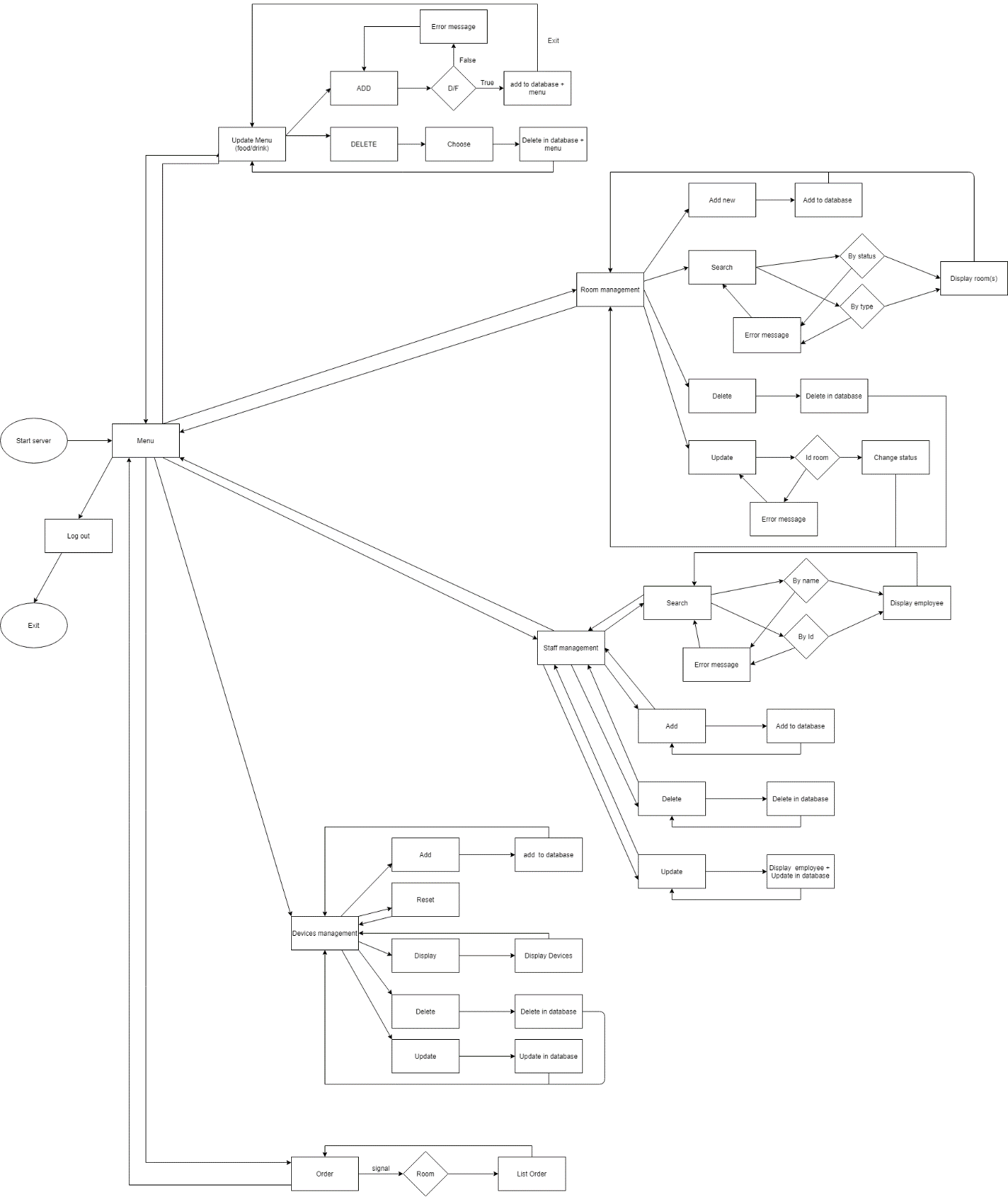


Figure 4. Admin case

Data Entity

\* Device:

|  |  |  |
| --- | --- | --- |
| Attribute name | Data type | Explain |
| id | int | ID code for each device |
| Name | varchar(10) | Machine name |
| CPU | nvarchar(50) | Cpu name |
| mainboard | nvarchar(100) | Mainboard name |
| VGA | nvarchar(150) | VGA name |
| SizeScreen | nvarchar(20) | Screen sizze |
| Screen | nchar(10) | How much screen |
| RAM | nvarchar(10) | Ram name |
| Gamingkeyboard | varchar(50) | Keyboard real or fake |
| Gamingmickey | varchar(10) | Mouse carrier name |
| GamingHeadPhone | varchar(10) | Wired headset or bluetooth |
| GamingMousePad | varchar(15) | Mouse lining category |
| Gamingchair | varchar(10) | Whether or not gaming chairs |
| ps4 | varchar(10) | Whether or not ps4 |

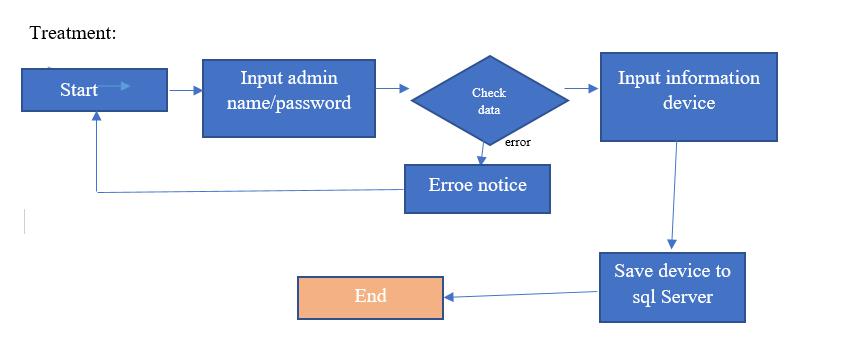


Figure 5. Device management

\* Staff:

|  |  |  |
| --- | --- | --- |
| Attribute name | Data type | Explain |
| id | nvarchar(5) | ID code for each employee |
| First\_name | nvarchar(10) | First name of employee |
| Last\_name | nvarchar(20) | last name of employee |
| birthdate | Date | birthdate of employee |
| ssn | nvarchar(12) | Social security number |
| address | nvarchar(50) | Address of employee |
| phone | nvarchar(12) | Phone number of employee |
| email | nvarchar(20) | Email of employee |
| status | nvarchar(10) | Status of employee |
| position | nvarchar(15) | Position of employee |
| Hire\_date | Date | Hire date |
| Renewal\_date | Date | Renewal date |
| salary | int | Salary of employee |

Treatment:

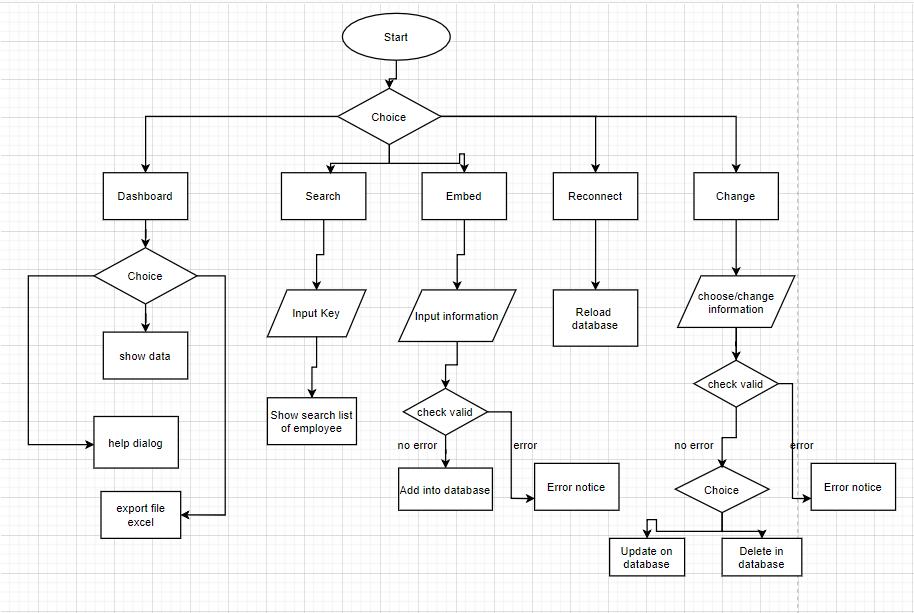


Figure 6. Staff Management

\* Room:

Entity Room:

Explain: Shows room information including room code, room type, number of people in that room, room status. In which the room code is set as the primary key to avoid duplication.

|  |  |  |
| --- | --- | --- |
| Attribute name | Datatype | Explain |
| Id\_room | Nvarchar(7) | Room code |
| Type\_room | Nvarchar(20) | Room name |
| Number\_of\_person | int | Number of player in one room |
| status | Nvarchar(15) | Available or not |

Treatment:

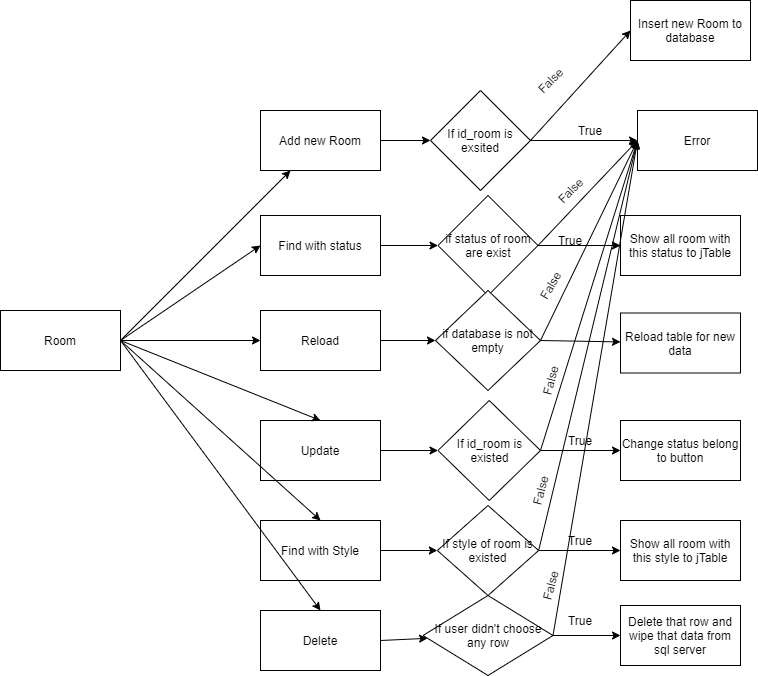


Figure 7. Room management

\*Customer:

|  |  |  |
| --- | --- | --- |
| Attribute name | Datatype | Explain |
| id | Nvarchar(20) | Id of user |
| username | Nvarchar(20) | user name |
| password | Nvarchar(32) | Password of user |

Playing:

|  |  |  |
| --- | --- | --- |
| Attribute name | Datatype | Explain |
| ID | int | Id of device |
| id | Nvarchar(20) | Id of user |

Treatment:

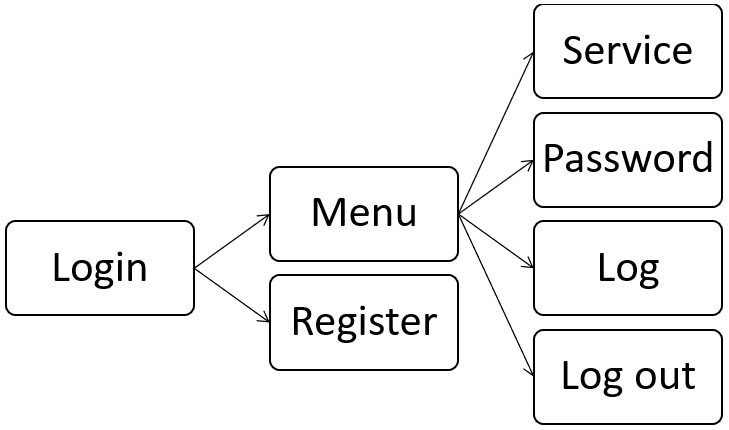


Figure 8. User login

\*Order Food & drink:

Entity Food:

Explain: Display information including product code, product name, product price (the product is the drink or the food). In which the product key is set as the primary key so that the codes do not overlap.

|  |  |  |
| --- | --- | --- |
| Attribute name | Datatype | Explain |
| id\_food | nvarchar(10) | Product code |
| name\_food | nvarchar(50) | Product name |
| price | int | Product original cost |

Entity Order:

Explain: Created with product code and room code as foreign key. Which shows the order of each room with dishes and drinks.

|  |  |  |
| --- | --- | --- |
| Attribute name | Datatype | Explain |
| Id\_room | Nvarchar(7) | Room code |
| id\_food | nvarchar(10) | Product code |
| name\_food | nvarchar(50) | Product name |
| Amount | int | the number of products |
| Price | int | Amount \* original cost |
| Status | Nvarchar(15) | Served or not |

Treatment:

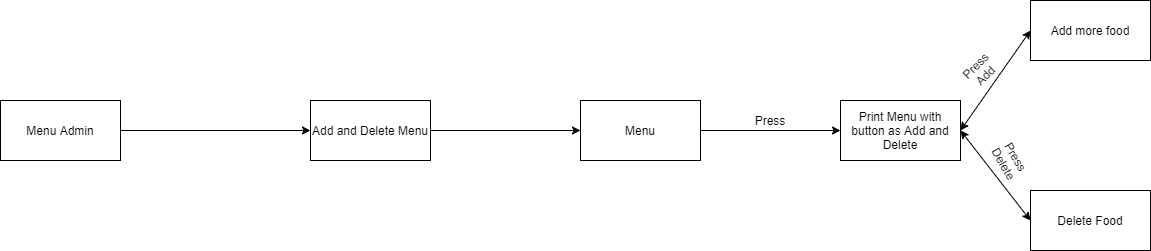


Figure 9. Food & Drink

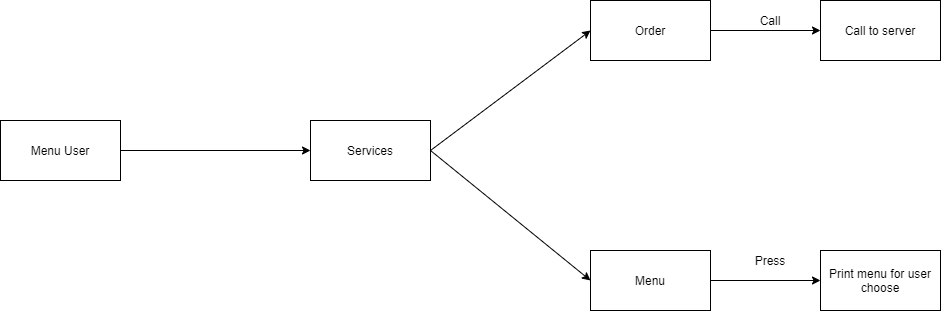


Figure 10. Customer interface

### Install solutions

Functional list

|  |  |  |
| --- | --- | --- |
| Number | Function | Using by |
| 1 | Room Management | Admin |
| 2 | Staff Management | Admin |
| 3 | Device Management | Admin |
| 4 | Food / Drink order system | User |
| 5 | Customer Management | Admin |

The main function of the system

Device:

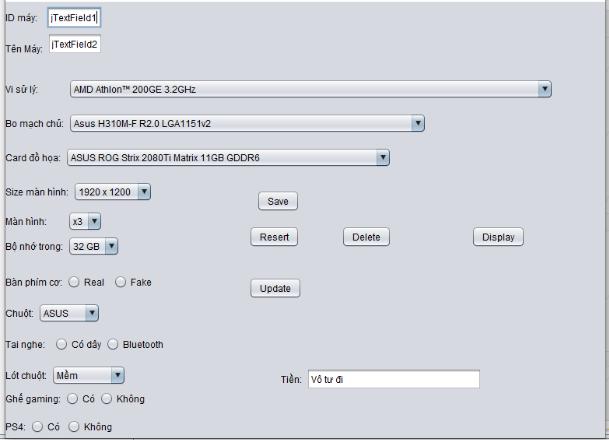


Figure 11. Device management interface

Staff:

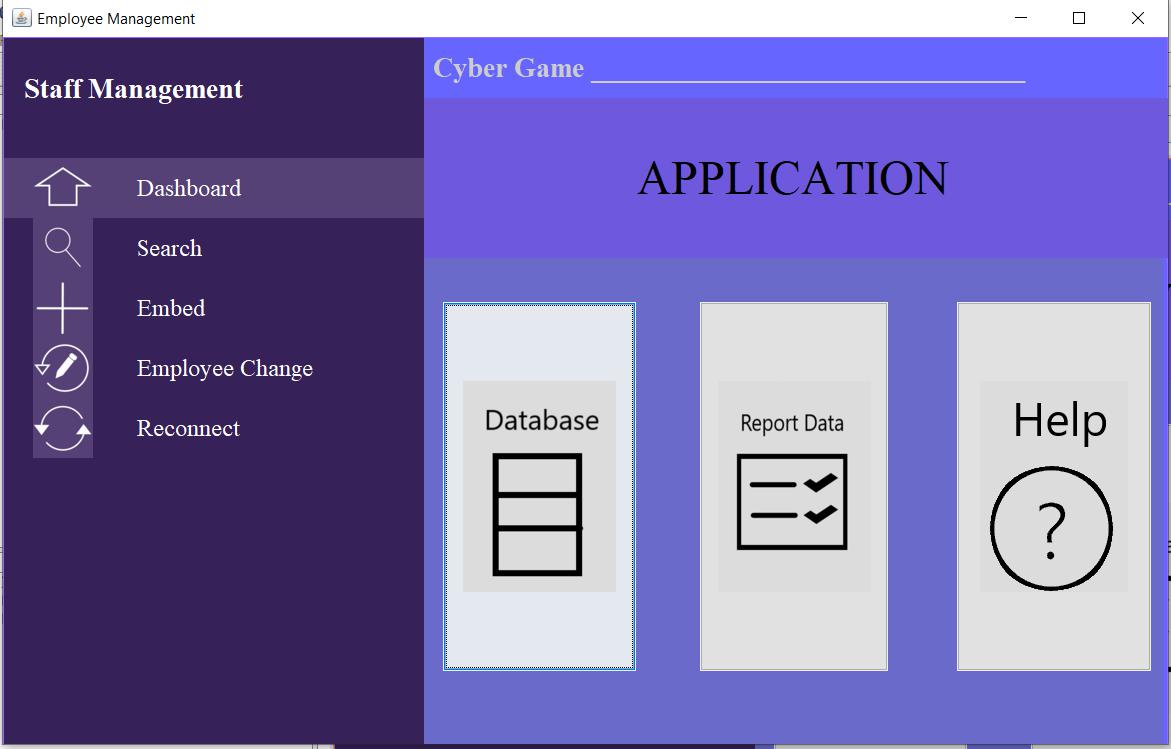


Figure 12. Staff management interface

Room:

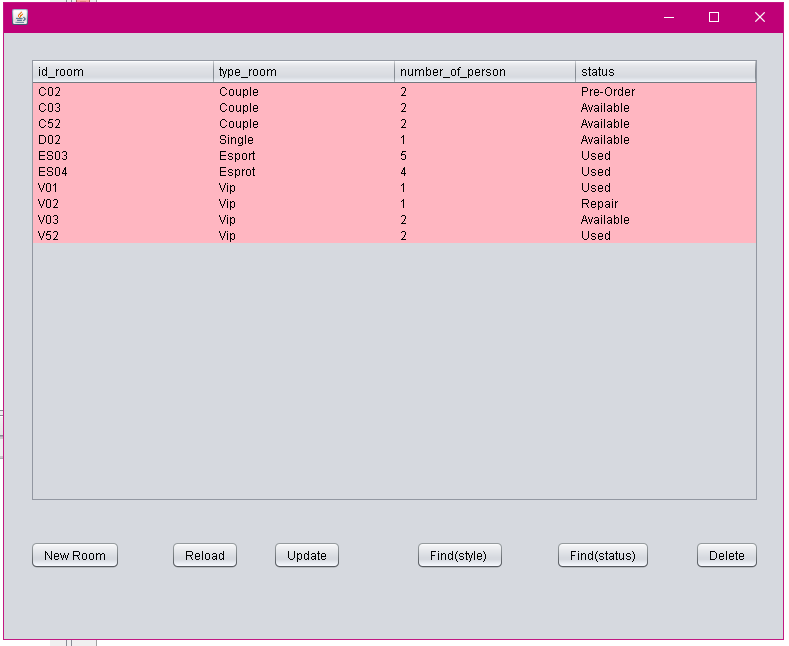


Figure 13. Room management interface

Customer:

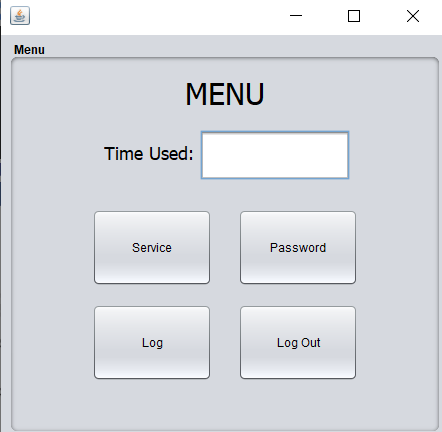


Figure 14. Customer menu interface

Orders:

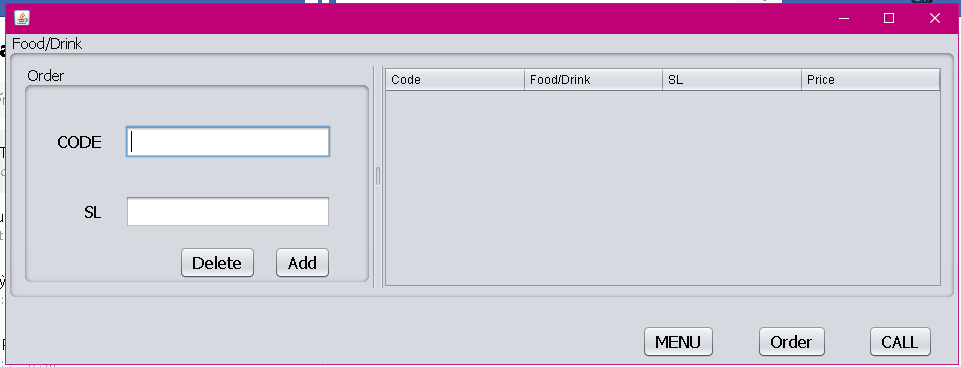


Figure 15. Orders management interface

Entity Relative:

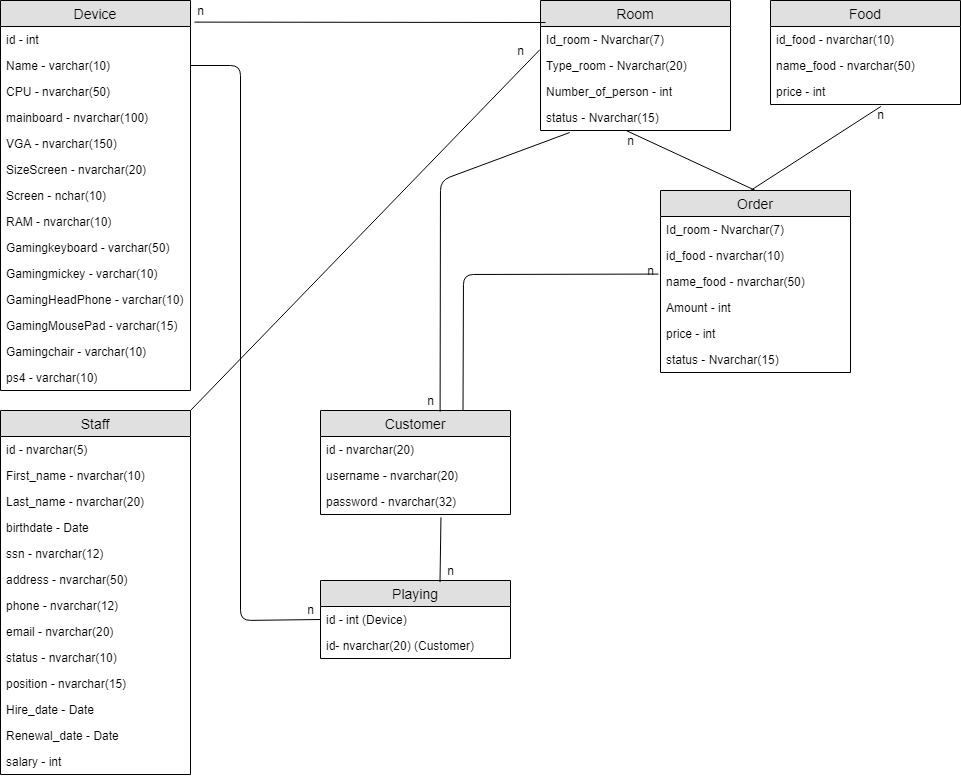


Figure 16. ER diagram

# III. Testing and evaluation

## III.1 Porpose

In order to check if the specification is correct, the functions work as expected. Presentation of testing methods, how to overcome errors.

## III.2 Scenario testing

|  |  |  |
| --- | --- | --- |
| Function | Scenario | Status |
| Device | Add new device  (id no duplicate, information not null) | Excute success |
| Device | Add new device  (id duplicate) | error |
| Staff | Delete choose employee, comfirm with yes option | Excute success |
| Staff | Add new employee  (do not input hire date) | error |
| Room | Find room  (input key or just stick button and search) | Excute success |
| Room | Reload  (reload database) | Excute success |
| Customer | Login with no input username / password | error |
| Customer | Login with wrong input username / password | error |
| Orders | Order  (choose item) | Excute success |
| Orders | Call  (chat with serving staff) | Excute success |

# IV. Conclusion

## IV.1 Achievements

- Solving the problem requires a good combination of database management system software SQL server, Apache poi.

- Use the optimal hardware solution.

- Successfully studied and applied the theoretical basis of Java programming as well as netbeans and SQL server systems.

## IV.2 limit

- The colors are not aesthetic yet

- The interface is not too beautiful

# V. References

https://www.tutorialspoint.com/apache\_poi/index.htm

<https://phongnet.com/>

https://www.vietngapc.vn/gamenet/tai-sao-mo-hinh-gaming-house-tai-cac-cyber-game-lai-thu-hut-cac-team-sport.html