Software Engineering Project

Equipment Rental System

Submitted By: Roman Hnatyshyn T00228949

Computing with Software Development

Date Submitted: 20/04/2024

**Table of Contents**

[1. Introduction/overview 4](#_Toc164517533)

[2. Functional Components 5](#_Toc164517534)

[3. User Requirements 6](#_Toc164517535)

[3.1. Manage Equipment 6](#_Toc164517536)

[3.2. Manage Clients 6](#_Toc164517537)

[3.3. Process Rentals 6](#_Toc164517538)

[3.4. Perform Admin 6](#_Toc164517539)

[4. System Requirements 7](#_Toc164517540)

[4.1. System Level Use Case Diagram 8](#_Toc164517541)

[4.2. Manage Equipment 9](#_Toc164517542)

[4.2.1. Add Equipment Category 9](#_Toc164517543)

[4.2.2. Update Equipment Category 11](#_Toc164517544)

[4.2.3. Add Equipment 13](#_Toc164517545)

[4.2.4. Update Equipment 15](#_Toc164517546)

[4.3. Manage Clients 18](#_Toc164517547)

[4.3.1. Register Client 18](#_Toc164517548)

[4.3.2. Update Client 21](#_Toc164517549)

[4.3.3. De-register/Restore Client 24](#_Toc164517550)

[4.4. Process Rentals 26](#_Toc164517551)

[4.4.1. Place Rental 26](#_Toc164517552)

[4.4.2. Cancel Rental 31](#_Toc164517553)

[4.4.3. Record Collection 35](#_Toc164517554)

[4.4.4. Record Return 37](#_Toc164517555)

[4.5. Perform Admin 39](#_Toc164517556)

[4.5.1. Analyse Category Profit 39](#_Toc164517557)

[4.5.2. Yearly Revenue Analysis 41](#_Toc164517558)

[5. System Model 43](#_Toc164517559)

[5.1. Level-0 DFD 45](#_Toc164517560)

[5.2. Level-1 DFD 46](#_Toc164517561)

[5.3. Level-2 DFD (Process P1: Manage Equipment) 47](#_Toc164517562)

[5.4. Level-2 DFD (Process P2: Manage Clients) 48](#_Toc164517563)

[5.5. Level-2 DFD (Process P3: Process Rentals) 49](#_Toc164517564)

[5.6. Level-2 DFD (Process P4: Perform Admin) 50](#_Toc164517565)

[6. Data Model (Class Diagram) 51](#_Toc164517566)

[6.1. Class Diagram 51](#_Toc164517567)

[6.2. Relational Schema 52](#_Toc164517568)

[6.3. Database Schema 53](#_Toc164517569)

[7. Conclusion 55](#_Toc164517570)

# Introduction/overview

The EquipmentSYS is a management system designed to revolutionize the way organizations handle equipment, clients, rentals, and administrative tasks. The system is developed to address all the needs by providing a comprehensive and user-friendly solution to streamline operations and improve decision-making processes.

This system aims to centralize and optimize the management of equipment items, client interactions, and rental transactions. The user-centric approach ensures that each aspect, from equipment categorization to client information management and rental process, is thoughtfully designed to create an efficient system.

The EquipmentSYS emphasizes user requirements, translating them into high-level functionalities that cater needs of managers and clients. The system's architecture and database schema lay a solid foundation for organized data storage, retrieval, and manipulation, ensuring data integrity and consistency.

Key features of EquipmentSYS include robust equipment and client management, efficient handling of the rental process, and the generation of administrative reports. Whether it's adding new equipment categories, updating client information, recording rental transactions, or producing yearly revenue reports, EquipmentSYS provides a comprehensive toolkit to make all the functions effective.

The EquipmentSYS combines cutting-edge technology, good design, and a commitment to meeting user needs, making it an asset for organizations with their management practices.

# Functional Components

This section presents the functional components of the Equipment Rental System.

# User Requirements

This section describes the user requirements (functional components) as high-level abstract statements.

## Manage Equipment

* + 1. EquipmentSYS will add an equipment category.
    2. EquipmentSYS will update an equipment category.
    3. EquipmentSYS will add an equipment.
    4. EquipmentSYS will change status of an equipment.

## Manage Clients

* + 1. EquipmentSYS will register a new client.
    2. EquipmentSYS will update a client’s information.
    3. EquipmentSYS will de-register or restore clients.

## Process Rentals

* + 1. EquipmentSYS will make a rental.
    2. EquipmentSYS will cancel a rental.
    3. EquipmentSYS will record collection of equipment.
    4. EquipmentSYS will record return of equipment.

## Perform Admin

* + 1. EquipmentSYS will produce a yearly equipment category profit analysis.
    2. EquipmentSYS will produce a yearly revenue analysis.

# System Requirements

The EquipmentSYS contains these top-level modules: Manage Equipment, Manage Clients, Process Rentals and Perform Admin.

Manage Equipment is a top-level module which is created to allow manager to manage categories: Add Equipment Category, Update Equipment Category; and equipment: Add Equipment, Update Equipment Status. This top-level module saves the necessary data in the Category file and Equipment file.

Manage Clients is another top-level module that allows clients to register and manage their accounts in the system. It contains these sub-modules: Register Client, Update Client and De-register Client. This part of software validates the data entered by the client and saves it in Clients file.

Process Rentals is a top-level module that allow client to place a rental or cancel it. It retrieves data from Clients file and Equipment file to validate it and save in Rentals file and RentalItems file after the rental proceed. The module also contains function allowing the receptionist to record collection and return dates and to calculate the price the client must pay for the rental.

Perform Admin is the last top-level module of the EquipmentSYS that is created to make the yearly revenue and equipment category profit reports available after the user (manager) request. It validates the data from the Rentals file and Categories file to display profit during the year specified by manager.

## System Level Use Case Diagram

The following system level use case diagram illustrates the high-level system requirements.

Manager

Client

Receptionist

## Manage Equipment

This module provides functions to add equipment category, update equipment category, add pieces of equipment and update equipment details. Manage Equipment module includes all the processes connected with equipment.

### Add Equipment Category

Add Equipment Category is a component of the EquipmentSYS that enables manager to create equipment types. This function is created to make the organization and management of equipment easier. It makes it easier for managers to identify, classify, and track various pieces of equipment.

Manager

<<includes>>

<<extends>>

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Add Equipment Category | |
| **Use Case Id** | ES001 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function adds new equipment category to the database for management purposes. | |
| **Preconditions** | * The actor must have access to the equipment database. * The actor must have the necessary information about the new equipment category to be added. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Manager** | **System** |
| **Manager adds Equipment Category to the database** | **Step 1:** Manager invokes Add Equipment Category function.  **Step 3:** Manager enters the required data:   * CategoryID Char(2) * Description Char(30) * Name Char(15)   **Step 6:** Confirm OK. | **Step 2:** Display UI.  **Step 4:** Validate all the data entered:   * All fields must be filled. * CategoryID must be unique and contain 2 capital letters. * Description must not be numeric. * Name must not be numeric.   **Step 5:** Show confirmation message.  **Step 7:** Save data in the **Categories file**:   * CategoryID * Description * Name   **Step 8:** Display a confirmation message.  **Step 9:** Reset UI. |
| **Alternate Scenarios** | **Manager** | **System** |
| **Invalid data Input.** |  | **Step 4:** Invalid data detected.  **Step 5:** Display an appropriate error message.  **Step 6:** Return to **Step 3.** |
| **Conclusions** | Category has been added to the system. | |
| **Post conditions** | The Category can now be assigned to the equipment. | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

### Update Equipment Category

Update Equipment Category function is a component of the EquipmentSYS that allows manager to update existing equipment categories. This function is created to maintain the categorization of equipment, ensuring that the category remains accurate and up to date.

Manager

<<includes>>

<<extends>>

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Update Equipment Category | |
| **Use Case Id** | ES002 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function updates an equipment category. | |
| **Preconditions** | * The actor must have access to the equipment database. * The actor must have the necessary information about the category needs to be updated. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Manager** | **System** |
|  | **Step 1:** Manager invokes Update Equipment Category function.        **Step 4:** Select Category to update.        **Step 6:** Update the required data:   * Description Char(30) * Name   Char(15) | **Step 2:** Retrieve CategoryID and Name from the **Categories file**, load on UI.    **Step 3:** Display UI.    **Step 5:** Retrieve all details for the selected Category from the **Categories file:**   * Description * Name     **Step 7:** Validate Data:   * All fields must be entered. * Description must not be numeric . * Name must not be numeric.     **Step 8:** Save Equipment Category details in **Categories file**:   * Description * Name     **Step 9:** Display Confirmation message.    **Step 10:** Reset UI. |
| **Alternate Scenarios** | **Manager** | **System** |
| **Categories file missing or empty** |  | **Step 2: Categories file** not found or no data.    **Step 3:** Display message “Categories file missing or empty”.    **Step 4:** Return to main menu. |
| **Invalid Data Input** |  | **Step 7:** Invalid data found.    **Step 8:** Display an appropriate error message.    **Step 9:** Return to **Step 6**. |
| **Conclusions** | Equipment Category details have been updated. | |
| **Post conditions** | New Equipment Category details can now be used. | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

### Add Equipment

Add Equipment function is a component of the EquipmentSYS that allows manager to add new equipment items. This function is created to make new equipment being visible and available for use in the EquipmentSYS.

Manager

<<includes>>

<<extends>>

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Add Equipment | |
| **Use Case Id** | ES003 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function adds new equipment to the database for management purposes. | |
| **Preconditions** | * The actor must have access to the equipment database. * The actor must have the necessary information about the new equipment to be added. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Manager** | **System** |
| **Manager adds equipment to the database** | **Step 1:** Manager invokes Add Equipment function.  **Step 4:** Manager enters the required data:   * Name Char(15) * Description Char(30) * Rate Numeric (5,2) * CategoryID Char(2) * Serial\_Number Char(10)   **Step 7:** Confirm OK. | **Step 2:** Retrieves Categories from **Categories file** and load on UI.  **Step 3:** Display UI.  **Step 5:** Validate all the data entered:   * All fields must be entered. * Description must not be numeric. * Rate must be numeric and >0. * Serial\_Number must only contain letters and digits.   **Step 6:** Show confirmation message.  **Step 8:** Unique EquipmentID is assigned.  **Step 9:** Save data in the **Equipment file**:   * EquipmentID * Name * Description * Rate * Serial\_Number * Category   **Step 10:** Display a confirmation message.  **Step 11:** Reset UI. |
| **Alternate Scenarios** | **Manager** | **System** |
| **Categories file missing or empty** |  | **Step 2: Categories file** not found or no data.    **Step 3:** Display message “Categories file missing or empty”.  **Step 4:** Return to main menu. |
| **Invalid data Input.** |  | **Step 5:** Invalid data detected.  **Step 6:** Display an appropriate error message.  **Step 7:** Return to **Step 4.** |
| **Conclusions** | Equipment has been added to the system. | |
| **Post conditions** | This equipment can now be rented. | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

### Update Equipment

Update Equipment function is a component of the EquipmentSYS that allows manager to modify equipment details. This function is created to provide real-time updates on equipment.

Manager

<<includes>>

<<extends>>

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Update Equipment | |
| **Use Case Id** | ES004 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function updates data for selected equipment. | |
| **Preconditions** | * The actor must have access to the equipment database. * The equipment status being changed must not be rented | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Manager** | **System** |
| **Manager changes status of the equipment** | **Step 1:** Manager invokes Update Equipment function.  **Step 4:** Manager selects the category of equipment.  **Step 6:** Manager selects the equipment to update.  **Step 8:** Manager updates the equipment details:   * Name Char(15) * Description Char(30) * Rate Numeric (5,3) * CategoryID Char(2) * Serial\_Number Char(10) * Equipment\_Status | **Step 2:** Retrieve a summary of category details from the **Categories file**.  **Step 3:** Display UI.  **Step 5:** Retrieve available equipment data from **Equipment file** with the category selectedand load on UI.  **Step 7:** Retrieve information for equipment selected and load on UI.  **Step 9:** Check if Equipment\_Status is NOT ‘R’.  **Step 10:** Validate Data:   * All fields must be entered. * Description must not be numeric. * Name must not be numeric. * Rate must be numeric.     **Step 11:** Save Equipment details in **Equipment file**:   * Name * Description * Rate * CategoryID * Serial\_Number * Equipment\_Status     **Step 12:** Display Confirmation message.    **Step 13:** Reset UI. |
| **Alternate Scenarios** | **Manager** | **System** |
| **Equipment file missing or empty** |  | **Step 2: Equipment file** not found or no data.    **Step 3:** Display message “Equipment file missing or empty”.  **Step 4:** Return to main menu. |
| **The equipment status being changed is rented** |  | **Step 7:** Detects that equipment is rented at that moment.  **Step 8:** Display message “Cannot update status of equipment that is rented. Please try again after return”.  **Step 9:** Return to **Step 4.** |
| **Conclusions** | Equipment details has been updated. | |
| **Post conditions** | Equipment with updated data is now available for rentals. | |
| **Business Rules** | Only available equipment can be updated. | |
| **Implementation Constraints** |  | |

## Manage Clients

This module provides functions to register client, update client’s information and de-register client. Manage Clients module includes all the processes connected with clients.

### Register Client

Register Client function is a component of the EquipmentSYS created to process the registration of new clients. This function is developed to create and maintain a record of clients for service-related purposes.

Client

<<includes>>

Receptionist

<<extends>>

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Register Client | |
| **Use Case Id** | ES005 | |
| **Priority** | High | |
| **Source** | Client | |
| **Primary Business Actor** | Receptionist | |
| **Other Participating Actors** |  | |
| **Description** | This function adds new user’s profile into the system. | |
| **Preconditions** | * User has access to the registration form. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Receptionist** | **System** |
| **Receptionist registers a client** | **Step 1:** Receptionist invokes Register Client function.  **Step 3:** Receptionist enters the required data:   * First\_Name Char(20) * Second\_Name Char(20) * Mobile\_Number Char(13) * Email\_Address Char(50) | **Step 2:** Display UI.  **Step 4:** Validate the data entered:   * All fields must be entered. * Mobile number must be numeric. * Email address should be valid according to the email setting rules, must be UNIQUE. * Name must not be numeric. * First name must not be numeric. * Second name must not be numeric.   **Step 5:** Assign a unique ClientID to the new client.  **Step 6:** Set Client\_Status to ‘A’ by default.  **Step 7:** Save data in the **Clients file**:   * ClientID * First\_Name * Second\_Name * Mobile\_Number * Email\_Address * Client\_Status.   **Step 8:** Display a confirmation message.  **Step 9:** Return to main menu. |
| **Alternate Scenarios** | **Receptionist** | **System** |
| **Invalid data Input** |  | **Step 4:** Invalid data detected.  **Step 5:** Display an appropriate error message.  **Step 6:** Return to **Step 3.** |
| **Email is already in the system** |  | **Step 4:** Data entered already exists.  **Step 5:** Display message “Account already exists”.  **Step 9:** Return to **Step 3.** |
| **Conclusions** | Client has been registered in the system. | |
| **Post conditions** | Client can now use his account to rent the equipment. | |
| **Business Rules** | Same mobile number can have different emails, but an email can have only 1 mobile number connected with it. | |
| **Implementation Constraints** |  | |

### Update Client

Update Client function is a component of the EquipmentSYS, allowing client to update the information needed for successful rental transaction and communication with receptionist or manager. This function is for keeping client information accurate and up to date.

Client

<<includes>>

Receptionist

<<extends>>

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Update Client | |
| **Use Case Id** | ES006 | |
| **Priority** | High | |
| **Source** | Client | |
| **Primary Business Actor** | Receptionist | |
| **Other Participating Actors** |  | |
| **Description** | This function updates an account of the client. | |
| **Preconditions** | * The client must have access to his account. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Receptionist** | **System** |
| **Receptionist updates client’s account information** | **Step 1:** Receptionist invokes Update Client function.  **Step 4:** Receptionist enters email address of the client.  **Step 6:** Receptionist updates the required data:   * First\_Name Char(20) * Second\_Name Char(20) * Mobile\_Number Char(13) * Email\_Address Char(50) * Password Char(20) | **Step 2:** Retrieve a summary of client’s details from the **Clients file:**   * Email Address   **Step 3:** Display UI.  **Step 5:** Retrieve a summary of information of the client with email entered from **Clients file** and load on UI.  **Step 7:** Validate all the data entered:   * All fields must be entered. * Mobile number must be numeric and may start with “+” * Email address should be valid according to the email setting rules, must be UNIQUE. * First name must not be numeric. * Second name must not be numeric.   **Step 8:** Update client’s details in the **Clients file:**   * First\_Name * Second\_Name * Mobile\_Number * Email\_Address   **Step 9:** Reset UI. |
| **Alternate Scenarios** | **Receptionist** | **System** |
| **Clients file missing or empty** |  | **Step 2: Clients file** not found or no data.  **Step 3:** Display message “Clients file is missing or has no data. Please try again later”  **Step 4:** Return to main menu. |
| **Invalid Email Input** |  | **Step 5:** Email was not found in the system.  **Step 6:** Display an appropriate error message.  **Step 7:** Return to **Step 4**. |
| **Invalid Data Input** |  | **Step 7:** Invalid data found.  **Step 8:** Display am appropriate error message.  **Step 9:** Return to **Step 6.** |
| **Conclusions** | Client’s details have been updated. | |
| **Post conditions** | Client now can use his updated account. | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

### De-register/Restore Client

De-register/Restore Client function is a component of the EquipmentSYS, allowing clients to make their user account unavailable or available again. This function is created to make managing of clients easier.

Receptionist

<<includes>>

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | De-register/Restore Client | |
| **Use Case Id** | ES007 | |
| **Priority** | High | |
| **Source** | Receptionist | |
| **Primary Business Actor** | Receptionist | |
| **Other Participating Actors** |  | |
| **Description** | This function changes client’s status. | |
| **Preconditions** | * The client must not have equipment rented with active status. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Receptionist** | **System** |
| **Receptionist de-registers client’s account** | **Step 1:** Receptionist invokes De-register Client function.  **Step 4:** Receptionist enters email of the client.  **Step 6:** Selects “De-register/Restore Client”.  **Step 8:** Confirm YES. | **Step 2:** Retrieve a summary of client’s details from the **Clients file:**   * Email Address   **Step 3:** Display UI.  **Step 5:** Retrieve a summary of information of the client with email entered from **Clients file** and load on UI.  **Step 7:** Show confirm dialog message.  **Step 9:** Check if client has current/upcoming rentals.  **Step 10:** Change Client\_Status to ‘U’/’A’ in the **Clients file**.  **Step 11:** Display confirmation message.  **Step 12:** Reset UI. |
| **Alternate Scenarios** | **Receptionist** | **System** |
| **Clients file missing or empty** |  | **Step 2: Clients file** not found or no data.  **Step 3:** Display message “Clients file is missing or has no data. Please try again later”  **Step 4:** Return to main menu. |
| **Invalid Email Input** |  | **Step 5:** Email was not found in the system.  **Step 6:** Display an appropriate error message.  **Step 7:** Return to **Step 4**. |
| **Receptionist chooses “NO” at Step 8** |  | **Step 9** Return to **Step 6.** |
| **Client has equipment rented** |  | **Step 9:** Display message “You have rented equipment now. Return the equipment first”.  **Step 10:** Return to **Step 4.** |
| **Conclusions** | Client\_Status has been changed to ‘U’ or ‘A’. | |
| **Post conditions** | Client now cannot use his account to rent the equipment or account is restored. | |
| **Business Rules** | The client whose Client\_Status being changed must not have equipment rented. | |
| **Implementation Constraints** |  | |

## Process Rentals

This module provides functions to place a rental, cancel rental, record collection and return of the equipment rented. Process Rentals module includes all the processes connected with rentals and operations with rentals.

### Place Rental

Place Rental of Equipment function is a component of the EquipmentSYS that allows clients to initiate and complete rent of equipment. This function is created to rent equipment by selecting it using UI.

Receptionist

Client

<<includes>>

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Place Rental | |
| **Use Case Id** | ES008 | |
| **Priority** | High | |
| **Source** | Client | |
| **Primary Business Actor** | Receptionist | |
| **Other Participating Actors** |  | |
| **Description** | This function places a rental into the system. | |
| **Preconditions** | * The client has access to the system. * The Rental Company has equipment available to rent. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Receptionist** | **System** |
| **Client rents the equipment** | **Step 1:** Receptionist invokes Place Rental function.  **Step 4:** Receptionist enters the required data:   * Collection\_Date * Return\_Date * Client’s Email   **Step 7:** Receptionist selects the required category.  **Step 9:** Receptionist selects equipment client needs to rent.  **Step 13:** Receptionist selects ‘CheckOut’. | **Step 2:** Retrieve categories from the **Categories file** and load on UI.  **Step 3:** Display UI.  **Step 5:** Validate the data entered:   * Collection\_Date * Return\_Date * Client’s Email   **Step 6:** Retrieve ClientID according to email entered.  **Step 8:** Retrieve summary details of the available equipment of the specified category and dates From the Equipment file and the Rentals file and load on UI:   * EquipmentID * Name * Rate   **Step 10:** Add selected equipment to the rental cart.  **Step 11:** Add Price of the selected Equipment to the cart total (Rate\*(Return\_Date-Collection\_Date))  **Step 12:** If more items are required, return to **Step 7**.  **Step 14:** Set Status to ‘R’.  **Step 15:** Save rental in the **Rentals file**:   * RentalID * ClientID * Collection\_Date * Return\_Date * Price * Status   **Step 16:** For each item rented: save data in **RentalItems file:**   * RentalID * EquipmentID   **Step 17:** Show confirmation message.  **Step 18**: Email rental confirmation to client  **Step 19:** Reset UI. |
| **Alternate Scenarios** | **Receptionist** | **System** |
| **Category file empty or missing** |  | **Step 2:** Category file is empty or missing.  **Step 3:** Display message “Unavailable at the moment, please try again later”.  **Step 4:** Return to the Main Menu. |
| **Invalid email address entered** |  | **Step 5:** Detect that email is incorrect.  **Step 6:** Display an appropriate error message.  **Step 7:** Return to **Step 4.** |
| **Conclusions** | The equipment has been rented. | |
| **Post conditions** | Client has rented the equipment. | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

**Receptionist**

**System**

Retrieve Categories

Invoke Place Rental

Display UI

Enter Collection\_Date, Return\_Date, Email

Validate the data entered

Display error message

Y

N

Valid?

Retrieve ClientID

Select Category

Retrieve Equipment

Select Equipment

Add equipment to the cart

Add Price of the equipment

Y

Require more?

N

Click “CheckOut”

Set status to “R”

Save Rental in **Rentals**

Save RentalID and EquipmentID in **RentalItems**

Y

More items?

N

Show confirmation message

Email rental confirmation

Reset UI

### Cancel Rental

Cancel Rental of Equipment function is a component of the EquipmentSYS that allows clients to terminate a previously initiated equipment rental. This function improves flexibility of the EquipmentSYS as client can cancel his rental made by accident.

Receptionist

<<extends>>

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Cancel Rental | |
| **Use Case Id** | ES009 | |
| **Priority** | High | |
| **Source** | Client | |
| **Primary Business Actor** | Receptionist | |
| **Other Participating Actors** |  | |
| **Description** | This function removes rental from the database. | |
| **Preconditions** | * The equipment must not be collected to cancel its rental. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Receptionist** | **System** |
| **Client cancels the rental** | **Step 1:** Receptionist invokes Cancel Rental function.  **Step 4:** Selects rental to cancel.  **Step 6:** Selects “Cancel”. | **Step 2:** Retrieve a summary of RentalIDs from the **Rentals file**.  **Step 3:** Display UI.  **Step 5:** Retrieve Rental details and load on UI.  **Step 7:** Check if SYSDATE is before Collection\_Date.  **Step 8:** Remove records associated with the rental from **RentalItems file.**  **Step 9:** Remove Rental from **Rentals file.**  **Step 10:** Display confirmation message.  **Step 11:** Reset UI. |
| **Alternate Scenarios** | **Receptionist** | **System** |
| **Collection\_Date is before the SYSDATE.** |  | **Step 8:** Display message “You have equipment rented now. Return the equipment first”.  **Step 9:** Return to **Step 4.** |
| **Conclusions** | Rental has been deleted. | |
| **Post conditions** | Equipment can now be rented by another client. | |
| **Business Rules** | The equipment must not be collected to cancel its rental. | |
| **Implementation Constraints** |  | |

**Receptionist**

**System**

Retrieve Rentals

Invoke Cancel Rental

Display UI

Select rental to cancel

Retrieve rental details and load on UI

Click “Cancel” button

Retrieve SYSDATE and Collection\_Date

N

Display error message

Valid?

Y

Remove records from **RentalItems**

Remove rental

Display confirmation message

Reset UI

### Record Collection

Record Collection function is a component of the EquipmentSYS that allows receptionist to record the collection date while collection is happening. The function is handy to use as the collection date is being assigned automatically after receptionist accepts that equipment has been collected.

Receptionist

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Record Collection | |
| **Use Case Id** | ES010 | |
| **Priority** | High | |
| **Source** | Receptionist | |
| **Primary Business Actor** | Receptionist | |
| **Other Participating Actors** |  | |
| **Description** | This function records date when equipment was collected by client. | |
| **Preconditions** | * The actor must have access to the system. * The equipment must be booked by client via “Place Rental” function. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Receptionist** | **System** |
| **Receptionist records collection** | **Step 1:** Receptionist invokes Record Return function.  **Step 4:** Enters the RentalID with equipment being collected.  **Step 6:** Selects “Record Collection”. | **Step 2:** Retrieves a summary of information from the **Rentals file** and load on UI.  **Step 3:** Display UI.  **Step 5:** Retrieve summary of information with RentalID and Status ‘R’ entered from **Rentals file**, list summary of information of equipment rented from **RentalItems file** and **Equipment file** and load on UI.  **Step 7:** Update the data in the **Rentals file:**   * Status=’C’   **Step 8:** Display a confirmation message.  **Step 9:** Reset UI. |
| **Alternate Scenarios** | **Receptionist** | **System** |
| **Rentals file missing or empty** |  | **Step 2: Rentals file** not found or no data.    **Step 3:** Display message “Rentals file missing or empty”.    **Step 4:** Return to main menu. |
| **RentalID entered is incorrect** |  | **Step 5:** RentalID entered is incorrect.  **Step 6:** Display an Error message.  **Step 7:** Reset UI. |
| **Status of the Rental is not ‘R’.** |  | **Step 5:** The rental being searched is not ‘R’.  **Step 6:** Display an Error message.  **Step 7:** Reset UI. |
| **Conclusions** | Rental Status is now “C” (Collected). | |
| **Post conditions** | Equipment now collected by client. | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

### Record Return

Record Return function is a component of the EquipmentSYS that allows receptionist to record the return date and calculate the rental price. The function is created to allow the receptionist to make changes to the Rentals file, adding return date and rental price.

Receptionist

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Record Collection | |
| **Use Case Id** | ES011 | |
| **Priority** | High | |
| **Source** | Receptionist | |
| **Primary Business Actor** | Receptionist | |
| **Other Participating Actors** |  | |
| **Description** | This function records date when equipment was returned by client. | |
| **Preconditions** | * The actor must have access to the system. * The equipment must be collected before being returned. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Receptionist** | **System** |
| **Receptionist records return** | **Step 1:** Receptionist invokes Record Return function.  **Step 4:** Selects the rental with equipment being returned.  **Step 6:** Selects the Equipment being returned.  **Step 7:** Selects “Record Return”. | **Step 2:** Retrieves a summary of information from the **Rentals file** and load on UI.  **Step 3:** Display UI.  **Step 5:** Retrieves a summary of information from **RentalItems file** and **Equipment file** and load on UI.  **Step 8:** Calculates Price\_Per\_Eq:   * If Actual\_Return\_Date is after Return\_Date: Price\_Per\_Eq = Rate\*(Actual\_Return\_Date – Return\_Date) + Rate \* Return\_Date * Else: Price\_Per\_Eq = Rate \* Return\_Date   **Step 9:** Save the data in the **Rentals file:**   * Status=’N’ (If all the equipment is returned) * Price+= Rate\*(Actual\_Return\_Date – Return\_Date)   **Step 10:** Save the data in the **RentalItems file:**   * Actual\_Return\_Date * Price\_Per\_Eq   **Step 11:** Display a confirmation message.  **Step 12:** Reset UI. |
| **Alternate Scenarios** | **Receptionist** | **System** |
| **Rentals file missing or empty** |  | **Step 2: Rentals file** not found or no data.    **Step 3:** Display message “Rentals file missing or empty”.    **Step 4:** Return to main menu. |
| **Conclusions** | Return has been recorded. | |
| **Post conditions** | Equipment is available for rent and rental Status is Not Active (‘N’). | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

## Perform Admin

This module provides functions to perform admin and get all the necessary data sorted and displayed to manager.

### Analyse Category Profit

Analyse Category Profit function is a component of the EquipmentSYS that displays profit from each equipment category during the year given by Manager.

Manager

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Analyse Category Profit | |
| **Use Case Id** | ES012 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function displays income from each equipment category within a year entered by the manager. | |
| **Preconditions** | * The actor must have access to the equipment database. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Manager** | **System** |
| **Manager analysis of profit per equipment category** | **Step 1:** Manager invokes Analyse Category Profit function.  **Step 3:** Selects the year.  **Step 5:** Selects “OK”. | **Step 2:** Display UI.  **Step 4:** Retrieve summary of rental details where Return\_Date is in year given from the **Rentals file,** Categories from **Categories file,** group by CategoryIDand load on UI.  **Step 6:** Reset UI. |
| **Alternate Scenarios** | **Manager** | **System** |
| **Rentals file missing or empty** |  | **Step 4: Rentals file** not found or no data.    **Step 5:** Display message “Rentals file missing or empty”.    **Step 6:** Return to main menu. |
| **Categories file missing or empty** |  | **Step 4: Categories file** not found or no data.    **Step 5:** Display message “Categories file missing or empty”.    **Step 6:** Return to main menu. |
| **Conclusions** | Profits from each equipment category have been listed. | |
| **Post conditions** | Profits can now be analysed by the manager. | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

|  |  |
| --- | --- |
| Year: 2024 | |
| CategoryID and Name | Income |
| CM - Camera | 2400.00 |
| LS - Lense | 0 |
| MC - Microphone | 1200.00 |

### Yearly Revenue Analysis

Yearly Revenue Analysis function is a component of the EquipmentSYS, allowing manager to see and analyse the profit for a given year. The function creates table that displays the profit for each month in the year.

Manager

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Yearly Revenue Analysis | |
| **Use Case Id** | ES013 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function displays income for each month within a year entered by the manager. | |
| **Preconditions** | * The actor must have access to the equipment database. | |
| **Trigger** | User request. | |
| **Expected Scenario** | **Manager** | **System** |
| **Manager requests yearly revenue analysis for date** | **Step 1:** Manager invokes Yearly Revenue Analysis function.  **Step 3:** Selects the year.  **Step 6:** Selects “OK”. | **Step 2:** Display UI.  **Step 4:** Retrieve summary of rental details where Return\_Date is in year given from **Rentals file**.  **Step 5:** Group Price of all rentals by month, load a chart and load on UI.  **Step 7:** Reset UI. |
| **Alternate Scenarios** | **Manager** | **System** |
| **Rentals file missing or empty** |  | **Step 4: Rentals file** not found or no data.    **Step 5:** Display message “Rentals file missing or empty”.    **Step 6:** Return to main menu. |
| **Conclusions** | Rental Prices has been listed. | |
| **Post conditions** | Yearly Revenue is now available for analysis. | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

# System Model

The following dataflow diagrams have been produced for the EquipmentSYS:

**Data Flow Diagrams – Elements:**

***External Entities:***

Client  
***Data Stores:***

D1: Categories file

D2: Equipment file

D3: Clients file

D4: Rentals file

D5: RentalItems file  
***Processes:***

P1: Manage Equipment

P1.1: Add Equipment Category

P1.2: Update Equipment Category

P1.3: Add Equipment

P1.4: Update Equipment

P2: Manage Clients

P2.1: Register Client

P2.2: Update Client

P2.3: De-register/Restore Client

P3: Process Rentals

P3.1: Place Rental

P3.2: Cancel Rental

P3.3: Record Collection

P3.4: Record Return

P4: Perform Admin

P4.1: Analyse Category Profit

P4.2: Yearly Revenue Analysis

## Level-0 DFD

Rental Request

Equipment Rental System

Client

Equipment

## Level-1 DFD

Client

Rental Request

Clients file

D3

P3

Client

Process Rentals

ClientID

Client Details

Client Details

P2

Manage Clients

P1

EquipmentID and RentalID

Rental Details

Equipment Details

Manage Equipment

RentalItems file

D5

Equipment Details

Rentals file

D4

Category Details

Category Details

Rental Details

Equipment file

D2

RentalItems Details

P4

Perform Admin

Categories file

D1

Category Details

## Level-2 DFD (Process P1: Manage Equipment)

P1.3

P1.1

Equipment Details

Add Equipment

Add Equipment Category

Equipment file

Category Details

Category Details

D2

Equipment Details

Category Details

Equipment Details

Categories file

D1

P1.2

P1.4

Update Equipment

Update Equipment Category

Category Details

## Level-2 DFD (Process P2: Manage Clients)

Client Details

Client Details

P2.1

Register Client

Client Details

P2.2

Client Details

Client Details

Clients file

D3

Client

Update Client

Client Details

P2.3

De-register Client

Client Status

## Level-2 DFD (Process P3: Process Rentals)

RentalID and EquipmentID

Client

P3.2

P3.1

D1

Categories file

Cancel Rental

Place Rental

Rental Details

Rental Details

Category Details

Email

Delete Rental

D3

Clients file

Delete Rental

ClientID

Rental Details

Equipment Details

D2

Equipment file

D5

RentalItems file

Actual\_Return\_Date, Price\_Per\_Eq

Rental Details

P3.3

Rentals file

D4

Rental Details

Rental Status

P3.4

Record Collection

Rental Status

Record Return

RentalID

RentalID

Client

Equipment Details

## Level-2 DFD (Process P4: Perform Admin)

D1

Categories file

P4.2

P4.1

Analyse Category Profit

Category Details

Yearly Revenue Analysis

Rental Details

Rental Details

EquipmentID

D4

Rentals file

RentalItems file

D5

# Data Model (Class Diagram)

The EquipmentSYS database schema is created for managing equipment, clients, and rental transactions. The schema is designed to organize information into relations, each created to store specific data related to equipment categories, pieces of equipment, clients, rentals, and rental items.

## Class Diagram

Equipment

Categories

Encompasses

1

-EquipmentID\*: int

-CategoryID\*: char

-Name: String

-Name: String

0..\* \*

-Description: String

-Description: String

-Rate: double

-Serial\_Number: String

1

RentalItems

-Actual\_Return\_Date: Date

1..\* \*

-Price\_Per\_Eq: double

Includes

1

Rentals

Clients

-RentalID\*: int

-ClientID\*: int

Places

1

-Collection\_Date: Date

-First\_Name: String

-Second\_Name: int

0..\* \*

-Return\_Date: Date

-Price: double

-Mobile\_Number: String

-Status: char

-Email\_Address: String

-Password: String

## Relational Schema

*Categories* (CategoryID, Name, Description)

*Equipment* (EquipmentID, Name, Description, Rate, CategoryID, Serial\_Number, CategoryID)

*Clients* (ClientID, First\_Name, Second\_Name, Mobile\_Number, Email\_Address, Client\_Status)

*Rentals* (RentalID, Collection\_Date, Return\_Date, Price, Rental\_Status, ClientID)

*RentalItems* (RentalID, EquipmentID, Actual\_Return\_Date, Price\_Per\_Eq)

## Database Schema

Schema: EquipmentSYS

Relation: Categories

Attributes:

CategoryID Char(2),

Description String NOT NULL,

Name String NOT NULL

Primary Key: CategoryID

Relation: Equipment

Attributes:

EquipmentID int(4),

CategoryID Char(2),

Name String NOT NULL,

Description String NOT NULL,

Rate double NOT NULL,

Serial\_Number String NOT NULL

Primary Key: EquipmentID

Foreign Key: CategoryID references Categories (CategoryID)

Relation: Clients

Attributes:

ClientID int(6)

First\_Name String NOT NULL,

Second\_Name String NOT NULL,

Mobile\_Number String NOT NULL,

Email\_Address String NOT NULL UNIQUE,

Client\_Status Char(1)

Primary Key: ClientID

Relation: Rentals

Attributes:

RentalID int(6),

ClientID int(6),

Collection\_Date Date,

Return\_Date Date,

Price double NOT NULL,

Status Char(1)

Primary Key: RentalID

Foreign Key: ClientID references Clients (ClientID)

Relation: RentalItems

Attributes:

RentalID int(6),

EquipmentID int(4),

Actual\_Return\_Date Date,

Price\_Per\_Eq double

Primary Key: RentalID, EquipmentID

Foreign Key: RentalID references Rentals (RentalID)

Foreign Key: EquipmentID references Equipment (EquipmentID)

# Conclusion

In conclusion, the EquipmentSYS stands as management system tailored to meet the needs of organizations dealing with equipment management, client interactions, and rental processes. The user requirements underscore the system's commitment to providing a comprehensive suite of features, ranging from equipment and client management to handling rental transactions and generating administrative reports.

The EquipmentSYS has thirteen functions provided. Four functions are developed for equipment management, three for interactions with clients and their profiles, another four for processing rentals and two for administrative purposes including yearly revenue analysis and analysis of profit made by each category of equipment.

DFD diagrams and system model can easily display that the EquipmentSYS uses five files for storing all the data and which data is stored and retrieved during each process.

The relational schema and database schema form the backbone of the system. They are defining the structure of the database and the relationships between various entities. This design ensures not only the efficient storage and retrieval of information but also promotes data integrity and consistency across the entire system.

The EquipmentSYS is going to become an integral tool for organizations looking to streamline their operations, reduce administrative overhead, and gain valuable insights into their equipment-related activities. The combination of user-centric design, a robust relational database model, and a comprehensive set of features positions EquipmentSYS as a good asset for organizations looking for excellence in their management needs.