Requirements Engineering

ToolSYS

Submitted By: Luke Foley (T00224345)

Computing with Software Development

Date Submitted: 21/04/2023

**Table of Contents**

[1. Introduction/overview 4](#_Toc121496336)

[2. Functional Components 5](#_Toc121496337)

[3. User Requirements 6](#_Toc121496338)

[3.1. ToolSYS will manage Rates 6](#_Toc121496339)

[3.2. ToolSYS will manage Tools 6](#_Toc121496340)

[3.3. ToolSYS will process Rentals 6](#_Toc121496341)

[3.4. ToolSYS will perform administrative reporting 6](#_Toc121496342)

[4. System Requirements 7](#_Toc121496343)

[4.1. System Level Use Case Diagram 7](#_Toc121496344)

[4.2. Manage Rates 8](#_Toc121496345)

[4.4.3 Set Tool Category 8](#_Toc121496346)

[4.4.3 Update Tool Rate 10](#_Toc121496347)

[4.3. Manage Tools 12](#_Toc121496348)

[4.4.3 Add Tool 12](#_Toc121496349)

[4.4.3 Update Tool 14](#_Toc121496350)

[4.4.3 Remove Tool 16](#_Toc121496351)

[4.4.3 View Tools 18](#_Toc121496352)

[4.4. Process Rentals 20](#_Toc121496353)

[4.4.3 Rent Tool 20](#_Toc121496354)

[4.4.3 Return Tool 22](#_Toc121496355)

[4.4.3 View Rentals 24](#_Toc121496356)

[4.5 Perform Admin Reporting 26](#_Toc121496357)

[4.5.1 Annual Revenue Analysis 26](#_Toc121496358)

[4.5.2 Annual Tool Category Analysis 27](#_Toc121496359)

[5. System Model 29](#_Toc121496360)

[5.1. Level-0 DFD 29](#_Toc121496361)

[5.2. Level-1 DFD 29](#_Toc121496362)

[5.3. Level-2 DFD (Process P1: Manage Rates) 30](#_Toc121496363)

[5.4. Level-2 DFD (Process P2: Manage Tools) 30](#_Toc121496364)

[5.5. Level-2 DFD (Process P3: Process Rentals) 31](#_Toc121496365)

[5.6. Level-2 DFD (Process P4: Perform Admin Reporting) 31](#_Toc121496366)

[6. Data Model (Class Diagram) 32](#_Toc121496367)

[6.1. Class Diagram 32](#_Toc121496368)

[6.2. Relational Schema 33](#_Toc121496369)

[6.3. Database Schema 34](#_Toc121496370)

[7. Conclusion 35](#_Toc121496371)

[8. Appendices 36](#_Toc121496372)

[8.1. Appendix A – Annual Revenue Analysis Example 36](#_Toc121496373)

[8.2. Appendix B – Annual Tool Type Analysis Example 36](#_Toc121496374)

# Introduction/overview

This system is designed to process and keep record of, as well as analyse the renting of tools. The System consists of many functions to meet the needs of a tool hire shop.

The System allows an administrator (In this case the manager) to add, update, remove, and view the tools recorded on the system. The administrator will also be able to control various elements of their tools such as the rental rate, tool category, tool description etc.

The system also allows for the administrator to generate a revenue analysis and a tool type analysis for a specified year.

The “process rentals” functions are available for staff members so that they may record, return and view rentals.

All system data is saved to one of four data stores:

* Rates File
* Tools File
* Rentals File
* RentalItems File
* The Rates file stores the tool category, and rate as these are directly linked.
* The Tools file stores all the tools on the system, each of which contains information about the tool such at ID, Status etc…
* The Rentals file stores the rental ID and the details of the associated customer
* The RentalItems file stores all the tools that are linked to a rental ID, as well as dates and rate totals.

# Functional Components

# User Requirements

## ToolSYS will manage Rates

* + 1. ToolSYS will set a Tool Category
    2. ToolSYS will update Tool Rate

## ToolSYS will manage Tools

* + 1. ToolSYS will add Tools
    2. ToolSYS will update Tools
    3. ToolSYS will remove Tools
    4. ToolSYS will view Tools

## ToolSYS will manage Customers

* + 1. ToolSYS will add Customers
    2. ToolSYS will update Customers
    3. ToolSYS will view Customers

## ToolSYS will process Rentals

* + 1. ToolSYS will process Tool Rentals
    2. ToolSYS will process Tool Returns
    3. ToolSYS will view Rentals

## ToolSYS will perform administrative reporting

* + 1. ToolSYS will produce an annual Revenue analysis
    2. ToolSYS will produce an annual Tool Category analysis

# System Requirements

## System Level Use Case Diagram

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

Customer

Manager

## Manage Rates

These functions manage the rates of the tools on the system.

### Set Tool Category

This function adds a tool category to the system

Manager

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Set Tool Category | |
| **Use Case Id** | TS1 | |
| **Priority** | High | |
| **Source** |  | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function adds a tool category to the system. This category determines the daily rate of the tool. | |
| **Preconditions** |  | |
| **Trigger** |  | |
| **Expected Scenario** | **Actor Name** | **System** |
|  | **Step 1:** Manager invokes the Set Tool Category function  **Step 3:** The Manager inputs the required data:   * Category\_Code (char) * Description (varchar) * Rate (numeric) | **Step 2:** The system displays the UI  **Step 4:** The data entered by the manager is validated by the system:   * All Fields must be entered * The category must not already exist * Rate must be numeric * Category\_Code must be char * Description must be varchar * Rate must be > 0   **Step 5:** The system saves the tool category details in the Rates file:   * Category\_Code (char) * Category\_Description (varchar) * Rate (numeric)   **Step 6:** System displays a confirmation message  **Step 8:** System UI is reset |
| **Alternate Scenarios** | **Actor** | **System** |
|  |  |  |
| **Conclusions** | The new tool category is added to the database | |
| **Post conditions** | Tools of this category can be added to the database | |
| **Business Rules** | The tool category must not already exist | |
| **Implementation Constraints** |  | |

### Update Tool Rate

Manager

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Update Tool Rate | |
| **Use Case Id** | TS2 | |
| **Priority** | High | |
| **Source** |  | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function updates the daily rate of an existing tool category. | |
| **Preconditions** |  | |
| **Trigger** |  | |
| **Expected Scenario** | **Actor Name** | **System** |
|  | **Step 1:** Manager invokes the Update Tool Rate function  **Step 4:** The Manager selects the tool category of which the rate is to be changed  **Step 6:** The Manager enters the required data:   * Rate (numeric) * Description(varchar) | **Step 2:** The system retrieves the existing tool categories from the Rates file  **Step 3:** System displays UI  **Step 5:** System displays UI  **Step 7:** The system validates the data entered:   * All fields must be entered * Rate must be numeric * Rate must be > 0   **Step 8:** The system updates the rate of the tool category in the Rates file   * Rate (numeric) * Description(varchar)   **Step 9:** System displays a confirmation message  **Step 10:** System UI is reset |
| **Alternate Scenarios** | **Actor** | **System** |
|  |  |  |
| **Conclusions** | The rate of an existing tool category is updated | |
| **Post conditions** | Tools rentals will be processed with this updated rate | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

## Manage Tools

These functions manage the tools on the system.

### Add Tool

Manager

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Add Tool | |
| **Use Case Id** | TS3 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function adds a tool record to the system | |
| **Preconditions** |  | |
| **Trigger** |  | |
| **Expected Scenario** | **Actor Name** | **System** |
|  | **Step 1:** The Manager invokes the Add Tool function  **Step 4:** The Manager inputs the Required data:   * Category * Description (varchar) * Manufacturer (varchar) | **Step 2:** The system retrieves the tool categories from rates file  **Step 3:** The system displays the UI  **Step 5:** The system validates the entered data:   * Category must be selected * Description must be varchar * Tool\_Manufacturer must be varchar   **Step 6:** The system assigns a new unique Tool\_ID  **Step 7:** The System sets the Tool\_Status to “I” (In)  **Step 8:** The tool details are saved in the Tools file:   * Tool\_ID * Category\_Code * Description * Manufacturer * Tool\_Status   **Step 9:** System displays appropriate confirmation message  **Step 10:** System UI is reset |
| **Alternate Scenarios** | **Actor** | **System** |
|  |  |  |
| **Conclusions** | The tool is added to the system | |
| **Post conditions** | This tool can now be rented | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

### Update Tool

Manager

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Update Tool | |
| **Use Case Id** | TS4 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function updates a tool on the system | |
| **Preconditions** |  | |
| **Trigger** |  | |
| **Expected Scenario** | **Actor Name** | **System** |
|  | **Step 1:** Manager invokes the Update Tool function  **Step 4:** The Manager enters the required data   * Tool\_ID (numeric)   **Step 7:** The Manager inputs the required data:   * Tool\_Description(varchar) | **Step 2:** The system retrieves the tool details from the Tools file  **Step 3:** The system displays the UI  **Step 5:** The system performs validation:   * Tool\_ID must be numeric * Tool\_ID must exist in the Tools file * The Tool\_Status must be “I” (In) or “U”(Unavailable)   **Step 6:** The system displays UI  **Step 8:** The system validates the entered data:   * All fields must be entered * Tool\_Description must be varchar2   **Step 9:** Tool details are updated in the Tools file  **Step 10:** System displays a confirmation message: “Tool has been updated”  **Step 11:** System UI is reset |
| **Alternate Scenarios** | **Actor** | **System** |
|  |  |  |
| **Conclusions** | The tool record has been updated | |
| **Post conditions** |  | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

### Remove Tool

Manager

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Remove Tool | |
| **Use Case Id** | TS5 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function marks a tool record as unavailable, making it unusable for renting | |
| **Preconditions** |  | |
| **Trigger** |  | |
| **Expected Scenario** | **Actor Name** | **System** |
|  | **Step 1:** The manager invokes the Remove Tool function  **Step 4:** The manager enters the required data:   * Tool\_ID (numeric) | **Step 2:** The system retrieves tool details from the Tools file  **Step 3:** The system displays UI  **Step 5:** The system validates the entered data:   * All fields must be entered * Tool\_ID must be numeric * Tool\_ID must exist in the Tools file * Tool\_ID must not currently be rented by a customer i.e. Tool\_Status must not be “O” (Out)   **Step 6:** System sets the Tool\_Status to “U” (Unavailable) in the tools file  **Step 7:** System displays confirmation message  **Step 8:** Reset UI |
| **Alternate Scenarios** | **Actor** | **System** |
|  |  | **Step 4:** The entered data does not validate  **Step 5:** Display appropriate error message  **Step 6:** Return to step 3 |
| **Conclusions** | The Tool\_Status is set to “U” (Unavailable) | |
| **Post conditions** | This tool can no longer be rented | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

### View Tools

Manager

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | View Tools | |
| **Use Case Id** | TS6 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function will list the recorded tools on the system based on the filters entered by the user | |
| **Preconditions** |  | |
| **Trigger** |  | |
| **Expected Scenario** | **Actor Name** | **System** |
|  | **Step 1:** The manager invokes the Search Tools function  **Step 4:** The manager enters the desired filters:   * Tool\_ID (numeric) * Tool\_Category * Tool\_Description (varchar) * Tool\_Manufacturer (varchar) * Tool\_Status | **Step 2:** The system retrieves the recorded tools from the tools file  **Step 3:** The system displays the UI  **Step 5:** The system validates the entered data:   * Tool\_ID must be numeric   **Step 6:** The system lists the tools within the Tools File according to the data entered by the manager  **Step 7:** System displays confirmation message  **Step 8:** Reset UI |
| **Alternate Scenarios** | **Actor** | **System** |
|  |  | **Step 5:** The entered data does not validate  **Step 6:** Display appropriate error message  **Step 7:** Return to step 3 |
| **Conclusions** | The tools have been listed according to the data entered | |
| **Post conditions** |  | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

## Process Rentals

These functions process the tools rented by customers

### Rent Tool

Staff

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Rent Tool | |
| **Use Case Id** | TS7 | |
| **Priority** | High | |
| **Source** | Staff | |
| **Primary Business Actor** | Staff | |
| **Other Participating Actors** |  | |
| **Description** | This function processes rentals | |
| **Preconditions** |  | |
| **Trigger** |  | |
| **Expected Scenario** | **Actor Name** | **System** |
|  | **Step 1:** The Staff invokes the Rent Tool function  **Step 4:** The Staff enters the customer details:   * Forename (varchar) * Surname (varchar) * Email (varchar) * Phone (numeric) * Eircode (varchar)   **Step 4:** The Staff selects the desired tools to rent by entering the required data:   * Tool\_ID (numeric) * Rent\_Date (date) * Due\_Return\_Date (date) | **Step 2:** The system retrieves the tool details from the tools file  **Step:**  The system retrieves the tool category details from the rates file  **Step 3:** The System Displays the UI  **Step 5:** The entered data is validated:   * All fields must be entered * Forename must be varchar * Surname must be varchar * Email must be varchar, and the correct format * Phone must be numeric, > 5 and < 17 * Eircode must be varchar, and the correct format   **Step 6:** A new unique Rental\_ID is assigned  **Step 7:** System displays UI  **Step**: The customer details are added to the Rentals File:   * Rental\_ID * Forename * Surname * Email * Phone * Eircode   **Step:** Each rented tool is added to the RentalItems File:   * Rental\_ID * Tool\_ID * Rent\_Date * Due\_Return\_Date * Rental\_Fee   **Step 8:** For each rented tool, the status is set to “O” (Out) in the Tools File:   * Tool\_Status   **Step:** The system sends a confirmation email to the customer  **Step 9:** System displays confirmation message  **Step 10:** Reset UI |
| **Alternate Scenarios** | **Actor** | **System** |
|  |  |  |
| **Conclusions** | The tool has been rented | |
| **Post conditions** | The tool cannot be rented by another customer until returned | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

### Return Tool

Staff

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Return Tool | |
| **Use Case Id** | TS8 | |
| **Priority** | High | |
| **Source** | Staff | |
| **Primary Business Actor** | Staff | |
| **Other Participating Actors** |  | |
| **Description** | This function processes tool returns | |
| **Preconditions** |  | |
| **Trigger** |  | |
| **Expected Scenario** | **Actor Name** | **System** |
|  | **Step 1:** The Staff invokes the Return Tool function  **Step 4:** The Staff enters the required data:   * Rental\_ID (numeric)   **Step 9:** The Staff selects tools to be returned by entering the required Data:   * Tool\_ID | **Step 2:** The system retrieves rental details from the Rentals file  **Step 3:** The System Displays the UI  **Step 5:** The system performs validation:   * Rental\_ID must be numeric * Rental\_ID must exist in the Rentals file   **Step 6:** The system retrieves details from RentalItems file  **Step 7:** The system retrieves tool details from Tools file  **Step 8:** The system displays the UI  **Step 10:** The entered data is validated:   * Tool\_ID must be numeric * Tool\_ID must exist in the RentalItems File   **Step 11:** The System sets the Tool\_Status of the returned tool to “I” (In)  **Step 10:** System displays confirmation message |
| **Alternate Scenarios** | **Actor** | **System** |
|  |  |  |
| **Conclusions** | The Tool has been returned | |
| **Post conditions** | The Tool can now be rented by another customer | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

### View Rentals

Staff

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | View Rentals | |
| **Use Case Id** | TS9 | |
| **Priority** | High | |
| **Source** | Staff | |
| **Primary Business Actor** | Staff | |
| **Other Participating Actors** |  | |
| **Description** | This function displays all Rentals recorded on the system | |
| **Preconditions** |  | |
| **Trigger** |  | |
| **Expected Scenario** | **Actor Name** | **System** |
|  | **Step 1:** The Staff invokes the View Rentals function  **Step 5:** The Staff enters the required details:   * Rental\_ID (numeric) | **Step 2:** The system retrieves rental details from the Rentals file  **Step 3:** The system retrieves the rental item details from the RentalItems File  **Step 4:** The System Displays the UI  **Step 5:** The system performs validation:   * Rental\_ID must be numeric * Rental\_ID must exist in the Rentals file   **Step 6:** The system displays all the rental items that match the Rental\_ID entered |
| **Alternate Scenarios** | **Actor** | **System** |
|  |  |  |
| **Conclusions** | The Rental details and Rental items can be viewed | |
| **Post conditions** |  | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

## Perform Admin Reporting

These functions perform administrative reporting on the tool rentals data

### Annual Revenue Analysis

Manager

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Perform annual revenue analysis | |
| **Use Case Id** | TS10 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function generates a revenue analysis for a selected year | |
| **Preconditions** |  | |
| **Trigger** |  | |
| **Expected Scenario** | **Actor Name** | **System** |
|  | **Step 1:** The manager invokes the perform annual revenue analysis function  **Step 4:** The manager enters the desired year for the revenue analysis | **Step 2**: The system retrieves details from RentalItems file  **Step 3:** The system displays UI  **Step 5:** The system generates the revenue analysis of the selected year  **Step 6:** The system displays the annual revenue analysis  **Step 7:** System UI is reset |
| **Alternate Scenarios** | **Actor** | **System** |
|  |  |  |
| **Conclusions** | The annual revenue analysis is generated and displayed | |
| **Post conditions** |  | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

### Annual Tool Category Analysis

Manager

|  |  |  |
| --- | --- | --- |
| **Use Case Name** | Perform annual revenue analysis | |
| **Use Case Id** | TS11 | |
| **Priority** | High | |
| **Source** | Manager | |
| **Primary Business Actor** | Manager | |
| **Other Participating Actors** |  | |
| **Description** | This function generates a tool category analysis for a selected year | |
| **Preconditions** |  | |
| **Trigger** |  | |
| **Expected Scenario** | **Actor Name** | **System** |
|  | **Step 1:** The manager invokes the perform annual tool category analysis function  **Step 4:** The manager enters the desired year for the tool category analysis | **Step 2**: The system retrieves details from RentalItems file  **Step 3:** The system displays UI  **Step 5:** The system generates a tool category analysis for the most popular tool category in the selected year  **Step 6:** The system displays the annual tool category analysis  **Step 7:** System UI is reset |
| **Alternate Scenarios** | **Actor** | **System** |
|  |  |  |
| **Conclusions** | The annual revenue analysis is generated and displayed | |
| **Post conditions** |  | |
| **Business Rules** |  | |
| **Implementation Constraints** |  | |

# System Model

The following dataflow diagrams have been produced for the system:

## Level-0 DFD

Rental details

Admin reports

Staff

ToolSYS

Manager

Confirmation

Tool details

## Level-1 DFD

Staff

D3

Rentals file

D1

Rates file

P3

Process Rentals

P2

Manage Tools

P1

Manage Rates

D2

Tools file

D4

RentalItems file

P4

Perform Admin

Manager

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

Rate Details

ss

P1.2

Update Tool Rate

P1.1

Set Tool Category

D1

Rates file

Rate Details

ss

Rate Details

ss

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

P2.4

View Tools

D1

Rates file

Tool Details

ss

Rate Details

ss

Tool Details

ss

P2.1

Add Tool

D2

Tools file

Tool Details

ss

Tool Details

ss

Tool Details

ss

Tool Details

ss

P2.3

Remove Tool

P2.2

Update Tool

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

D2

Tools file

D1

Rates file

Rate Details

ss

Tool Details

ss

P3.3

View Rentals

Rental Details

ss

P3.1

Rent Tools

Rental Details

ss

Item Details

ss

RentalItem Details

ss

D4

RentalItems file

D3

Rentals file

RentalItem Details

ss

D4

RentalItems file

Rental Details

ss

RentalItem Details

ss

Rental Details

ss

P3.2

Return Tools

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

Rental Item Details

ss

P4.1

Generate Annual Revenue Analysis

P4.2

Generate Annual Tool Type Analysis

D4

RentalItems file

Rental Item Details

ss

# Data Model (Class Diagram)

Brief introduction….

## Class Diagram

|  |
| --- |
| Rentals |
| Rental\_ID\*: int  Forename: String  Surname: String  Email: String  Phone: int  Eircode: String |

|  |
| --- |
| RentalItems |
| Rental\_ID: int  Tool\_ID: int  Rent\_Date: date  Due\_Return\_Date: date  Rental\_Fee: Decimal |

1..\*

1

1..\*

|  |
| --- |
| Tools |
| Tool\_ID\*: int  Category\_Code: String  Tool\_Description: String  Tool\_Manufacturer: String  Tool\_Status: Char  1..\* |

1

1

|  |
| --- |
| Rates |
| Category\_Code\*: String  Category\_Description: String  Rate: Decimal |

## Relational Schema

Rates (CategoryCode, Description, Rate)

Tools (ToolID, CategoryCode, ToolDescription, ToolManufacturer, ToolStatus)

Customers(CustomerID, Forename, Surname, Email, Phone, Eircode)

Rentals (RentalID, CustomerID, TransactionDate, TotalFee)

RentalItems (RentalID, ToolID, RentDate, Due\_ReturnDate, RentalFee)

## Database Schema

A definition of the database to be implemented.

This includes primary key, foreign key and other constraints to be implemented.

**Relation Rates**

**Attributes:**

CategoryCode: Char (2) NOT NULL

Description: String (20) NOT NULL

Rate: Decimal (5,2) NOT NULL

**Primary Key:** CategoryCode

**Relation Tools**

**Attributes:**

ToolID: Integer (4) NOT NULL

CategoryCode: Char (2) NOT NULL

ToolDescription: String (50) NOT NULL

ToolManufacturer: String (50) NOT NULL

ToolStatus: Char (1) NOT NULL

**Primary Key**: ToolID

**Foreign Key**: Category\_Code **REFERENCES** Rates(Category\_Code)

**Relation Customers**

**Attributes:**

CustomerID: Integer (4) NOT NULL

Forename: String (40) NOT NULL

Surname: String (40) NOT NULL

Email: String (300) NOT NULL

Phone: String (15) NOT NULL

Eircode: Char (7) NOT NULL

**Primary Key**: CustomerID

**Relation Rentals**

**Attributes:**

RentalID: Integer (4) NOT NULL

CustomerID: Integer (4) NOT NULL

TransactionDate: Date NOT NULL

TotalFee: Decimal (7,2) NOT NULL

**Primary Key**: RentalID

**Foreign Key**: CustomerID **REFERENCES** Customers(CustomerID)

**Relation RentalItems**

**Attributes:**

RentalID: Integer (5) NOT NULL

ToolID: Integer (4) NOT NULL

RentDate: Date NOT NULL

ReturnDate: Date NOT NULL

RentalFee: Decimal (9,2) NOT NULL

**Primary Key**: (RentalID, ToolID)

**Foreign Key**: RentalID **REFERENCES** Rentals(RentalID)

**Foreign Key**: ToolID **REFERENCES** Tools(ToolID)

# Conclusion

Creating a requirements document for the system is a crucial step in the development process. This helps to clearly define the specific functions that must be met for in order for the system to function effectively.

This document is an analysis of the various functions required for the ToolSYS tool hire system. Displaying the requirements in a clear way via use-case narratives, diagrams etc. provided a clear blueprint for how the system would potentially work. This blueprint eased the process of building a prototype system significantly.

# Appendices

## Appendix A – Annual Revenue Analysis Example

Chart, bar chart

Description automatically generated

## Appendix B – Annual Tool Type Analysis Example

Chart, bar chart

Description automatically generated