



# Cruise Ship Management System

## Abstract

This project is a web-based Cruise Ship Management System developed using **HTML, CSS, JavaScript, and Firebase**. It provides role-based dashboards for Admin, Supervisor, Manager, and Voyager, enabling functionalities such as managing items, registering voyagers, booking services, and viewing reports. The system focuses on security, usability, and responsive design.

## Introduction

The Cruise Ship Management System ensures smooth communication between administrators, supervisors, managers, and voyagers with modern web technologies.

## Objectives

- Provide separate dashboards for different roles.
- Implement secure authentication and role-based access.
- Enable item management, bookings, and registrations.
- Ensure responsive design for multiple devices.
- Maintain a user-friendly and efficient UI/UX.

## Methodology

The system uses Firebase for authentication and data storage, while the frontend is implemented with HTML, CSS, and JavaScript. Each role has restricted access to functionalities:

- **Admin:** Register voyagers, manage items.
- **Supervisor:** Monitor orders.
- **Manager:** Oversee bookings and cancellations.
- **Voyager:** Register, log in, book services, and manage bookings.

## Features

- Role-based login (Admin, Supervisor, Manager, Voyager).

- Item management (add/delete items).
- Voyager registration with success/error messages.
- Booking and cancellation system.
- Responsive design with media queries.
- Scrollable dashboards where needed.
- User feedback messages with styled backgrounds.

## Project Structure

```
CruiseShipApp/
├── Images/                #BG image
├── Scripts/                # JS logic for all pages(includes
firebase.js)
├── styles/                # Styling for all pages
├── html_docs/             # Admin page
├── Cruise_Ship_Management_Report.ipynb
├── Cruise_Ship_Management_Report.pdf
├── Usage Instructions.pdf
├── README.pdf
├── index.html             # Redirect to register page
└── README.md             # Documentation
```

## Test Cases

Test Case ID	Feature	Steps	Expected Output
TC01	Login	Enter valid credentials	Redirects to correct dashboard
TC02	Login	Enter invalid credentials	Error message displayed
TC03	Admin - Add Item	Add a new item	Success message shown
TC04	Admin - Delete Item	Delete an item	Success message shown
TC05	Register Voyager	Fill form and submit	Voyager registered successfully
TC06	Voyager - Book Item	Select and book an item	Booking success message shown
TC07	Voyager - Cancel Booking	Cancel an existing booking	Success message shown
TC08	Supervisor - Monitor Order	Open Orders dashboard	All Orders displayed
TC09	Manager - Monitor	Open bookings	All bookings displayed

Test Case ID	Feature	Steps	Expected Output
	Bookings	dashboard	
TC10	Responsive Design	Resize window < 540px	Hamburger menu appears

## Conclusion

The Cruise Ship Management System successfully demonstrates a role-based management application with authentication, data handling, and responsive UI. It reduces manual errors, improves efficiency, and enhances the voyager experience.