

**University of Technology, Jamaica**  
**School of Computing and Information Technology**  
**CIT3009 – Advanced Programming**

|                        |   |                     |                          |
|------------------------|---|---------------------|--------------------------|
| <b>Facilitator(s):</b> | <b>Christopher Panther   Horrett Scarlett</b>                                   |                     |                          |
| <b>Assessment:</b>     | <b>Group Assignment (Group size, no less than 4 persons, maximum 5 persons)</b> |                     |                          |
| <b>Given week of:</b>  | <b>September 27, 2021</b>   | <b>Due week of:</b> | <b>November 10, 2021</b> |

Grizzly's Entertainment is a stage equipment business that offers the rental of equipment for events requiring staging, lighting, power, and sound. The managers of the company are currently facing an issue scheduling the equipment for all the events for the upcoming Jamaican Christmas holiday season. You are hired to write a software that will assist the company in the management and scheduling of their assets.

Design a JAVA application that will meet the requirements. Your application should be built using the ***Client/Server*** architecture. ***Your Database should be located on the server machine*** where the client will make requests to the server over the network. Your client application **must** send requests to the server application as **only the server** application can establish connections to the database. The server will grant the corresponding request from the client. Your application should be developed with a **graphical user interface** that will aid the users of the system in their respective data entry and processing. **Please include a GUI to interact with the data for all database tables.** The server application need not have a GUI. However, you may include graphical components to enhance user experience.

**Instructions:**

1. System Name: Grizzly's Entertainment Equipment Rental
2. Logging should be on both client side and server side
3. Use appropriate exception handling
4. Graphical User Interface should be implemented as a Multi-document Interface (MDI)
5. Develop your group solution with the use of a GitHub Repository or its equivalent to host the project's source code files, to which each member of the group is

expected to contribute. Proof of members' contribution will be verified via the project's history logs.

6. I am to receive an invitation from each group, to collaborate on their project, within seven (7) days of this document being distributed.

## **Development Tools**

- Eclipse IDE, IntelliJ IDE, or NetBeans IDE (JDK 1.8 or higher)
- MySQL 8 current version (Administered with MySQL Workbench or PhpMyAdmin)
- MySQL Connector/J (JDBC)
- Hibernate 5 (Object Relational Mapping)
- Apache Log4J2 (Logging and Audit Trails)
- SpringFramework (Inversion of Control, Already Packaged in Netbeans)
- Git or its equivalent (Version Control)

## **Deliverables**

- Group Report
  - Details of each group members contribution
  - Declarations of Authorship - one for each group member
- Application User Manuals
  - Employee User Guide (Document or Screen Recording)
  - Customer User Guide (Document or Screen Recording)
- Server-side application
- Client-side application – Customer
- Client-side application – Employees

## **Assessment**

- Comments, Programming Conventions, and documentation
  - UML Class and Database ER Diagrams
  - Application User Manuals
- Graphical User Interface

- Client/Server Networking Model
- Database Connectivity (mandatory) / Use of Files (optional)
- Logging, Exception Handling, User & data input validation
- Classes, Interfaces, Inheritance and Polymorphism
- Core Functionality described above
- 30 minutes grading interview – all group members must be able to fully present the project.

### **Late Submission**

As per the university's policy, late submission of projects will attract a penalty as follows:

| <b>Time of Submission after Deadline</b> | <b>Penalty (%)</b> |
|--|--------------------|
| One Day                                  | 10                 |
| Two Days                                 | 20                 |
| Three Days                               | 50                 |
| Beyond Three Days                        | 100                |

**Milestones: Due at the start of each lab session after the week project is given.**

**Milestone 1: 100 Marks – Weight 20% (Due week 2)**

**Database Design and Domain**

**[30 Marks]**

- Create Tables (Fields, Data Types, Constraints, Relationships)
- Normalized Tables (At least 3rd Normal form)
- Create Domain Objects (Including Inheritance)

**Traditional Database Connectivity**

**[35 Marks]**

- CRUD operations (select with condition, multiple select, insert, update, delete)
- Managing Result Sets

- Manage and Log All Exceptions

### **Object-Relational Mapping**

**[35 Marks]**

- Setup ORM
- CRUD operations (select with condition, multiple select, insert, update, delete)
- Managing Result Sets
- Manage and Log All Exceptions

### **Milestone 2: 100 Marks – Weight 20% (Due week 3)**

#### **Events Logging**

**[50 Marks]**

- Setup Loggers
- Setup Appenders (Console and Rolling File)
- Setup Layouts
- Logging All Events (Information, Warnings)
- Logging All Exceptions (Errors and Fatal)

#### **Version Control / Continuous Integration**

**[50 Marks]**

- Create Project Repository (Git Desktop Client, Eclipse Git or GitHub/Equivalent)
- Collaboration by group members using Git.

### **Milestone 3: 100 Marks – Weight 20% (Due week 4)**

#### **TCP/IP Socket-based Networking and Object Serialization**

**[70 Marks]**

- Setup TCP/IP Client
- Setup TCP/IP Server
- Mechanisms to send serialized objects over network streams.
- Log All Events
- Manage and Log All Exceptions

#### **Multi-Threading**

**[30 Marks]**

- Thread TCP/IP Server Engine to manage multiple client requests.
- Log All Events

- Manage and Log All Exceptions

#### **Milestone 4: 100 Marks – Weight 10% (Due week 5)**

##### **Advanced Graphical User Interface**

**[60 Marks]**

- Create Parent Window
- Create Internal Forms for each database table (insert, select, update, delete)
- JTables with Table Models must be used.
- Implement Menu Bars, Keyboard Short Cuts, Tool Tips, Mnemonics.

##### **Inversion of Control**

**[40 Marks]**

- Create Interfaces
- Create Plug-ins
- Configure IOC Framework
- Load plug-ins at runtime

#### **System Functionalities: 70 Marks – Weight 20%**

##### **Authentication:**

- As a customer, I should be able to log in with my customer ID and password to gain access to my dashboard. **[5 marks]**
  - As an employee, I should be able to log in with my staff ID and password to gain access to my dashboard. **[5 marks]**
- \*\* You may generate customer account information including their account balance.**

##### **Customer Services**

- As a customer, I should be able to choose the equipment I would like to rent from a list of available equipment by category (staging, lighting, power, and sound). I should also be able to indicate the date for which I would require the equipment, so that an employee may view my enquiry at a later time and advise me on the equipment's availability for that date along with a quotation on cost. **[5 marks]**

- As a customer, I should be able to view a list of my past transactions, showing for each transaction in the list, the date of rental, equipment rented, and cost. **[6 marks]**
- As a customer, I should be able to view a past transaction and all the associated details. **[7 marks]**

### **Employee Services**

- As an employee, I should be able to view a list of rental requests from my dashboard. **[8 marks]**
- As an employee, I should be able to view all equipment acquired and stored in the company's inventory, along with its rental status: booked or available. **[8 marks]**
- As an employee, I should be able to schedule a piece of equipment for an upcoming event. I should not be allowed to schedule the same piece of equipment for two events simultaneously. **[8 marks]**
- As an employee, I should be able to respond to a customer, advising of the availability of the equipment, they are interested in. **[8 marks]**
- As an employee I should be able to create quotations, invoices and receipts for bookings and rentals. **[10 marks]**

### **Documentation: 10 Marks – 10%**

**[10 marks]**

### **Bonus Feature**

- Your application should be able to print the invoices in a presentable format. For this, you may export the invoice to a PDF document. The marks for this section may be awarded at the discretion and satisfaction of your lab tutor. This does not entail a screenshot of your Java GUI application form. **[10 marks]**