

Work Space Management System

Dhivakaran R, Thillaikumar P, Arunkumar S

Project Guide : Dr S Prakash

Information Technology

Bachelor of Technology

Sri Shakthi Institute of Engineering and Technology (Autonomous), Coimbatore 641 062

ABSTRACT

In today's dynamic work environment, effective utilization of physical work spaces is paramount for enhancing productivity, collaboration, and employee satisfaction. This abstract presents a comprehensive Work Space Management System (WSMS) designed to address the multifaceted needs of modern organizations.

Key features of the WSMS include real-time monitoring of space occupancy, intelligent scheduling of meetings and resources, personalized work space preferences for employees, and dynamic adjustment of space configurations based on usage patterns and employee feedback. Through a user-friendly interface accessible via desktop and mobile devices, employees can easily reserve work spaces, locate colleagues, and access relevant information, fostering a more connected and agile work environment. Furthermore, the WSMS fosters collaboration and communication among employees by facilitating seamless booking of meeting rooms, virtual collaboration spaces, and shared resources. The WSMS integrates advanced technologies such as Internet of Things (IoT), data analytics, and artificial intelligence to optimize the allocation and utilization of work spaces.

Overall, the Work Space Management System offers a holistic approach to optimizing work space utilization, enhancing employee satisfaction, and maximizing organizational efficiency in the modern workplace.

1. INTRODUCTION

1.1 OVERVIEW OF THE PROJECT

A workspace management system is a software platform or application designed to help organizations efficiently manage their physical workspaces, such as offices, meeting rooms, and spaces. It typically includes features to facilitate various aspects of workspace utilization, allocation, and optimization. Here's a brief explanation of its key components and functionalities:

Space Allocation: The system enables administrators to allocate specific spaces to individuals or teams based on their requirements, such as assigning desks or offices to employees or reserving meeting rooms for specific events.

Resource Booking: Users can easily book available resources like meeting rooms, equipment, or other facilities through the system. This helps prevent double bookings and ensures smooth scheduling of activities.

Optimization and Utilization Analysis: Workspace management systems often come with analytics tools that provide insights into space utilization trends. By analyzing data on occupancy rates, peak usage times, and other metrics, organizations can optimize their workspace layouts and improve resource allocation.

Visitor Management: Some systems include features for managing visitors, such as issuing visitor passes, registering guests, and tracking their movements within the workspace for security and compliance purposes.

Integration with Other Systems: Integration capabilities allow the workspace management system to sync with other software tools used within the organization, such as calendar applications, HR systems, or facility management software. This ensures seamless data flow and avoids duplication of effort.

Mobile Accessibility: Many workspace management systems offer mobile apps or responsive web interfaces, allowing users to access and manage workspace resources from anywhere using their smartphones or tablets.

Flexible Work Arrangements: With the rise of remote work and flexible schedules, modern workspace management systems often accommodate various work arrangements, such as hot desking, hoteling, or remote workspace booking, to cater to the diverse needs of employees.

Overall, a workspace management system streamlines the process of managing physical workspaces, improves resource utilization, enhances productivity, and provides a better experience for both employees and administrators.

1.2 MODULE DESCRIPTION

The module is to ensure the objectives those are :

1. Space Allocation Module:

Provides tools for assigning and managing workspace resources.

Allows administrators to allocate desks, offices, meeting rooms, etc., to individuals or teams.

Includes features for tracking occupancy and availability of spaces.

Resource Allocation Module:

Allows users to reserve workspace resources such as meeting rooms, equipment, or parking spots.

Provides a calendar interface for checking availability and scheduling bookings.

Sends notifications and reminders to users about upcoming bookings.

3. Analytics and Reporting Module:

Provides insights into workspace utilization patterns and trends.

Generates reports on occupancy rates, peak usage times, and other key metrics.

Helps administrators make data-driven decisions to optimize space usage.

4. Visitor Management Module:

Allows for the registration and management of visitors to the workspace.

Provides tools for issuing visitor passes and tracking visitor movements within the premises.

Enhances security by monitoring and controlling visitor access.

5. Integration Module:

Enables integration with other software systems used within the organization.

Allows for synchronization of data with HR systems, calendar applications, access control systems, etc.

1.2 SYSTEM SPECIFICATION

1.3.1 HARDWARE SPECIFICATION

Processor : INTEL(R)2.10GHz
Installed memory (RAM) : 4 GB
Hard Disk : 160 GB
Operating System : Windows (11)

1.3.2 SOFTWARE SPECIFICATION

Front End : .NET, C SHARP
Back End : SQL SERVER
Tool : Visual studio
Database : SQL

1.4 SOFTWARE FEATURES

About .NET:

.NET is a robust and versatile software development framework created by Microsoft. Launched in the early 2000s, it has since become a cornerstone of modern application development, powering a wide range of software solutions across various platforms and devices. In this brief essay, we'll explore the key features, components, and benefits of .NET.

Firstly, .NET provides developers with a comprehensive set of tools and libraries for building types of applications, including web applications, desktop applications, mobile apps, cloud-based services, and more. Its extensive class libraries offer pre-built functions for common tasks, enabling developers to streamline the development process and focus on building innovative features.

One of the core components of .NET is the Common Language Runtime (CLR), which serves as the execution engine for .NET applications. The CLR provides features such as automatic memory management (garbage collection), exception handling, and language interoperability, developers to write code in multiple languages such as C#, Visual Basic, F#, or even managed C++.

Another key aspect of .NET is its support for cross-platform development. With the introduction .NET Core, Microsoft's open-source, cross-platform implementation of .NET, developers can now build and deploy .NET applications on Windows, macOS, and Linux environments. This flexibility is particularly valuable in today's multi-platform world, where businesses need to research users across different devices and operating systems.

Additionally, .NET offers seamless integration with other Microsoft technologies and services, as Azure cloud services, SQL Server databases, and Visual Studio development environment. This integration simplifies the development and deployment process, enabling developers to leverage a comprehensive ecosystem of tools and services to build scalable, secure, and efficient applications.

In conclusion, .NET is a powerful and versatile framework that empowers developers to build a wide range of applications efficiently. With its rich set of tools, cross-platform support, seamless integration, and strong security features, .NET continues to be a preferred choice for businesses developers worldwide seeking to create modern, high-performance software solutions. Whether you're building web applications, mobile apps, or cloud-based services, .NET provides the tools and capabilities to bring your ideas to life.

Why .NET

.NET is a robust and versatile software development framework created by Microsoft. Launched in the early 2000s, it has since become a cornerstone of modern application development, powering a wide range of software solutions across various platforms and devices. In this brief essay, we'll explore the key features, components, and benefits of .NET.

We can see the description of the features in detail.

- o Versatility
 - o Common Language Runtime
 - o Cross Platform Development
 - o Security
 - o Extensive Libraries
 - o Community Support
 - o Scalability and Performance
-
1. Comprehensive Framework: .NET is a robust and comprehensive software development Framework developed by Microsoft.
 2. Versatile Application Development: It supports building a wide range of applications, including web applications, desktop applications, mobile apps, and cloud-based services.
 3. Extensive Libraries: .NET offers a vast collection of class libraries that provide pre-built functions for common tasks, speeding up the development process.
 4. Common Language Runtime (CLR): The CLR serves as the execution engine for .NET applications, offering features such as automatic memory management and language interoperability.

5. **Cross-Platform Development:** With .NET Core, developers can build and deploy applications on Windows, macOS, and Linux environments, enabling cross-platform development.

6. **Integration with Microsoft Technologies:** .NET seamlessly integrates with other Microsoft technologies and services, such as Azure cloud services, SQL Server databases, and Visual Studio development environment.

7. **Security Features:** .NET provides built-in security features like code access security, role-based security, and encryption libraries to help developers implement robust security measures in their applications.

8. **Regular Updates and Patches:** Microsoft regularly releases updates and patches for .NET, ensuring that applications built on the framework remain protected against security threats.

9. **Strong Community Support:** .NET has a vibrant and active community of developers, providing resources, support, and updates to help developers succeed in their projects.

10. **Scalability and Performance:** .NET applications are known for their scalability and performance, making them suitable for building high-performance, enterprise-grade solutions.

Overall, .NET is a versatile, scalable, and secure framework that empowers developers to build modern and innovative applications across various platforms and devices.

C SHARP :

C# (pronounced "C sharp") is a powerful and versatile programming language developed by Microsoft. It is designed for building a wide range of applications, including web applications, desktop applications, mobile apps, games, and more. C# is part of the .NET framework and is widely used for its simplicity, flexibility, and performance.

Key features of C# include:

1. Object-Oriented Programming: C# is an object-oriented language, allowing developers to create modular and reusable code through classes and objects.
2. Type Safety: C# is a strongly typed language, which means variables must be declared with a specific data type. This enhances code reliability and reduces runtime errors.
3. Garbage Collection: C# features automatic memory management through garbage collection, freeing developers from manual memory management tasks and reducing the risk of memory leaks.
4. Platform Independence: C# code can run on various platforms, including Windows, macOS, and Linux, thanks to the cross-platform capabilities of the .NET framework.
5. Modern Language Constructs: C# includes modern language features such as LINQ (Language Integrated Query), async/await for asynchronous programming, and lambda expressions, which enhance code readability and productivity.
6. Rich Standard Library: C# benefits from a rich standard library (Base Class Library or BCL) that provides pre-built functions and classes for common programming tasks, reducing the need for developers to reinvent the wheel.
7. Integration with .NET Ecosystem: C# seamlessly integrates with other technologies in the .NET ecosystem, such as ASP.NET for web development, Xamarin for cross-platform mobile development, and Unity for game development.

8. Community Support: C# has a large and active community of developers who contribute libraries, frameworks, and resources, making it easier for developers to find solutions to problems and learn new technique

SQL:

SQL, or Structured Query Language, is the cornerstone of database management systems, serving as a universal language for interacting with relational databases. Originally developed by IBM in the 1970s, SQL has evolved into a standardized language used by virtually all modern database systems, including Oracle, MySQL, SQL Server, and PostgreSQL. In this essay, we'll explore the fundamental aspects and significance of SQL in the realm of data management.

At its core, SQL provides a set of commands and syntax rules for defining, manipulating, and querying data stored in relational databases. Its versatility lies in its ability to handle various data operations seamlessly, ranging from creating and modifying database structures to retrieving and analyzing data.

One of the primary functions of SQL is data definition, where users can create, alter, or drop database objects such as tables, views, indexes, and constraints. This aspect of SQL facilitates the organization and structuring of data within the database, laying the foundation for efficient data management and retrieval.

Data manipulation is another essential aspect of SQL, enabling users to insert, update, delete, and retrieve data from the database. The SELECT statement, in particular, is a powerful tool for query data, allowing users to filter, sort, aggregate, and join data from multiple tables based on specific criteria.

SQL also plays a crucial role in ensuring data integrity and security within the database. Through use of constraints, triggers, and referential integrity mechanisms, SQL allows users to enforce data consistency rules, prevent data corruption, and protect sensitive information from unauthorized access.

SYSTEM STUDY

2.1 EXISTING SYSTEM

- The existing system does not have a proper channel of communication between the workspace management and the employee.
- In this the employee who need to book seats should come to their respective office only where it may lead to unavailable of seats.
- Sometimes if the employee is late to office means the seats will be occupied by some other employee where it may lead to confusions.

2.2 PROPOSED SYSTEM

In-order to avoid the limitation in the existing system is being developed.

- The current system that is going to be built overcomes all the difficulties given above.
- Employee who need to book their seats can easily book through online.
- The seats booked by Employee through online cannot be occupied by other employee.
- Eventhough the Employee is late to office the seats booked will remain for sometime.

2.2.1 ADVANTAGES OF PROPOSED SYSTEM

- ❖ User friendly interface
- ❖ High Efficiency
- ❖ Fast access
- ❖ Time Consumption

SYSTEM DESIGN

3.1 FILE DESIGN

System design is the process of planning a new system to complement or altogether replace the old system. The purpose of the design phase is the first step in moving from the problem domain to the solution domain. The design of the system is the critical aspect that affects the quality of the application. System design is also called top-level design. The design phase translates the logical aspects of the system into physical aspects of the system.

3.2 INPUT DESIGN

The data, which is input to a computer – based information system, must be correct. If data is carelessly input and errors enter the system, it will lead to incorrect results whose consequences will be expensive and embarrassing to the designer. In data processing, the data entry operator often makes errors. This can be controlled by input design by using menu, interactive dialogue, consistent format etc.

In this system the users are provided with user friendly pages to give the input and if the user gives any wrong input validations are done and message boxes are provided in the necessary places. The message specified in the message box is specified in a polite and in an informative manner.

System is interactive dialogue, which simplifies the data entry or access, instead of remembering what to enter. User can choose from a list of options and type it in the cursor position. This will reduce the number of corrections while entering the data.

3.3 DATABASE DESIGN

The database design involves creation of tables that are represented in physical database as stored files. They have their own existence. Each table consists of rows and columns where each row can be viewed as record that consists of related information and column can be viewed as field of data of same type. The table is also designed with some position can have a null value.

The database design of project is designed in such a way values are kept without redundancy and with normalized format. Refer the appendix for screen shots of database design.

OUTPUT DESIGN
The proposed system is a web oriented system and hence it does not provide any reports. The output results are viewed in the web pages itself. Outputs from the computer system are required primarily to communicate the result of processing to users. They are also used to override a permanent copy of the results for later consultation. The output reports and input documents should be documented in terms of data content .

4 SYSTEM TESTING AND IMPLEMENTATION

4.1 TESTING

Testing is a series of different tests that whose primary purpose is to fully exercise the computer based system. Although each test has a different purpose, all work should verify that all system element have been properly integrated and performed allocated function. Testing is the process of checking whether the developed system works according to the actual requirement and objectives of the system.

The philosophy behind testing is to find the errors. A good test is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers the undiscovered error. Test cases are devised with this purpose in mind. A test case is a set of data that the system will process as an input. However the data are created with the intent of determining whether the system will process them correctly without any errors to produce the required output.

Types of Testing:

- Unit Testing
- Integration Testing
- Output Testing
- User acceptance Testing
- Performance Testing
- Output Testing

Unit Testing

All modules were tested and individually as soon as they were completed and were checked for their correct functionality.

Integration Testing

The entire project was split into small program; each of these single programs gives a frame as an output. These programs were tested individually; at last all these programs were combined together by creating another program where all these constructors were used. It gives a lot of problem by not functioning in an integrated manner.

The user interface testing is important since the user has to declare that the arrangements made in frames are convenient and it is satisfied. When the frames were given for the test, the end user gave suggestion. Based on their suggestions the frames were modified and put into practice.

Validation Testing

At the culmination of the black box testing software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of test i.e., Validation succeeds when the software function in a manner that can be reasonably accepted by the customer.

Output Testing

After performing the validation testing the next step is output testing of the proposed system. Since the system cannot be useful if it does not produce the required output. Asking the user about the format in which the system is required tests the output displayed or generated by the system under consideration. Here the output format is considered in two ways. One is on screen and another one is printed format. The output format on the screen is found to be corrected as the format was designed in the system phase according to the user needs. And for the hardcopy the output comes according to the specifications requested by the user.

White box testing

White box testing (also known as Clear Box Testing, Open Box Testing, Glass Box Testing, Transparent Box Testing, Code-Based Testing or Structural Testing) is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester. The tester chooses inputs to exercise paths through the code and determines the appropriate outputs. Programming know-how and the implementation knowledge is essential.

Whitebox testing is testing beyond the user interface and into the nitty-gritty of a system. This method is named so because the software program, in the eyes of the tester, is like a white/transparent box; inside which one can clearly see.

Black box testing

Black box testing, also known as Behavioral Testing, is a software testing method in which the internal structure/design/implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional.

This method is named so because the software program, in the eyes of the tester, is like a black box; inside which one cannot see.

This method attempts to find errors in the following categories:

- Incorrect or missing functions
- Interface errors
- Errors in data structures or external database access
- Behavior or performance errors
- Initialization and termination errors

Definition by ISTQB

- **Black box testing:** Testing, either functional or non-functional, without reference to the internal structure of the component or system.
- **Black box test design technique:** Procedure to derive and/or select test cases based on an analysis of the specification, either functional or non-functional, of a component or system without reference to its internal structure.

Acceptance testing

This testing is done to verify the readiness of the system for the implementation. Acceptance testing begins when the system is complete. Its purpose is to provide the end user with the confidence that the system is ready for use. It involves planning and execution of functional tests, performance tests and stress tests in order to demonstrate that the implemented system satisfies its requirements. Tools of special importance during acceptance testing include:

Test coverage Analyzer

Records the control paths followed for each test case.

Timing Analyzer

Also called a profiler, reports the time spent in various regions of the code and areas to concentrate on to improve system performance.

4.2 SYSTEM IMPLEMENTATION

System implementation of a workspace management system involves the development, testing, and deployment of the software to effectively manage physical workspaces. It encompasses phases /

such as requirements gathering, system design, development, testing, deployment, and ongoing

maintenance to ensure smooth operations. Through careful planning and execution, organizations

can optimize space utilization, improve productivity, and enhance user experience within their work environments.

i. Feasibility study

A feasibility study for a workspace management system assesses the practicality and viability of implementing such a system within an organization. It typically examines technical, economic, and operational factors to determine whether the project is feasible. This study helps stakeholders understand the potential benefits, costs, and risks associated with deploying the system, guiding decision-making regarding its implementation.

It consists of the following:

- Statement of the problem
- Summarizing of findings and recommendations
- Details of findings
- Recommendations and conclusions

I addressed three types of feasibility study in my research, they include the following.

1. Economic Feasibility:

A network-based system requires a lot of equipment such as cables, hubs etc. This requires a lot of initial capital to install the network. On the other hand, it allows sharing of resources and information and centralized administration hence cheaper.

2. Technical Feasibility

Since it is not a complex system, we have the technical feasibility of developing the system.

3. Time Feasibility

The system is a small one and hence the time frame of three months allocated for development is enough hence there is time feasibility.

From the above we choose to use a network based database system because as compared to the other strategies, it is more feasible. It will contain an interface that is distributed in the network and is connected to a central database.

Feasibility study involves cost/benefit analysis. In the process, the cost and benefits are estimated with greater accuracy. If cost and benefit should be quantified to make a good system that is affordable.

ii Analysis

Analysis starts with a system request that describes the problems or desired changes in the system. It identifies the nature and scope of the business opportunity and problem by performing a feasibility study.

iii Design

The Design phase creates a blueprint for the new system that will satisfy all documented requirements. It identifies all necessary outputs, inputs, interfaces and processes. Designs internal and external controls that will ensure:

- Reliability
- Security
- Maintainability
- Accuracy

The design is documented in the systems design specification and presented to the management and users for their review and approval. The involvement of Management and users is to avoid any misunderstanding about what the system will do, how it will do it and how much it will cost.

Implementation

In the implementation phase, the new system is constructed by the programmers and designers and finally given to the final user. After implementation data is converted into system files, users are trained, and the actual transition to the new system is undertaken.

A Systems Evaluation is later done to determine if the system operates properly and if the cost of the system and benefits are within expectations.

Post implementation and maintenance

During this phase the IT department and staff maintains (corrects the errors and adapt to changes in the environment) and enhances the system. Enhancements provide a maximized return on IT investments.

5. CONCLUSION

I believe I have done enough research on the Project and am ready to start and complete the project over the period specified and also make the Output.

With the help of this project,

- This project as described in detail, the application developed for the easy and time consumption to the people to book their seats through online.
- The application enables the organization to carry out all the report effectively after the implementation.
- When all the suggestions forwarded in the software proposal have been successfully completed.

Thank you in advance for your consideration.

FUTURE ENHANCEMENT

- a. This project has been developed keeping in mind all the given possible conditions to overcome the disadvantages of existing system.
- b. This application is implemented successfully.
- c. After implementation of the application the user may require some changes to be made with the project.
- d. Whenever changes are made it will not affect the performance or efficiency of the existing system. Several navigations can be provided for easy access of data.

REFERENCE

BOOK REFERENCES

- 1) J.F. DiMarzio, "Android: A Programmers Guide", First Edition, Tata McGraw-Hill, 2010.
- 2) Jason Morris, "Android User Interface Development", Packt Publishing, 2011.
- 3) "C# 4.0 The Complete Reference" by Herbert Schildt

WEBSITES

- <http://www.androiddeveloper.com/>
- <https://www.geeksforgeeks.org/csharp-programming-language/>
- <http://www.tutorialspoint.com//>

OTHER REFERENCES

<https://www.iofficecorp.com/blog/workplace-management-system-guide>

<https://www.gartner.com/reviews/market/integrated-workplace-management-systems>

<https://www.ibm.com/topics/integrated-workplace-management-system>

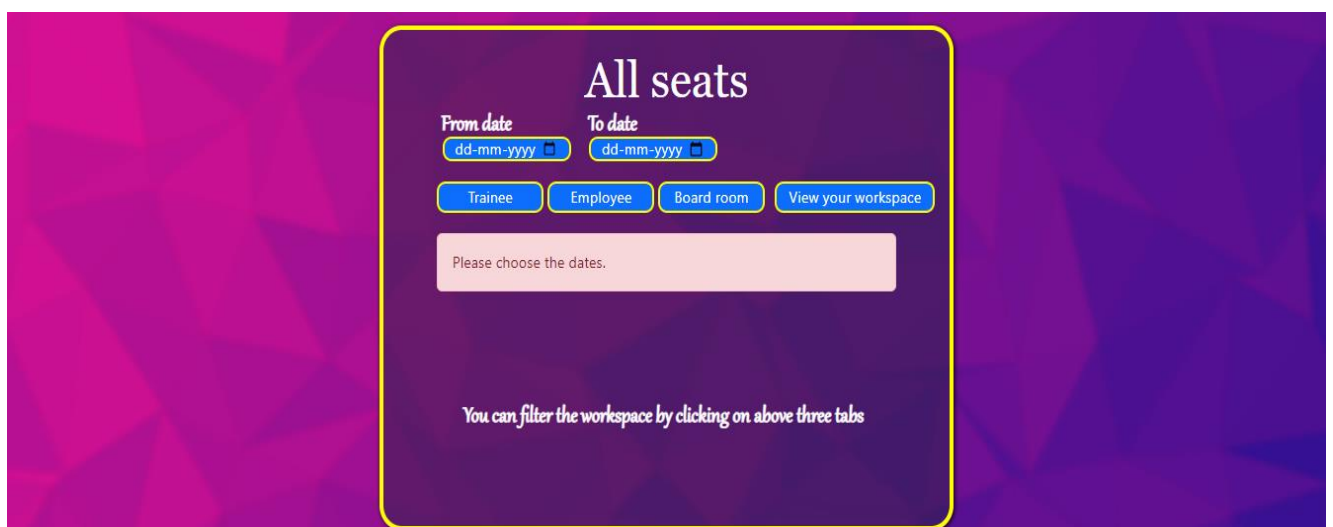
OUTPUT SCREENSHOTS

USER SIDE

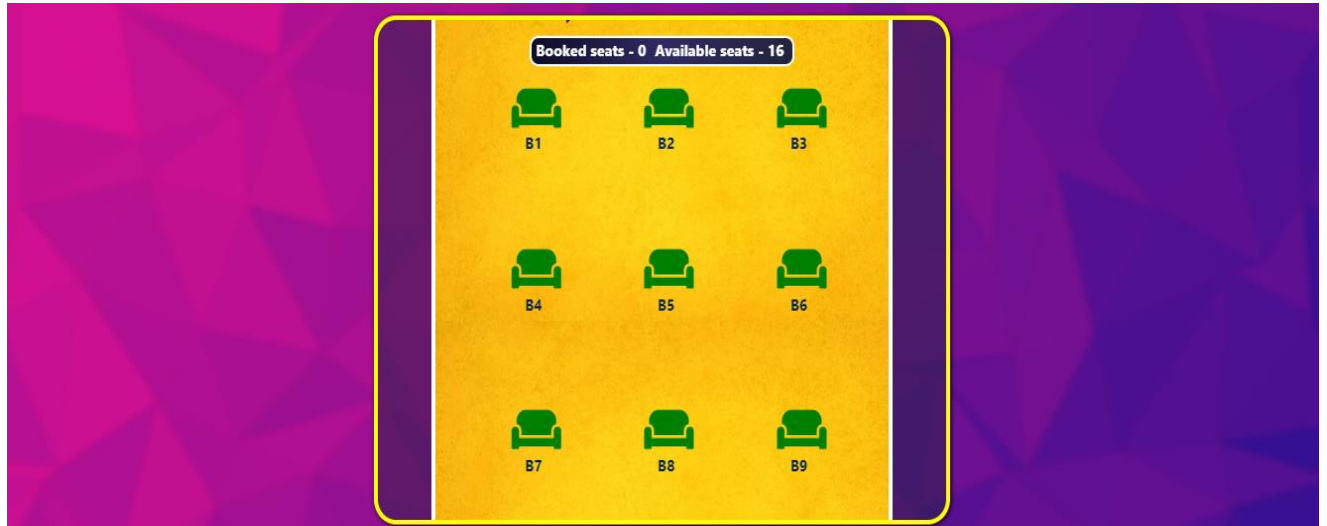
HOME PAGE



BOOKING 1



BOOKING 2



BOOKING PROCESS

Seat number	<input type="text" value="B1"/>
Employee code	<input type="text" value="CS2008"/>
Employee name	<input type="text" value="dhiva"/>
From date	<input type="text" value="10-04-2024"/>
To date	<input type="text" value="10-04-2024"/>
Priority	<input type="text" value="High"/>
<input type="button" value="Book"/>	

BOOKING SUCCESSFUL EMAIL

Booking confirmed Inbox x



infomessage24x7@gmail.com
to me ▾

9:51AM (4 minutes ago)



Your Booking has been Confirmed!

Seat number: B1

Booked from: 10-04-2024 to 10-04-2024

Priority: High

WORKSPACE

Your workspace

[Show all bookings](#)[Back](#)

Seat number	Employee code	Employee name	From date	To date	Total Days	Priority	Status	Remarks
B1	CS2008	dhiva	10-04-2024	10-04-2024	1	High	Active	-

ADMIN SIDE

ADMIN DASHBOARD :

[Dashboard](#)[User](#)[Manage workspace bookings](#)[Feedback](#)

Admin Dashboard

Employee count
22

Total seats
40

Booked seats
1

Available seats
24

Reserved seats
15

MANAGER USER

User details

First name	Last name	Email	Phone number	Action
Nikhil	Srinivasan	nikhil.seetha@claysys.info	8870869072	Make Admin Details Delete
Harshini	Anandhakumar	harshini.anandhakumar@claysys.info	9566958238	Make Admin Details Delete
Muraliprasad	Manoharan	murali.manoharan@claysys.info	7010518962	Make Admin Details Delete
Surya	Duraisamy	surya.duraisamy@claysys.info	7339675297	Make Admin Details Delete
Nivethitha	Ramanathan	nivethitha.ramanathan@claysys.info	9790605168	Make Admin Details Delete
Indhusree	Veerabahu	indhu.veerabahu@claysys.info	8778293216	Make Admin Details Delete
Test	karan	nikhil@gmail.com	122222586	Make Admin Details Delete

WORKSPACE MANAGE

All seats

Add seats

From date
01-04-2024

To date
12-04-2024

All seats

Booked seats

Available seats

Reserved seats

Unreserved seats

Seat number	Seat availability	Seat status	Action
 B1	Booked	Available	 Make reserved Delete seat

Booking details

Employee code: CS2008

Employee name: dhiva

Booking from: 10-04-2024

Booking to: 10-04-2024

BOOKING DETAILS

Booking details

From date

10-04-2024

To date

10-04-2024

search

All bookings

Active bookings

Upcoming bookings

Cancelled bookings

Expired bookings

Booking id	Seat number	Employee code	Employee name	Booked from	Booked till	Total days	Priority	Status	Action
59	B1	CS2008	dhiva	10-04-2024	10-04-2024	1	High	Active	Cancel booking

FEEDBACK:

Feedback

First name	Last name	Email	Country code	Phone number	Message	Action
Dhivakaran	ravi	dhivakaran2k2@gmail.com	91	6379969029	I can't login to the portal	Replied Delete
sample1	sample2	test@gmail.com	91	1222222255	Working	Replied Delete
Murali	Prasad	muraliprasad2002@gmail.com	91	8776789876	This is for testing purpose	Replied Delete

Reply: Yeah got it.