**PGatEase - Business Requirement Specification (BRS)**

**1. Executive Summary**

PGatEase is a digital platform designed to simplify and organize the management of Paying Guest (PG) accommodations. It provides PG owners, especially multi-PG owners, with a centralized dashboard to manage tenant complaints, rent tracking, and announcements, while giving tenants a transparent, structured communication channel, customizable food menu, and laundry services.

**Owner Features:**

* **Multi-Property Management**: Handle multiple PG properties from a single dashboard.
* **Complaint Management System**: View, track, and resolve tenant complaints easily.
* **Rent and Payment Tracking**: Monitor and update rent payment statuses.
* **Announcement and Notice Board**: Share important updates directly with tenants.
* **Customizable Food Menu Management**: Upload and manage weekly/monthly food menus for tenant viewing.
* **Analytics Dashboard (Future Version)**: View trends on tenant satisfaction, complaints, and payments.

**Tenant Features:**

* **Raise and Track Complaints**: Tenants can submit issues and monitor resolution status.
* **Rent Payment History View**: Tenants can check past payments and pending dues.
* **View Announcements**: Access all official notices and updates from PG owners.
* **Customize Food Plans**: Choose preferred meal options weekly (where applicable).
* **Laundry Service Requests**: Submit and track laundry service requests.

**2. Business Objectives**

* **Digitize** and organize PG management operations traditionally done manually.
* **Increase efficiency** for PG owners handling multiple properties.
* **Improve tenant satisfaction** by ensuring faster complaint resolution, better communication, and better food options.
* **Acquire 100+ PG owners** within 6 months of MVP launch in Hyderabad.
* **Build brand trust** among PG owners and residents for future expansion into metro cities.

**3. Problem Statement**

* PG owners managing multiple properties often struggle with tracking tenant complaints, rent payments, and sending timely notices.
* Manual record-keeping leads to errors, delays, and tenant dissatisfaction.
* Tenants feel unheard and disconnected when complaints or dissatisfaction with food and laundry services occur.
* There is a lack of affordable, easy-to-use software specifically focused on mid-sized, unorganized PGs.

**4. Solution Overview**

PGatEase provides:

* A centralized **Owner Dashboard** to manage all PGs, tenants, payments, and complaints.
* A **Tenant Portal** to raise complaints, view payment history, receive updates, customize their food plan (weekly), and laundry services.
* A **Complaint Tracking System** with real-time status updates (Pending ➔ In Progress ➔ Resolved).
* A **Payment Management Feature** to record and view rent payments.
* A **Notice Board** for announcements and important updates.

The platform is designed to be simple, mobile-friendly, and cost-effective.

**5. Scope**

**In Scope for MVP:**

* Owner dashboard for complaint, rent management.
* Tenant portal for raising complaints, viewing information.
* Authentication system for owners and tenants.
* Announcement/Notice posting.
* Manual payment status tracking.

**Out of Scope for MVP (Future versions):**

* Online rent payment gateway integration.
* Staff management module.
* Mobile app version (initially web only).

**6. Key Stakeholders**

* **Founders**: Asif and Anas Faaiz.
* **Primary Customers**: Multi-PG owners.
* **Secondary Users**: Tenants (students, working professionals).
* **Future Stakeholders**: Mentors, investors, real estate agents.

**7. Target Audience**

* PG owners managing more than one property with 100–500 beds.
* Student and working professional tenants aged 18–30.
* Initially targeting the Hyderabad market, later expanding to other metro cities.

**8. Business Requirements**

The platform must fulfill the following high-level business needs:

* Allow PG owners to view and manage multiple PG properties from a single dashboard.
* Enable tenants to submit and track complaints easily.
* Tenants should be able to select their preferable food options.
* Allow owners to track rent payments for each tenant.
* Provide a notice board for quick communication to all tenants.
* Ensure data privacy and basic security compliance for sensitive information.

**9. Constraints and Assumptions**

**Constraints:**

* Limited budget for MVP development and hosting.

**Assumptions:**

* PG owners are familiar with basic smartphone/computer usage.
* Tenants are willing to engage with digital complaint systems for faster resolution.
* Internet connectivity is available at PG premises.

**10. Functional Requirements**

* **User Authentication**: Owners and tenants will be able to register and log in to the platform securely.
* **Dashboard Features**: PG owners should be able to manage multiple PG properties, track complaints, payments, and post announcements.
* **Complaint Management**: Tenants can raise complaints, and owners can track and resolve them.
* **Food Plan Management**: Owners can upload and manage food menus; tenants can choose their preferences.
* **Payment Tracking**: Owners can track rent payments; tenants can view past payment history and pending dues.
* **Notice Board**: Owners can post announcements; tenants can view them in real time.
* **Real-Time Notifications**: Tenants will be notified about complaint resolutions, rent dues, and announcements.

**11. Non-Functional Requirements**

* **Performance**: The platform should be able to handle at least 100 PG properties with 500 tenants per property without degradation in performance.
* **Security**: Data encryption, secure login, and authentication mechanisms must be implemented.
* **Usability**: The platform must be simple, intuitive, and mobile-friendly for both PG owners and tenants.
* **Scalability**: The platform should be scalable to handle increased user traffic as the platform grows.

**12. Tech Stack**

The tech stack for PGatEase will include the following technologies:

**Frontend:**

* **NextJs**: For building the user interface of both the Owner Dashboard and Tenant Portal.
* **React Router**: For managing navigation and routing in the application.

**Backend:**

* **Node.js (with Express)**: For building the server-side logic, API management, and handling user authentication.
* **JWT (JSON Web Tokens)**: For secure user authentication and session management.
* **REST API**: For handling client-server communication between the frontend and backend.

**Database:**

* **MySQL or PostgreSQL**: For storing user data, complaints, payments, and announcements.
* **Sequelize (if using Node.js)** or **Django ORM**: For interacting with the database.

**Hosting & Deployment:**

* **AWS or Heroku**: For hosting and scaling the web platform.
* **Docker**: For containerization and deployment, ensuring consistency across environments.

**Third-Party Integrations:**

* **Twilio or SendGrid**: For SMS and email notifications to tenants for important updates like complaint resolution and rent payment reminders.
* **Payment Gateway (Future Version)**: For online rent payment processing (e.g., Stripe, Razorpay).

**13. Future Enhancements**

* **Online Rent Payment Gateway Integration**: To facilitate seamless online payments.
* **Mobile App Version**: Expanding to mobile platforms for increased accessibility.
* **Staff Management**: Allow owners to manage staff across multiple properties.
* **Advanced Analytics Dashboard**: Detailed insights into tenant satisfaction, food quality, complaint trends, etc.