TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

THAPATHALI CAMPUS

Proposal

On

Rental system

Submitted By:

Amar Mahato (THA081BCT002)

Birendra Giri (TH081BCT005)

Bishrut Kafle (THA081BCT006)

Karun Lamsal (THA081BCT013)

Submitted To:

Department of Electronics and Computer Engineering

Thapathali Campus

Kathmandu, Nepal

February,2025

ABSTRACT

A C-based software program called the "Rent-All" Rental Management System was created to automate the rental and return procedure while keeping precise transaction logs and stock information. The goal of this project is to ensure rental firms operate efficiently by removing human error from inventory monitoring and billing. The system makes use of dynamic memory allocation for adaptable inventory management and file handling for long-term storage. The system automatically updates inventory and creates transaction bills while allowing users to rent, refund, and see available goods. Modular programming improves the project's scalability and maintainability. The system's main features are organized around the C standard libraries (stdio.h, stdlib.h, and string.h) and were created using GCC/VS Code. In addition to illustrating basic C programming ideas, this project offers a useful example of companies in the rental sector, guaranteeing precision, effectiveness, and dependability.

1. INTRODUCTION

As a part of the first semester of the Bachelor of Computer Technology (BCT), this project aims to develop a C-program used in renting places. The goal is to use the fundamental ideas of the programming language C to create a system used for management of the renting bills and the inventory of the stock available in the rental shop.

1.1 Background

In today's time when people have needed to buy anything needed for the life that is used for less time then the renting system comes in handy. As renting helps to use the equipment for a time interval. For example, you want to learn to ride a bike but you don't have one Or you have a wedding party to attend but you don't have suitable suit to wear. We have solutions for such needs. We created this program "Rent-All" with that very thing in mind. Our aim is to allow users to rent any product at reasonable price for certain period of time without having to actually purchase the actual product.

1.2 Motivation

As in the rental business, human error occurs in the management of the stocks and the bill writing of the bill for it. So, this program helps to reduce human error in the rental stock and bill management of the renting shop. This demonstrates the use of dynamic memory management and file I/O in C and creates a management of the billing system. It helps to auto-update the stock of the equipment to be rented when costumer returns the rented equipment. It helps to print the rent and returns bill of the customer.

1.3 Problem Definition

When the rental management system is done manually then there are chances of mis-management of the stock and rewritten of the stock. Also, when someone returns the rented equipment late that causes revenue loss if the interest calculated is incorrect due to the manual error. Without persistent storage, data is lost after each session.

1.4 Objectives

- To create a rental system with extensive use of file handling ability of C
- Generate transaction bills and log all rental/return operations
- Implement dynamic memory management for handling the inventory

1.5 Scope and Applications

This is assumed to offer a wide range of services to the users as they can get the products and get the bill without any error and the stock management of the rental equipment are done automated. As the program is written in C language, it allows for easier extendibility of the program. So, although it provides only a few services at present, it can be easily extended in the future as per the desire of the seller and the demand of the market.

2. LITERATURE REVIEW

Rental system has been in need for the shop-keeper as for the manual error happening in the stock management and the bill making. There are different applications created based on this problem. This literature review explores existing works related to rental management system.

2.1 Borrow Nepal

Borrow Nepal is a mobile application developed by Vajra Studio. Borrow Nepal is Nepal's pioneer application which lets user lend, earn and borrow. This app is brought out to serve the clients all over Nepal to have a single platform to lend and rent in various categories such as:

- Talent and Services - Register yourself as a contractor, errand runner, freelancer, one-day server, task completer, etc., and earn.

- Equipment - Post your equipment for other people to rent while you earn from them; post your books, kits, gear, furniture, art, etc., and earn.

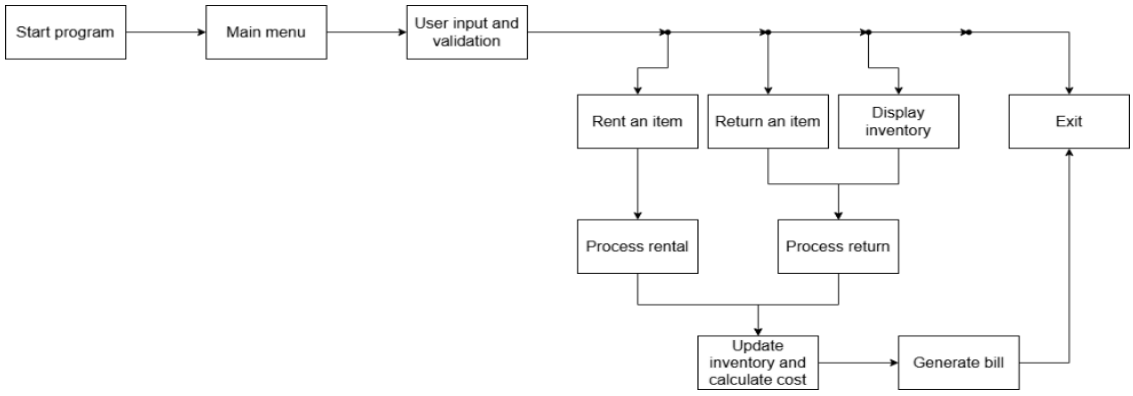
- Rooms/ Spaces - Advertise your To-let apartments, lands, and rooms for free to have a renter/tenant so that you can earn from your unused spaces.

- Automobiles - Post your ads to lend your motor/non-motor vehicles to earn, lend your bicycle, motorbikes, cars, trucks, rickshaws, and others for others to find you.

3. METHODOLOGY

3.1 Block Diagram System Architecture

The rental management system follows a modular architecture consisting of start of the program as it call inventory, main menu, user input to choose rent, return or display inventory, according to the user input it updates inventory and calculates costs, generate the bill and log transaction, save the updated inventory, return to main menu.



3.2 Parts of the program

3.2.1 Display the main menu

The system starts with calling of the program and showing the main menu of the system where users choose whether to rent, return or show the inventory.

3.2.2 User Input

The user needs to choose either rent an item or return an item or show the inventory or exit.

3.2.3 Processing Transactions

- **Renting**

When user gets into the renting then inventory will open. It follows the following process

1) Receives user input for the item, quantity, and the rental duration time period.
2) Check availability of the item updates inventory.

3) Calculates the cost and any applicable interest.
4) Generates a bill and logs the transaction.

- **Returning**

When user gets into the returning then inventory will open. It follows the following process

1) Receives user input for the item to be return and quantity to return, and rental duration.
2) Updates inventory by adding the returned quantity.
3) Calculates the cost (including interest if applicable).
4) Generates a bill and logs the transaction.

- **Displaying Inventory**

It displays the inventory what are left in the stock for the rent.

3.2.4 Exiting

When the user chooses to exit, the system saves the updated inventory back to .txt file and frees any allocated memory before terminating.

3.3 Tools and Environment

- **Programming Language:** C
- **Compiler:** GCC or any standard C compiler
- **File I/O:** For persistent storage
- **Development Environment:** Code::Blocks, VS Code

4. METHODOLOGY

4.1 Transaction process

4.1.1 Renting

When user tries to rent an equipment then, user has the first look on the stock available in the shop. User then selects the equipment and quantity and also declares the time duration of the rent keeping. The program checks the availability of the equipment and updates the inventory of the stock. The program calculates the cost of the rent and generates the bill and logs the transaction of the rent.

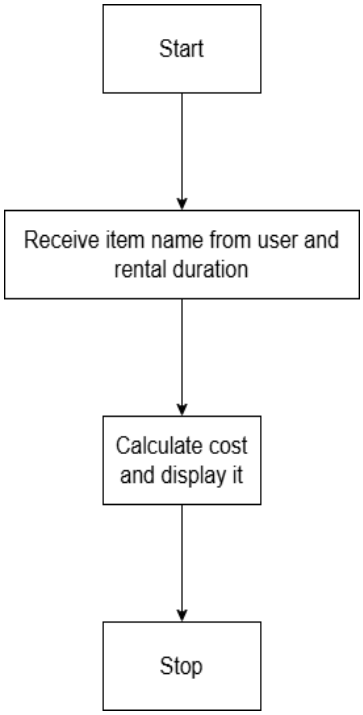


Fig:- Block Diagram Representing processes of Renting

4.1.2 Returning

When user returns the equipment then user selects the returning equipment and the quantity of the equipment to be returned also declares the time that the equipment has been taken. The program then updates the stock available. Then the program calculates the cost and adds interest

to the cost if the time exceeds the deadline and generates the bill of the return and logs the transaction of the return.

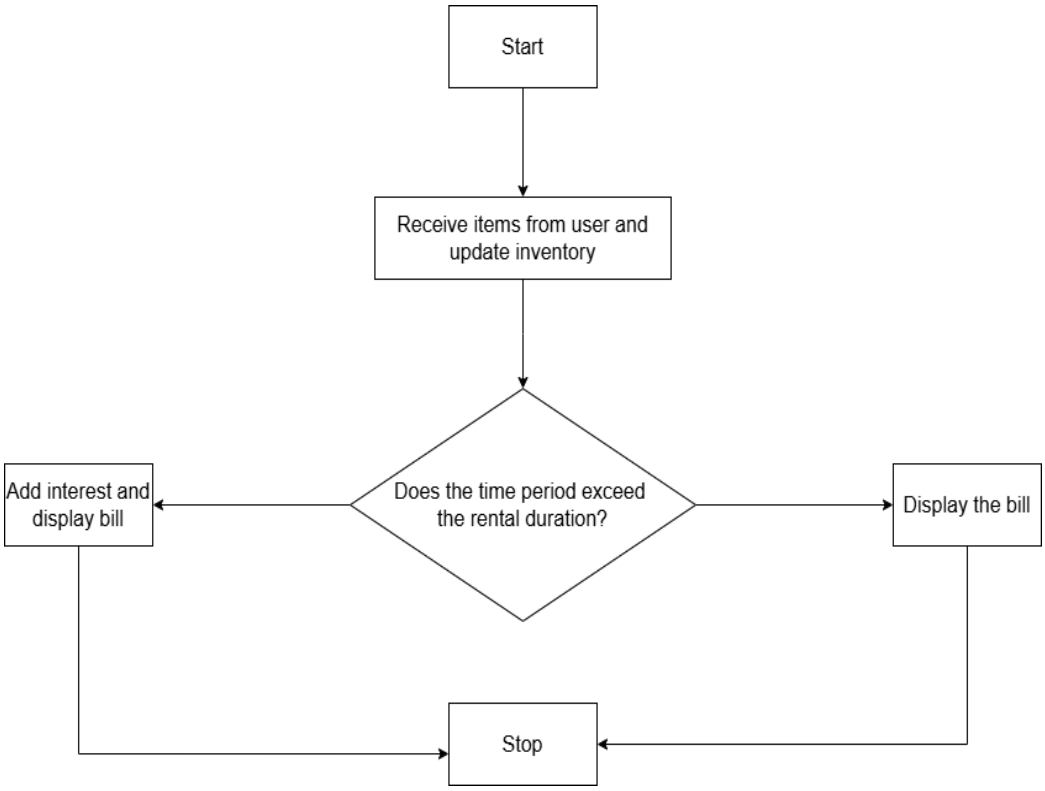


Fig:- Block diagram of processes of Returning

4.2 Header File

In C programming, header files contain declarations of functions and macros used across multiple source files. For this rental management system, the following header files are used:

i. Stdio.h
ii. Stdlib.h
iii. String.h

4.3 Function and Conditional Statements

Functions are mainly used in the modularity for better understanding of the code. The program is divided into several functions, each responsible for a specific task. Conditional statements are

used throughout the program. They are used for input validation, error handling and for the flow control. In the program if/else-if/else and switch case is used.

4.4 File Handling

File handling is an essential aspect of the system as it allows data storage and retrieval. This ensures that inventory and transaction records are preserved between program runs. The logs of the transactions stored help to get the review of the equipment which are frequently used. File handling ensures persistence in data, allowing the system to maintain a history of transactions of the equipment.

4.5 DMA

DMA is used to allocate memory for the inventory dynamically based on the number of items. This allows the program to handle a variable number of items rather than a fixed-size array. The program can adjust the size of the inventory at runtime. Memory is allocated based on actual needs rather than a predetermined fixed size.

5. SCOPE AND APPLICATIONS

This project is assumed to offer a wide range of services to the users as they can get varieties of products at reasonable cost. Hence, it is focused for use by people of all age groups. Also, as the program has the ability to keep track of customers, it is easier to manage the circulation of goods available in the warehouse. With the addition of penalty charge(if the items are used for extended period of time without updating the company), customers are likely to return the rented goods within the given duration of time.

6. TIME ESTIMATION

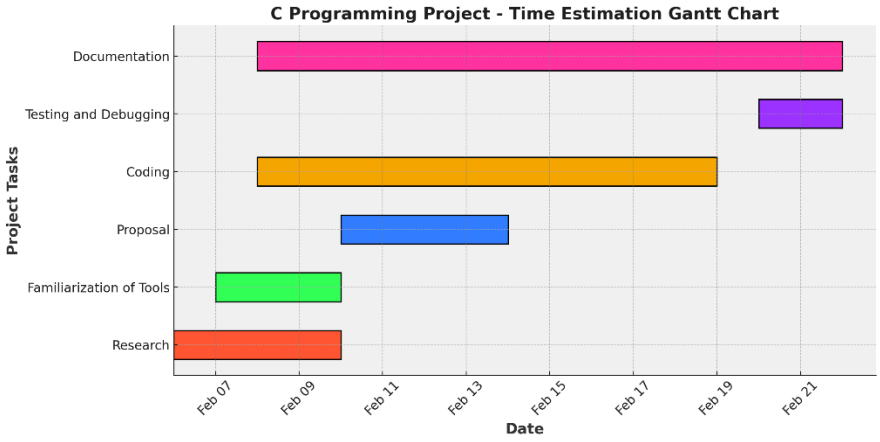


Fig:- Time Estimation Gantt Chart

7. FEASIBILITY ANALYSIS

Technical Feasibility

- **Programming Language:** C is well-suited for systems programming and provides excellent performance.
- **File I/O and Dynamic Memory:** These features ensure that the system can handle real-time data persistence and scalability.
- **Modular Design:** The separation of functionalities improves maintainability and future integration of advanced features.

Economic Feasibility

- **Low Development Cost:** Utilizes open-source tools and libraries.
- **Maintenance Cost:** Modular design reduces future maintenance costs.

Operational Feasibility

- **User-Friendliness:** A console-based interface is simple and effective.
- **Scalability:** The design allows for future enhancements such as GUI integration and network capabilities.

8. REFERENCES

1. **https://borrownepal.com**
2. **Kernighan, B.W., & Ritchie, D.M.(1988). The C Programming(2nd ed.). Prentice Hall**
3. **Stallings, W.(2018). Operating Systems: Internals and Design Principles(9th ed.). Pearson**