



**INSTITUTE FOR ADVANCED COMPUTING
&
SOFTWARE DEVELOPMENT (IACSD),
AKURDI, PUNE**

Documentation on

Urban Property Management System

PG-DAC Sept 2023

Submitted By:

Group No: 91

Roll No:

239213

239158

Name:

Sujeet Sankadiya

Hitesh Birla

Mrs. Vaishnavi Ghodke

Project Guide

Mr. Rohit Puranik

Centre Coordinator

Abstract

The Urban Property Management System (UPMS) is an innovative software solution tailored to address the complex challenges of managing urban properties efficiently. In today's fast-paced urban environments, the demand for streamlined property management systems is greater than ever. UPMS aims to fulfill this need by providing a comprehensive platform that caters to the diverse requirements of administrators, property owners, and tenants alike.

At the heart of UPMS lies its user authentication and authorization system, which ensures secure access to the platform. Users are able to log in using their unique email IDs and passwords, with the system employing robust JWT authentication for enhanced security measures. Furthermore, UPMS implements role-based access control, allowing users to sign up with specific roles such as administrator, property owner, or tenant. This role-based approach ensures that each user has access only to the features and functionalities relevant to their role, thereby enhancing data security and privacy.

Profile management is another key aspect of UPMS, enabling users to maintain and update their personal information as needed. Administrators, property owners, and tenants alike can effortlessly update their profiles, ensuring that all data within the system remains accurate and up-to-date. This feature enhances communication and transparency between stakeholders, fostering a collaborative environment for effective property management.

Property listing and booking functionalities are seamlessly integrated into UPMS, empowering property owners to showcase their properties with detailed descriptions and high-quality images. Tenants can browse through the list of available properties, select their desired options, and proceed to book them directly through the platform. This streamlined process eliminates the need for cumbersome paperwork and manual coordination, saving both time and effort for all parties involved.

.

ACKNOWLEDGEMENT

I take this occasion to thank God, almighty for blessing us with his grace and taking our endeavor to a successful culmination. I extend my sincere and heartfelt thanks to our esteemed guide, Mrs. Vaishnavi Ghodke for providing me with the right guidance and advice at the crucial juncture sand for showing me the right way. I extend my sincere thanks to our respected Centre Co-Ordinator Mr. Rohit Puranik, for allowing us to use the facilities available. I would like to thank the other faculty members also, at this occasion. Last but not the least, I would like to thank my friends and family for the support and encouragement they have given me during the course of our work.

Sujeet Sankadiya (230941220185)

Hitesh Birla (230941220068)

INDEX

Sr. No.	Title	Pg. No.
1	INTRODUCTION	5
1.1	PURPOSE	5
1.2	SCOPE	5
1.3	OBJECTIVE OF PROJECT	6
2	REQUIREMENT	7
3	DATABASE DESIGN	10
5	APPENDIX	12
7	REFERENCES	13

1. **INTRODUCTION**

The Urban Property Management System (UPMS) is an innovative solution tailored to address the complexities of managing properties in urban environments. With its user-friendly interface and comprehensive features, UPMS facilitates efficient property management for administrators, property owners, and tenants alike. Users can securely log in using their email IDs and passwords, with role-based access control ensuring appropriate permissions for each user category. Property owners can easily list their flats with detailed descriptions and images, while tenants can browse and book available properties hassle-free. The system also simplifies rent and utility management, allowing tenants to conveniently handle payments online. Built on a solid foundation of four core tables, UPMS ensures data integrity and reliability. Overall, UPMS revolutionizes urban property management, enhancing transparency, accessibility, and efficiency for all stakeholders involved.

1.1 **PURPOSE**

The purpose of an Urban Property Management System is to streamline the management of properties within urban areas, offering a comprehensive platform for property owners, tenants, and administrators to efficiently handle various aspects of property ownership, rental, and administration.

1.2 **SCOPE**

The scope of an Urban Property Management System encompasses various functionalities and features aimed at facilitating the efficient management of urban properties. Here's a detailed scope:

1. User Authentication and Authorization:

- User roles include admin, property owner, and tenant.
- Authentication via email ID and password with JWT support.
- Role-based access control to ensure appropriate access levels.

2. User Management:

- User registration and profile management.
- Ability for users to update their profiles and manage their account settings.
- Property Listing and Management:
- Property owners can list their properties with details such as address, amenities, rental terms, and images.
- Ability to categorize properties based on type, location, size, etc.
- Property management features including editing, archiving, and deleting listings.

3. Tenant Management:

- Tenants can search and view available properties based on their preferences and requirements.
- Booking functionality for tenants to request property viewings and submit rental applications.
- Ability for property owners to review and approve rental applications.

4. Rent and Utility Payments:

- Online payment gateway integration for tenants to pay rent and utility bills securely.
- Tracking of rent payments and utility usage.
- Generation of payment receipts and invoices.

1.3 OBJECTIVE

The objective of an Urban Property Management System is to provide a comprehensive and efficient platform for managing urban properties, catering to the needs of property owners, tenants, and administrators. It aims to streamline property management processes, enhance user experience, and optimize financial performance.

By centralizing property listings, lease management, and tenant interactions, the system facilitates easier access to information and improves communication between stakeholders. It enables property owners to effectively market their properties, manage lease agreements, and track rental income and expenses.

For tenants, the system offers an intuitive interface for property search, rental application submission, and online rent payment, thereby simplifying the rental process and improving tenant satisfaction.

Administrators benefit from tools for monitoring property performance, generating reports, and ensuring compliance with regulatory requirements.

2. **REQUIREMENTS**

Functional Requirements :

FR 1. User Registration and Authentication:

- Users can register by providing necessary details.
- User authentication is required to access the system.
- Forgot password functionality allows users to reset their passwords.

FR 2. Profile Management :

- All users, including admins, owners, and regular users, should be able to update their profiles.
- Profile updates may include personal information such as name, contact details, and profile picture.

FR 3. Property Listing and Booking :

- Owners should be able to add flats to the system with details and images.
- Users can browse through the list of available flats and book them.
- Once booked, the flat should be marked as unavailable for further bookings.
- Users who have rented a flat can initiate rent payments and manage utility payments.

FR 4. Rent Utility Management:

- The system should maintain a rent utility table containing data such as rent amount, electric bill, water bill, etc.
- Users should be able to view and pay their rent and utility bills through the system.

Non Functional Requirements:

NFR 1. Security:

- User passwords are securely stored using encryption techniques.

- Access controls ensure that users can only access authorized features.

NFR 2. Performance:

- The system should handle a large number of simultaneous users without significant slowdowns.
- Image loading and retrieval should be efficient for a smooth user experience.

NFR 3. Scalability:

- The system should be designed to accommodate future growth and increased user activity.

NFR 4. Usability:

- The user interface should be intuitive and user-friendly for both customers and administrators.
- Clear and concise error messages should guide users through any issues.

NFR 5. Reliability:

- The system should be available and operational 24/7 with minimal downtime.

NFR 6. Data Integrity:

- Data integrity and consistency are maintained through proper validation and database design.

NFR 7. Data Privacy:

- User data, especially personal and sensitive information, should be stored securely.

Other Requirements:

Hardware and Network Interfaces:

Back-end Server Configuration:

- Intel Pentium-IV Processor
- 128 MB RAM
- 1 Raid Controller Card
- 32-bit Ethernet Controller (100 Base-T)
- 8 x 2.0 GB Fast SCSI/2 with Raid Support
- 2.88 MB FDD
- 48x CD ROM Drive
- SVGA Colour Monitor on PCI with 1MB RAM
- 101 Keys Keyboard
- 1 Microsoft Mouse with pad

- 4/8 GB DAT
- One Serial & Two Parallel Ports
- Internet Information Server (IIS)
- Microsoft Transaction Server (MTS)

Front-end Client Configuration:

- Intel Pentium-III @ 650 MHz Processor
- 128 MB SDRAM
- 10 GB Hard Disk Drive
- 1.44 MB Floppy Disk Drive
- 15" SVGA Digital Color Monitor
- One Serial, One Parallel port, and One USB port
- 104 Keys Keyboard
- PS2 Mouse with pad
- 32-bit PCI Ethernet Card
- 48X CD Drive

Software Interfaces:

Software configuration for back-end Services:

- Windows NT – Server 4.0
- SQL Server 7.0

Software configuration for front-end Services:

- Virus Protection Software
 - Client Workstation
 - Office 2000
- Web Browser – Internet Explorer/Netscape

3. DATABASE DESIGN

Database Design

The following table structures depict the database design.

Table 1: user:

```
mysql> desc users;
```

Field	Type	Null	Key	Default	Extra
user_type	varchar(31)	NO		NULL	
id	bigint	NO	PRI	NULL	auto_increment
address	varchar(255)	NO		NULL	
contact_no	varchar(255)	NO		NULL	
email_id	varchar(255)	NO		NULL	
name	varchar(255)	NO		NULL	
password	varchar(255)	NO		NULL	
role	varchar(255)	NO		NULL	

8 rows in set (0.00 sec)

Table 2: Flat:

```
mysql> desc flat;
```

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
vacant_status	bit(1)	NO		NULL	
address	varchar(255)	NO		NULL	
flat_no	int	NO		NULL	
floor_no	int	NO		NULL	
flat_type	varchar(255)	NO		NULL	
owner_id	bigint	YES	MUL	NULL	
tenant_id	bigint	YES	MUL	NULL	

8 rows in set (0.08 sec)

Table 3: Flat image:

```
mysql> desc flat_image;
```

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
image_data	longblob	NO		NULL	
flat_id	bigint	YES	MUL	NULL	

3 rows in set (0.00 sec)

Table 4: Utility:

```
mysql> desc utility;
```

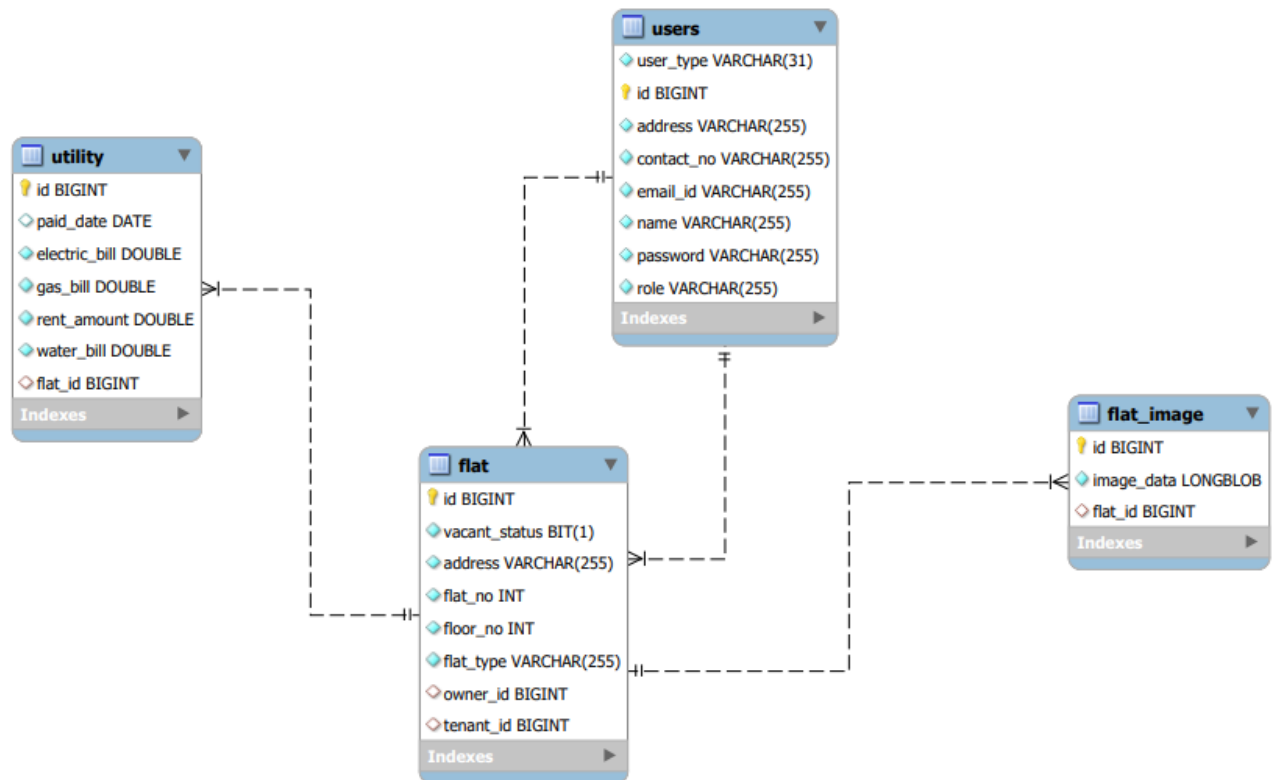
Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
paid_date	date	YES		NULL	
electric_bill	double	NO		NULL	
gas_bill	double	NO		NULL	
rent_amount	double	NO		NULL	
water_bill	double	NO		NULL	
flat_id	bigint	YES	MUL	NULL	

7 rows in set (0.00 sec)

```
mysql>
```

4. APPENDIX

Entity Relationship Diagram:



Use Case Diagram:

Urban Property Management Use Case Diagram



5.REFERENCES

<http://www.google.com>

<http://www.javatpoint.com/javatutorial>

<http://www.w3.org> <http://www.wikipedia.org>

<https://www.tutorialspoint.com/java>