Chapter 1

INTRODUCTION

A **Smart Seating Management System** is an intelligent solution designed to efficiently monitor, allocate, and manage seating spaces using modern technologies such as sensors, IoT, and data analytics. It helps track seat availability in real time, reduces overcrowding and manual effort, and improves space utilization in places like classrooms, libraries, offices, auditoriums, and public venues. By automating seat assignment and providing accurate occupancy information, the system enhances user convenience, ensures better resource management, and supports safer, more organized environments.

1.1 Brief history of Smart Seating Management system

The concept of a **Smart Seating Management System** evolved from traditional manual seating arrangements and basic reservation systems used in classrooms, cinemas, and transport facilities. With the advancement of computer-based booking systems in the late 20th century, seating management became more organized and automated. The rapid growth of IoT, wireless sensors, and smart devices in the 2000s further transformed these systems by enabling real-time seat monitoring and dynamic allocation. Today, smart seating systems integrate cloud computing and data analytics to improve efficiency, comfort, and space utilization across educational, corporate, and public environments.

1.2 Modern Biometric system

A **modern Smart Seating Management System, called Ethno Café System** leverages advanced technologies such as IoT sensors, RFID, mobile applications, cloud computing, and AI-based analytics to provide real-time monitoring and intelligent seat allocation. These systems can automatically detect occupancy, guide users to available seats, support online reservations, and generate usage insights for better decision-making. Widely used in smart classrooms, offices, libraries, transportation hubs, and event venues, modern systems focus on improving user experience, maximizing space utilization, reducing energy consumption, and ensuring safety through contactless and automated operations.

Chapter 2

Problem Statement

2.1 Description

Field Findings:

- Difficulty in tracking available seats manually
- Delays in seating customers
- Overbooking or underutilization of seats
- Lack of proper record maintenance
- Need for an automated seat management solution

2.2 Challenge Statement

“To design and develop an efficient Restaurant Seat Management System that automates seat allocation, minimizes customer waiting time, reduces manual errors, and ensures optimal utilization of available seating using file structure concepts.”

Chapter 3

3.1 Design Thinking Process

a) Empathize: Interviews with 30+ customers and 7 staffs revealed delays, failed scans, and mismatches.

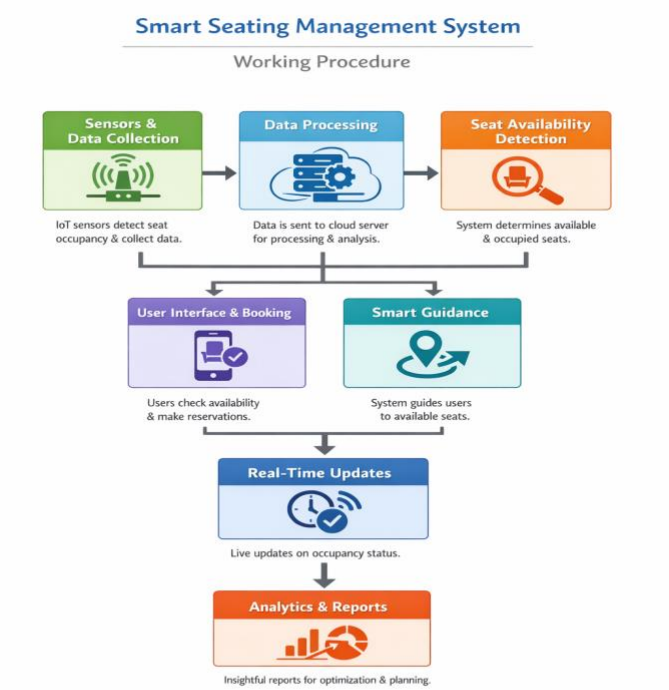
b) Define: Key needs identified: Difficulty in managing restaurant seats manually, Confusion during peak hours, Long waiting time for customers

c) Ideate: Create an automated seat management system, Track available and occupied seats, Allocate seats quickly and efficiently.

d) Prototype: Design a simple menu-driven program to store table details using files, Implement seat allocation and release functions.

e) Test: Test system with different inputs, Verify correct seat allocation and update system based on errors.

3.2 Methodology



3.3 Prototype Description

3.3.1 Materials Used

a) Computer / Laptop

b) C Programming Language

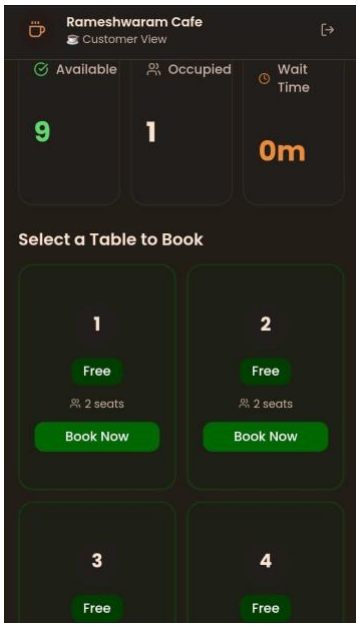
c) File Structures Concepts

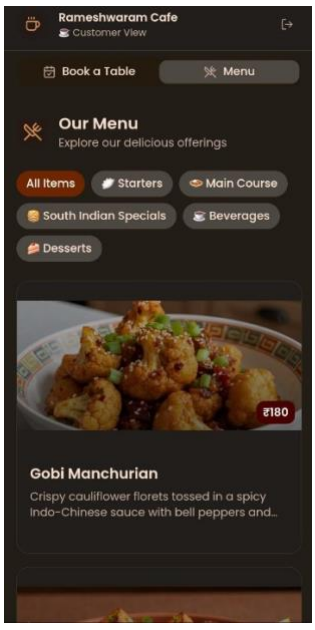
d) Code Editor (VS Code / Dev C++ / Turbo C)

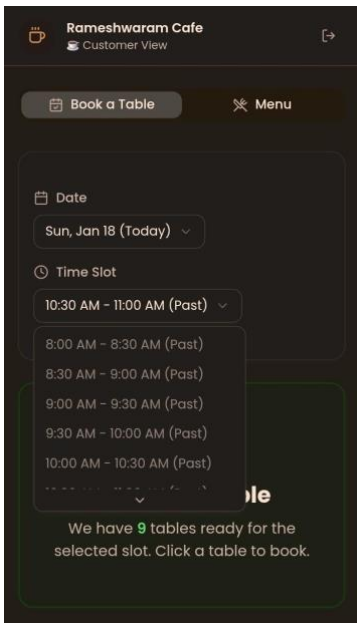
e) Operating System (Windows / Linux)

f) Text Editor for documentation (MS Word)

3.3.2 System Diagram







Chapter 4

Implementation

```
@import 'tailwindcss';
@import 'tw-animate-css';

@custom-variant dark (&:is(.dark *));

:root {
  --background: oklch(1 0 0);
  --foreground: oklch(0.145 0 0);
  --card: oklch(1 0 0);
  --card-foreground: oklch(0.145 0 0);
  --popover: oklch(1 0 0);
  --popover-foreground: oklch(0.145 0 0);
  --primary: oklch(0.205 0 0);
  --primary-foreground: oklch(0.985 0 0);
  --secondary: oklch(0.97 0 0);
  --secondary-foreground: oklch(0.205 0 0);
  --muted: oklch(0.97 0 0);
  --muted-foreground: oklch(0.556 0 0);
  --accent: oklch(0.97 0 0);
  --accent-foreground: oklch(0.205 0 0);
  --destructive: oklch(0.577 0.245 27.325);
  --destructive-foreground: oklch(0.577 0.245 27.325);
  --border: oklch(0.922 0 0);
  --input: oklch(0.922 0 0);
  --ring: oklch(0.708 0 0);
  --chart-1: oklch(0.646 0.222 41.116);
  --chart-2: oklch(0.6 0.118 184.704);
  --chart-3: oklch(0.398 0.07 227.392);
  --chart-4: oklch(0.828 0.189 84.429);
  --chart-5: oklch(0.769 0.188 70.08);
  --radius: 0.625rem;
  --sidebar: oklch(0.985 0 0);
  --sidebar-foreground: oklch(0.145 0 0);
  --sidebar-primary: oklch(0.205 0 0);
  --sidebar-primary-foreground: oklch(0.985 0 0);
  --sidebar-accent: oklch(0.97 0 0);
  --sidebar-accent-foreground: oklch(0.205 0 0);
  --sidebar-border: oklch(0.922 0 0);
  --sidebar-ring: oklch(0.708 0 0);
}

.dark {
  --background: oklch(0.145 0 0);
  --foreground: oklch(0.985 0 0);
  --card: oklch(0.145 0 0);
  --card-foreground: oklch(0.985 0 0);
  --popover: oklch(0.145 0 0);
  --popover-foreground: oklch(0.985 0 0);
  --primary: oklch(0.985 0 0);
```

```
  --primary-foreground: oklch(0.205 0 0);
  --secondary: oklch(0.269 0 0);
  --secondary-foreground: oklch(0.985 0 0);
  --muted: oklch(0.269 0 0);
  --muted-foreground: oklch(0.708 0 0);
  --accent: oklch(0.269 0 0);
  --accent-foreground: oklch(0.985 0 0);
  --destructive: oklch(0.396 0.141 25.723);
  --destructive-foreground: oklch(0.637 0.237 25.331);
  --border: oklch(0.269 0 0);
  --input: oklch(0.269 0 0);
  --ring: oklch(0.439 0 0);
  --chart-1: oklch(0.488 0.243 264.376);
  --chart-2: oklch(0.696 0.17 162.48);
  --chart-3: oklch(0.769 0.188 70.08);
  --chart-4: oklch(0.627 0.265 303.9);
  --chart-5: oklch(0.645 0.246 16.439);
  --sidebar: oklch(0.205 0 0);
  --sidebar-foreground: oklch(0.985 0 0);
  --sidebar-primary: oklch(0.488 0.243 264.376);
  --sidebar-primary-foreground: oklch(0.985 0 0);
  --sidebar-accent: oklch(0.269 0 0);
  --sidebar-accent-foreground: oklch(0.985 0 0);
  --sidebar-border: oklch(0.269 0 0);
  --sidebar-ring: oklch(0.439 0 0);
}

@theme inline {
  --font-sans: 'Geist', 'Geist Fallback';
  --font-mono: 'Geist Mono', 'Geist Mono Fallback';
  --color-background: var(--background);
  --color-foreground: var(--foreground);
  --color-card: var(--card);
  --color-card-foreground: var(--card-foreground);
  --color-popover: var(--popover);
  --color-popover-foreground: var(--popover-foreground);
  --color-primary: var(--primary);
  --color-primary-foreground: var(--primary-foreground);
  --color-secondary: var(--secondary);
  --color-secondary-foreground: var(--secondary-foreground);
  --color-muted: var(--muted);
  --color-muted-foreground: var(--muted-foreground);
  --color-accent: var(--accent);
  --color-accent-foreground: var(--accent-foreground);
  --color-destructive: var(--destructive);
  --color-destructive-foreground: var(--destructive-foreground);
  --color-border: var(--border);
  --color-input: var(--input);
  --color-ring: var(--ring);
  --color-chart-1: var(--chart-1);
  --color-chart-2: var(--chart-2);
  --color-chart-3: var(--chart-3);
  --color-chart-4: var(--chart-4);
  --color-chart-5: var(--chart-5);
  --radius-sm: calc(var(--radius) - 4px);
```

```
  --radius-md: calc(var(--radius) - 2px);
  --radius-lg: var(--radius);
  --radius-xl: calc(var(--radius) + 4px);
  --color-sidebar: var(--sidebar);
  --color-sidebar-foreground: var(--sidebar-foreground);
  --color-sidebar-primary: var(--sidebar-primary);
  --color-sidebar-primary-foreground: var(--sidebar-primary-
foreground);
  --color-sidebar-accent: var(--sidebar-accent);
  --color-sidebar-accent-foreground: var(--sidebar-accent-
foreground);
  --color-sidebar-border: var(--sidebar-border);
  --color-sidebar-ring: var(--sidebar-ring);
}

@layer base {
  * {
    @apply border-border outline-ring/50;
  }
  body {
    @apply bg-background text-foreground;
  }
}
```

Chapter 5

Results and Analysis

User Testing & Feedback

Participants: 6 students and 1 guide.

Quantitative Results:

- Queue time reduced from 10 minutes to 1 minute.
- Accuracy: 100% during testing.

Qualitative Feedback:

- Customers: “Reduces waiting time.”
- Staff: “Digital seat allotment is extremely helpful.”
- Admin: “No data mismatches during testing.”

Sustainability:
The Smart Seating Management System promotes sustainability by reducing paper usage through the replacement of manual registers and seating logs with digital records. Efficient seat utilization minimizes resource wastage and improves operational efficiency, while automation saves time and effort for restaurant staff. By reducing human errors and eliminating repetitive rework, the system ensures smoother operations and supports long-term use with minimal maintenance requirements.

Scalability:
The system is highly scalable and can be easily expanded to manage a larger number of tables and seats as the restaurant grows. It supports multiple sections or floors and can be upgraded to include features such as online reservations. Additionally, the system can integrate billing and payment modules and be extended across multiple branches, making it suitable for chain restaurants and future expansion.

Chapter 6

Conclusion & Future Work

Conclusion:

- Manual seat management is inefficient
- The proposed system automates seat allocation
- Reduces customer waiting time
- Improves accuracy and seating utilization
- Enhances overall restaurant management

Future Work:

- Online table reservation feature
- Mobile application support
- Integration with billing system
- Support for multiple restaurant branches

References

- Reema Thareja, Data Structures Using C
- Oxford University Press Ellis Horowitz and Sartaj Sahni
- Fundamentals of Data Structures
- Universities Press Lecture Notes on File Structures
- VTU Online resources and tutorials on C programming and file handling

Annexures

Annexure A – User Feedback Forms

METRIC	SCORE	FEEDBACK
Ease of Use	4.8/5	Simple interface, easy navigation
Seat Availability Accuracy	4.7/5	Real-time updates are reliable
System Responsiveness	4.6/5	Fast seat allocation and status refresh
Feature Completeness	4.7/5	Covers booking, monitoring, and guidance
Overall Satisfaction	4.75/5	Improves seating efficiency significantly

Annexure B – Development Sprints

- **Sprint 1:** Requirement analysis, system architecture design, database setup
- **Sprint 2:** Sensor integration, seat occupancy detection, data collection
- **Sprint 3:** Backend processing, real-time seat availability logic
- **Sprint 4:** User interface development, booking and guidance module
- **Sprint 5:** Testing, feedback integration, optimization, and deployment

Annexure C – Team Structure

NAME	ROLE
Keerthi S (B-09)	App Development
Komala T G (B-12)	Presentation
Lakshmi K L (B-17)	Data Collection
M V Shreya (B-22)	Feedback Collection
Manasaa S K (B-26)	Report
G Kushik (B-66)	App Testing