Hotel PMS “Paseo Las Mercedes” - Project Summary

Project Overview

- **Hotel Name**: Paseo Las Mercedes
- **Project Type**: Complete Hotel Property Management System (PMS)
- **Location**: `/home/ubuntu/hotel_pms_paseo_las_mercedes`
- **Status**: Fully operational with all core functionalities implemented
- **Current Phase**: Phase 5 Complete (Reporting & Analytics)

Technical Stack

- **Framework**: Next.js 14 with TypeScript
- **Database**: PostgreSQL with Prisma ORM
- **Authentication**: NextAuth.js with JWT
- **UI Library**: Radix UI components with Tailwind CSS
- **Package Manager**: Yarn
- **Styling**: Tailwind CSS with custom components

Completed Development Phases

Phase 1: Foundation and Configuration ✅

- Project initialization and structure
- Database setup with Prisma
- Authentication system with NextAuth.js
- Basic UI components and layout
- Environment configuration

Phase 2: Room Management ✅

- Room CRUD operations API (`/api/rooms/`)
- Room types and categories management
- Room availability tracking
- Room overview dashboard
- Interactive room grid component

Phase 3: Reservation System ✅

- Complete reservation API (`/api/reservations/`)
- Guest management system (`/api/guests/`)
- Booking creation and management
- Availability checking algorithms
- Reservation calendar interface

Phase 4: Check-in/Check-out Process ✅

- Check-in API and UI (`/api/checkin/` , `/checkin`)
- Check-out API and UI (`/api/checkout/` , `/checkout`)
- Guest services management
- Room assignment workflows
- Billing integration

Phase 5: Reporting & Analytics ✅

- Comprehensive reporting system (`/api/reports/`)
- Analytics dashboard (`/reports`)
- Financial reports and occupancy metrics
- Revenue tracking and forecasting
- Data visualization components

Current Database Schema (Prisma)

Core Models

```
User - Authentication and staff management
Room - Room inventory with types and amenities
RoomType - Room categorization
Guest - Guest information and history
Reservation - Booking management
CheckIn/CheckOut - Arrival/departure tracking
Service - Additional services and requests
Report - Analytics and reporting data
```

Key Relationships

- Reservations linked to Guests and Rooms
- Check-in/Check-out tied to Reservations
- Services connected to Guests and Reservations
- Historical tracking for all operations

Key Application Features

Dashboard (`/dashboard`)

- Real-time occupancy overview
- Revenue summaries
- Quick action buttons
- Key performance indicators

Room Management (`/rooms`)

- Visual room grid with status indicators
- Room details and amenities management
- Availability calendar
- Maintenance tracking

Reservations (`/reservations`)

- Advanced booking interface
- Guest search and selection
- Availability checking
- Reservation modification tools

Check-in/Check-out (`/checkin` , `/checkout`)

- Guest arrival processing
- Room key assignment
- Document verification
- Bill settlement and payment

Reporting (`/reports`)

- Occupancy analytics
- Revenue reports
- Guest demographics
- Performance dashboards

Technical Implementation Details

API Structure

```
/api/
├── auth/ - Authentication endpoints
├── rooms/ - Room management APIs
├── reservations/ - Booking system APIs
├── guests/ - Guest management APIs
├── checkin/ - Check-in process APIs
├── checkout/ - Check-out process APIs
├── services/ - Additional services APIs
└── reports/ - Analytics and reporting APIs
```

UI Components

```
/components/
├── ui/ - Reusable UI components (Radix-based)
├── dashboard/ - Dashboard-specific components
├── rooms/ - Room management components
├── reservations/ - Booking interface components
├── checkin/ - Check-in process components
├── checkout/ - Check-out process components
└── reports/ - Analytics and reporting components
```

Key Configuration Files

- `prisma/schema.prisma` - Database schema
- `lib/auth-options.ts` - Authentication configuration
- `lib/prisma.ts` - Database connection
- `.env` - Environment variables (DATABASE_URL, NEXTAUTH_SECRET, etc.)

Current Project State

- ✅ All APIs functional and tested
- ✅ Database schema complete and seeded
- ✅ Authentication system working
- ✅ All UI components implemented
- ✅ Navigation and routing complete
- ✅ Responsive design implemented
- ✅ Error handling in place

Recent Checkpoints Saved

1. “Room management functionality implemented”
2. “Reservation system Phase 3 completed”
3. “Check-in checkout system implemented”
4. “Reporting and analytics system”

Development Setup

```
cd /home/ubuntu/hotel_pms_paseo_las_mercedes
yarn install
yarn dev # Starts development server on localhost:3000
yarn build # Production build
```

Environment Variables Required

- `DATABASE_URL` - PostgreSQL connection string
- `NEXTAUTH_URL` - Application URL
- `NEXTAUTH_SECRET` - Authentication secret
- `ABACUSAI_API_KEY` - LLM API access (if needed)

Areas for Future Enhancement

1. **Payment Integration** - Stripe/PayPal integration for billing
2. **Housekeeping Module** - Room cleaning and maintenance scheduling
3. **Inventory Management** - Hotel supplies and amenities tracking
4. **Staff Management** - Employee scheduling and task assignment
5. **Guest Communication** - Automated emails and notifications
6. **Mobile Responsiveness** - Enhanced mobile interface
7. **Multi-language Support** - Internationalization
8. **Advanced Analytics** - ML-based insights and predictions

Common Commands for Continuation

```
# Test the application
cd /home/ubuntu/hotel_pms_paseo_las_mercedes && yarn build

# Create new checkpoint
# Use build_and_save_nextjs_project_checkpoint tool

# Add new features
# Use file_str_replace or batch_file_write tools

# Database updates
npx prisma db push
npx prisma generate
```

Notes for Next Developer

- The project follows a modular structure with clear separation of concerns
- All major functionalities are implemented and working
- Database is properly seeded with sample data
- Authentication is configured and functional
- UI is consistent and follows modern design patterns
- Code is TypeScript-compliant with proper error handling

The system is production-ready and can handle typical hotel operations for a mid-size property like “Paseo Las Mercedes”.