



PROGRAMME NAME

SOFTWARE ENGINEERING (SCIENCE COMPUTER)

COURSE NAME

CSF3023 – SYSTEM THINKING AND LOGIC (K4)

LECTURER NAME

PROFESSOR TS. DATO' DR. AZIZ DERAMAN

GROUP MEMBER

NAME	MATRIC
NUR AQILAH SYAHIRAH BINTI AHMAD TAJUDDIN	S80850
SADAF ALTAF	S81166
MUHAMMAD AMIN BIN ROSSAIDI	S81053

TABLE OF CONTENT

1.0 INTRODUCTION

1.1 Background of the System

The car rental industry has undergone a massive shift from traditional physical storefronts to digital platforms. In the past, customers had to visit a rental office, manually fill out paperwork, and rely on the staff's verbal confirmation of car availability. This process was often slow, prone to human error, and limited by business operating hours.

The Online Car Rental System is a web-based solution designed to modernize this process. It acts as a centralized platform where vehicle data, customer information, and booking schedules are integrated into a single database. This allows for real-time synchronization, ensuring that once a car is booked, it is immediately removed from the available list, preventing double-bookings.

1.2 Purpose of the System

The primary purpose of this system is to simplify the rental experience for both the service provider and the customer. Specifically, it aims to:

- **Enhance Accessibility:** Allow customers to browse and book vehicles 24/7 from any location via the internet.
- **Improve Efficiency:** Automate the booking and payment process, reducing the need for manual data entry and administrative overhead.
- **Data Accuracy:** Maintain a real-time record of vehicle availability, maintenance status, and financial transactions.
- **Customer Trust:** Provide a transparent platform where users can see clear pricing, car specifications, and manage their own booking history.

1.3 Overall Idea and Context

The main idea of the system is to build a self-service environment where customers actively manage their own rental process. Rather than simply receiving information, users take control by registering their own accounts, searching for cars that match their preferences such as budget, brand, or vehicle size and completing payments securely on their own.

In a wider perspective, this system aligns with the demands of the modern digital economy. It enables a car rental business to grow efficiently by replacing traditional, paper-based operations with a centralized, cloud-based platform. Beyond just renting vehicles, the system supports the entire rental lifecycle, from user registration, car selection and payment, to reservation management and status tracking.

2.0 FUNCTIONAL REQUIREMENT

2.1 User Registration & Login

The system allows users to create an account and securely log in using their credentials to access car rental services.

2.1.1 Pseudocode User Registration & Login

```
START

DISPLAY "Enter email"
INPUT email

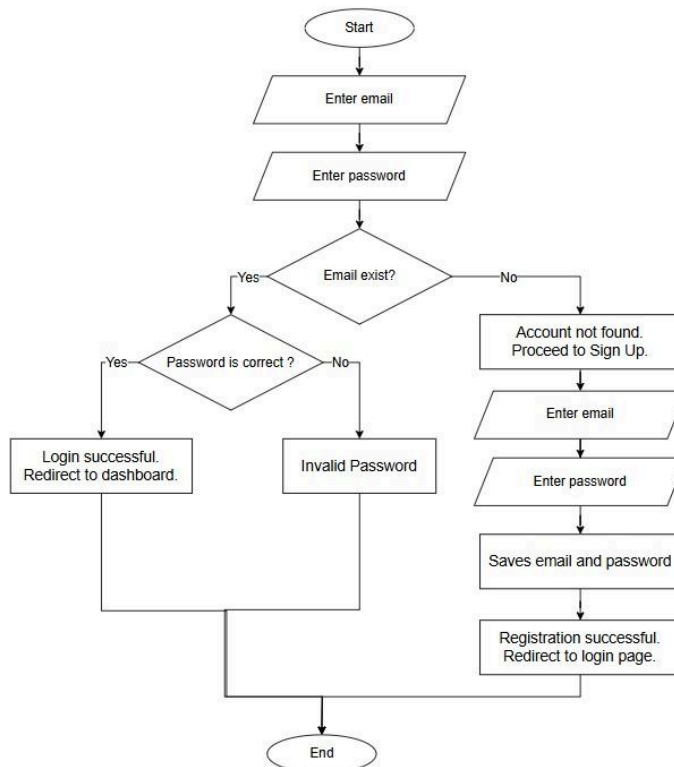
DISPLAY "Enter password"
INPUT password

IF email exists THEN
    IF password is correct THEN
        DISPLAY "Login successful"
        DISPLAY "Redirect to dashboard"
    ELSE
        DISPLAY "Invalid password"
    ENDIF
ELSE
    DISPLAY "Account not found"
    DISPLAY "Proceed to Sign Up"

    SAVE email, password
    DISPLAY "Registration successful"
    DISPLAY "Redirect to login page"
ENDIF

END
```

2.1.2 User Registration & Login Flowchart



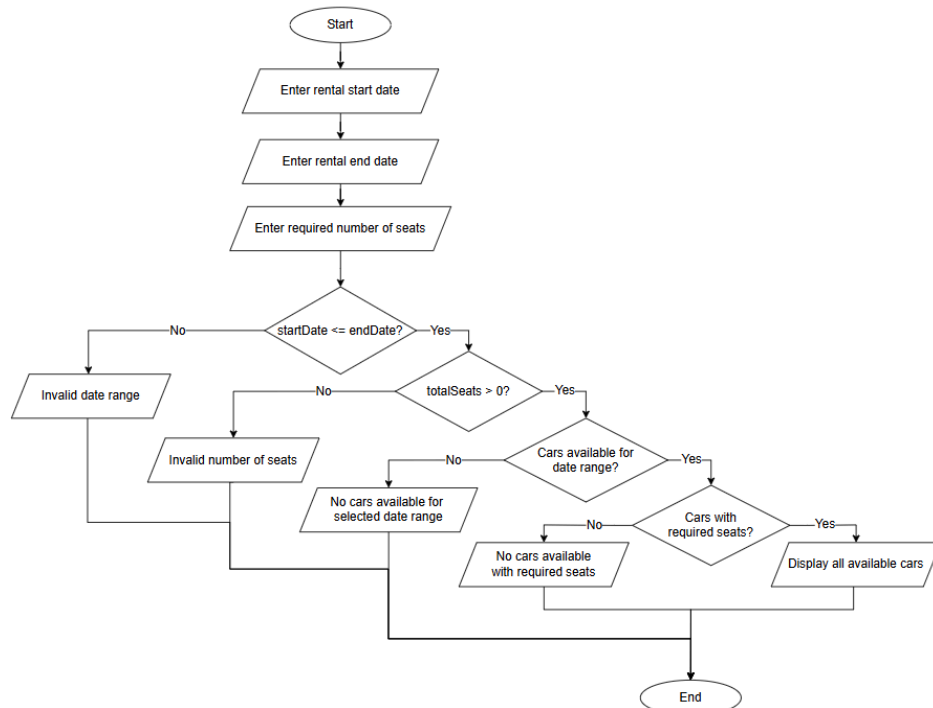
2.2 View Available Cars

The system displays a list of available cars, including details such as brand, price, and availability, to help users make informed choices.

2.2.1 View Available Cars Pseudocode

```
START  
  
DISPLAY "Enter rental start date"  
INPUT startDate  
  
DISPLAY "Enter rental end date"  
INPUT endDate  
  
DISPLAY "Enter required number of seats"  
INPUT totalSeats  
  
IF startDate <= endDate THEN  
    IF totalSeats > 0 THEN  
        IF there are cars available for the selected date range THEN  
            IF there are cars available with required number of seats THEN  
                DISPLAY "Available cars will be displayed here"  
            ELSE  
                DISPLAY "No cars available with required seats"  
            ENDIF  
        ELSE  
            DISPLAY "No cars available for selected date range"  
        ENDIF  
    ELSE  
        DISPLAY "Invalid number of seats"  
    ENDIF  
ELSE  
    DISPLAY "Invalid date range"  
ENDIF  
  
END
```

2.2.2 View Available Cars Flowchart



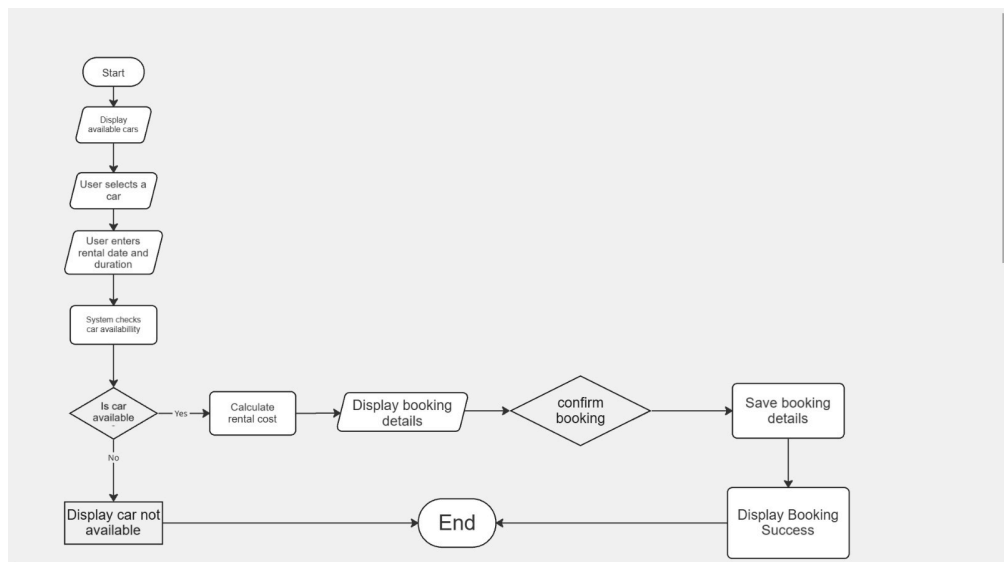
2.3 Car Booking Process

The system enables users to select a car, choose rental dates, and submit a booking request easily.

2.3.1 Car Booking Process Pseudocode

```
START
DISPLAY "Available Cars List"
INPUT selected_car
INPUT rental_start_date
INPUT rental_end_date
SEARCH "cars_table" FOR selected_car
IF selected_car is available FOR given dates THEN
    CALCULATE rental_cost
    DISPLAY rental_cost
    DISPLAY "Confirm Booking? (Yes/No)"
    INPUT user_confirmation
    IF user_confirmation == "Yes" THEN
        SAVE booking_details
        DISPLAY "Booking successful."
    ELSE
        DISPLAY "Booking cancelled by user."
    ENDIF
ELSE
    DISPLAY "Selected car is not available."
ENDIF
END
```

2.3.2 Car Booking Process Flowchart



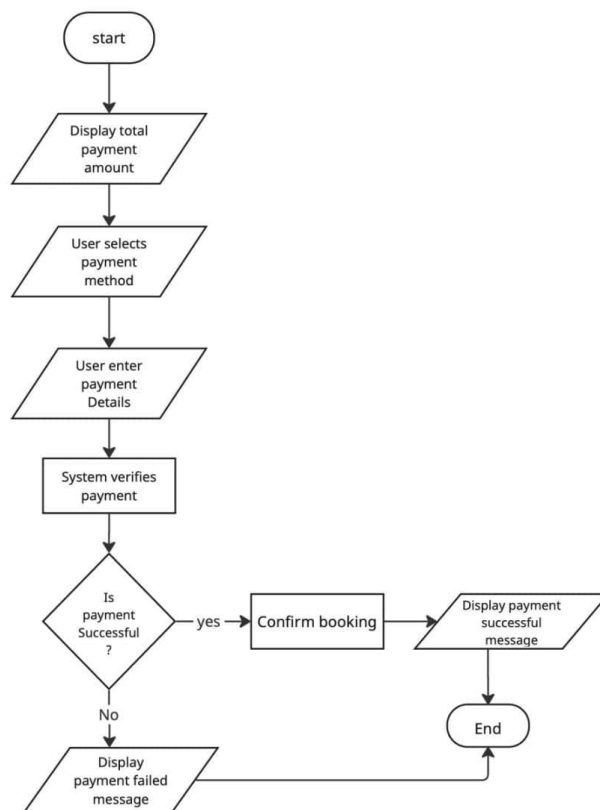
2.4 Payment Processing

The system processes customer payments securely and confirms successful transactions for car bookings.

2.4.1 Payment Processing Pseudocode

```
START
DISPLAY total_payment_amount
DISPLAY "Select Payment Method"
INPUT payment_method
INPUT payment_details
VERIFY payment_details
IF payment is successful THEN
    UPDATE booking status TO "Confirmed"
    DISPLAY "Payment successful."
ELSE
    DISPLAY "Payment failed. Please try again."
ENDIF
END
```

2.4.2 Payment Processing Flowchart



2.5 Booking Management (Cancel and View Booking)

The system allows users to view their booking details and cancel reservations when necessary.

2.5.1 Pseudocode

```
START
DISPLAY "1. View Bookings"
DISPLAY "2. Cancel a Booking"
INPUT user_choice

IF user_choice == 1 THEN
    READ customer_id FROM session
    SEARCH "bookings_table" WHERE status is NOT 'Deleted'
    IF bookings found THEN
        DISPLAY list of bookings (ID, Car Model, Date, Total Price)
    ELSE
        DISPLAY "No active bookings found."
    ENDIF

ELSE IF user_choice == 2 THEN
    INPUT booking_id_to_cancel
    SEARCH "bookings_table" FOR booking_id_to_cancel

    IF booking exists AND status is 'Confirmed' THEN
        UPDATE status TO 'Cancelled'
        DISPLAY "Booking has been successfully cancelled."
        REFUND_PROCESS
    ELSE
        DISPLAY "Invalid Booking ID or booking cannot be cancelled."
    ENDIF

ELSE
    DISPLAY "Invalid Selection."
ENDIF
END
```

2.5.2 Flowchart

