Business Requirements Document (BRD)

Project Title: ChitLink A Transparent Chit Fund Management Platform

1. Project Overview

ChitLink is an innovative digital platform designed to enhance the transparency and efficiency of traditional savings and borrowing groups within communities. By providing a secure and user-friendly interface, ChitLink automates the management of participants and fund allocations, ensuring that all members have real-time access to the group's financial status.

2. Objectives

1. **Boost User Engagement**: Achieve a 70% participation rate among new users within their first month.
2. **Maximize Chit Fund Success**: Ensure 90% of created chit funds reach their completion with successful payouts.
3. **Elevate User Satisfaction**: Target an average user satisfaction score of 4 out of 5 through continuous improvement.
4. **Maintain System Reliability**: Aim for 99.9% system uptime to provide a consistent and secure user experience

3. Scope

This project will include:

- **Customer-facing features:** shop search, order placement, measurement management, notifications, chat with tailors, order tracking, reviews and ratings**.**
- **Tailor-facing features:** order management, customer notifications, revenue tracking, shop status (open/closed), and chat functionality**.**
- **Common features:** tailor shop map discovery using Open Maps, notifications for both parties, and optional delivery or pickup services**.**

4. Key Features and Requirements

1. **Role-Based Access Control (RBAC)**:
    - **Admin**: Full control over the platform, managing user registrations, KYC approvals, and overseeing chit fund activities.
    - **Chit Creator (Organizer)**: Empowered to create and manage chit funds, set rules, and engage participants effectively.
    - **Participant**: Engages actively in chit funds by making contributions and participating in bidding processes.
    - **Guest**: Can view available chit funds and register to participate.
2. **User-Friendly Interface**:
    - Tailored dashboards for each role to provide easy access to relevant functionalities and information.
3. **Secure Transactions**:
    - Robust security measures to protect user data and ensure safe financial transactions.
4. **Analytics and Reporting**:

    - Admins have access to comprehensive reports and analytics to monitor activities, resolve disputes, and ensure compliance.
5. **Notification System**:
    - Automated alerts for participants regarding contributions, bidding events, and important updates.

5. Functional Requirements

User Service

- **User Registration:** Users must be able to register with their name, email, phone, and address.
- **Profile Management:** Users can update their profile information, including KYC documents and role changes.
- **Role Management:** Admins can assign or change user roles (Chit Creator, Participant, Admin).
- **Authentication:** Users must be able to log in and authenticate their sessions.

Chit Fund Service

- **Chit Fund Creation:** Chit creators can create chit funds with defined rules (total amount, duration, etc.).
- **Manage Participants:** Chit creators can view and manage participants and their contributions.
- **Join Chit Fund:** Participants can view and join available chit funds.
- **Contribution Tracking:** System must track contributions made by participants.

Payment Service

- **Payment Processing:** Users can make payments for their contributions securely.
- **Payment Status:** The system must provide updates on payment status (e.g., pending, completed).
- **Transaction History:** Users should be able to view their payment history related to chit funds.

Bid Service

- **Bid Submission:** Participants can place bids on ongoing chit funds.
- **Bid Management:** Users can view, update, or cancel their bids.

Notification Service

- **Notification Creation:** The system must generate notifications for important events (e.g., new fund created, bid placed).
- **User Alerts:** Users receive alerts about their activities, such as payment confirmations and bid status updates.
- **Notification Management:** Users can view and manage their notifications.

6. Non-Functional Requirements

Performance

- **Response Time:** API requests should have a response time of less than 200 ms for 95% of requests.
- **Throughput:** The system should support at least 1,000 concurrent users without performance degradation.

Scalability

- **Horizontal Scalability:** The microservices architecture should allow for easy scaling of individual services based on load.
- **Load Balancing:** Implement load balancing to distribute incoming requests across multiple service instances.

Security

- **Data Encryption:** All sensitive data (e.g., personal information, payment details) must be encrypted in transit and at rest.
- **Access Control:** Implement role-based access control (RBAC) to restrict access to features based on user roles.
- **Authentication:** Use secure authentication methods (e.g., OAuth 2.0, JWT) for user sessions.

Reliability

- **Availability:** The system should have an uptime of 99.9% or greater, ensuring availability of services.
- **Error Handling:** Implement comprehensive error handling and logging for all microservices.

Usability

- **User Interface:** The application should have a user-friendly interface that is easy to navigate for all user roles.
- **Documentation:** Provide comprehensive API documentation and user manuals for end-users.

7.User Stories

For Participants

- **As a participant**, I want to register for an account so that I can join chit funds.
- **As a participant**, I want to view available chit funds so that I can choose one to join.
- **As a participant**, I want to make contributions securely so that my financial information is protected.
- **As a participant**, I want to receive notifications about my contributions and bid status so that I stay informed.

For Chit Fund Creators

- **As a chit fund creator**, I want to create new chit funds with specific rules so that I can manage contributions effectively.
- **As a chit fund creator**, I want to view and manage participants in my chit funds so that I can track their contributions.
- **As a chit fund creator**, I want to communicate updates to participants so that they are aware of any changes.

8. Stakeholders

For Participants

1. As a participant, I want to register an account so that I can join chit funds.
2. As a participant, I want to view available chit funds so that I can decide which ones to join.
3. As a participant, I want to make contributions to my chosen chit funds securely so that my financial information is protected.
4. As a participant, I want to place bids on ongoing chit funds so that I can compete for a share of the fund.
5. As a participant, I want to receive notifications about my contributions and bid status so that I stay informed.

For Chit Fund Creators

1. As a chit fund creator, I want to create new chit funds so that I can attract participants and manage contributions.
2. As a chit fund creator, I want to set rules for my chit fund so that participants understand the terms of engagement.
3. As a chit fund creator, I want to view participants and their contributions so that I can manage the fund effectively.
4. As a chit fund creator, I want to communicate updates and changes to my participants so that everyone is informed.

For Admins

1. As an admin, I want to manage user accounts so that I can ensure proper role assignments and handle issues.
2. As an admin, I want to monitor chit fund activities so that I can address disputes or irregularities promptly.
3. As an admin, I want to generate reports on user activities and fund performance so that I can analyze system health.

9. Success Metrics

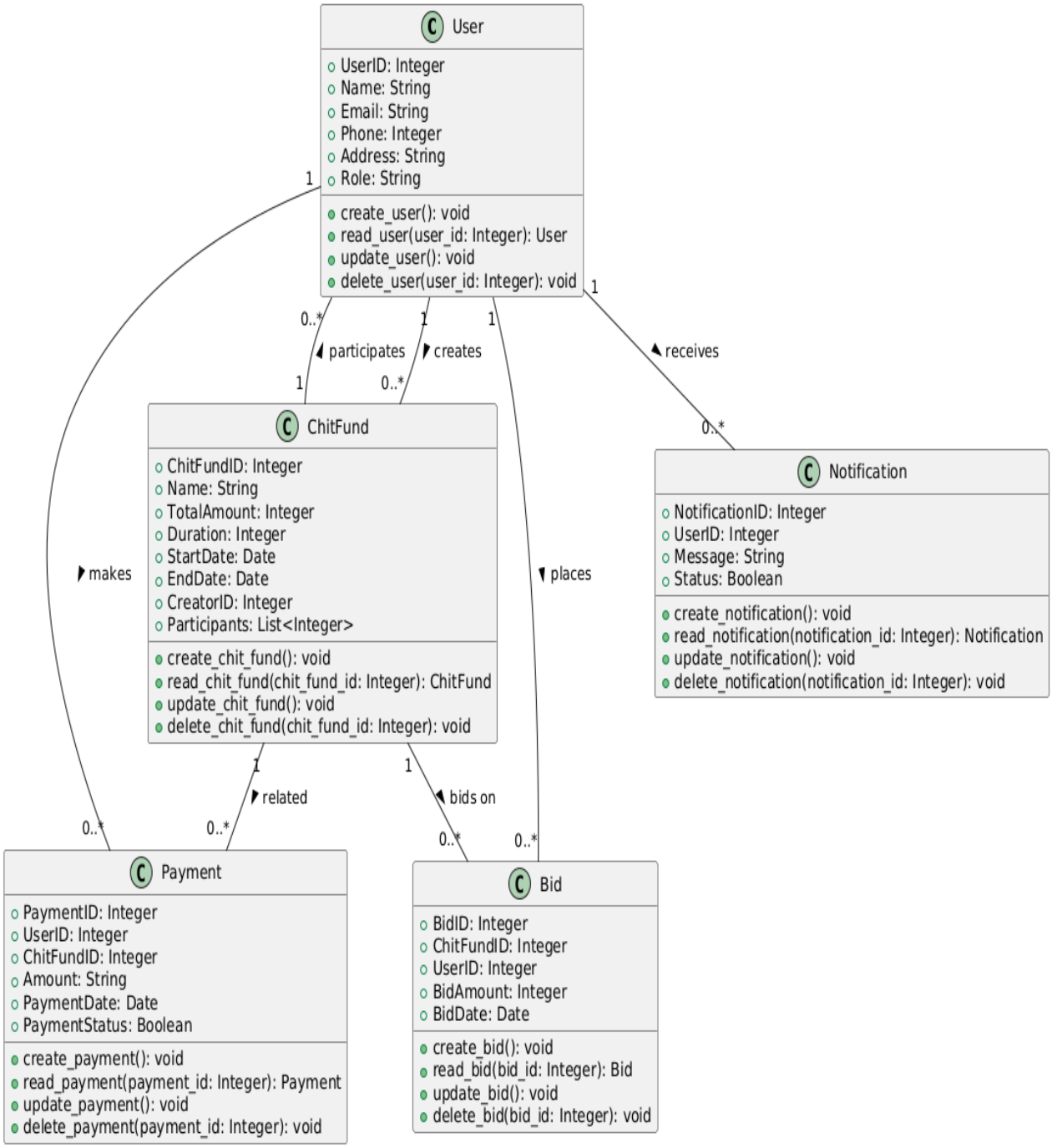
User Adoption Rate: Aim for 70% of registered users actively participating in chit funds within the first month.

Chit Fund Completion Rate: Target 90% of chit funds successfully completing their terms with payouts to participants.

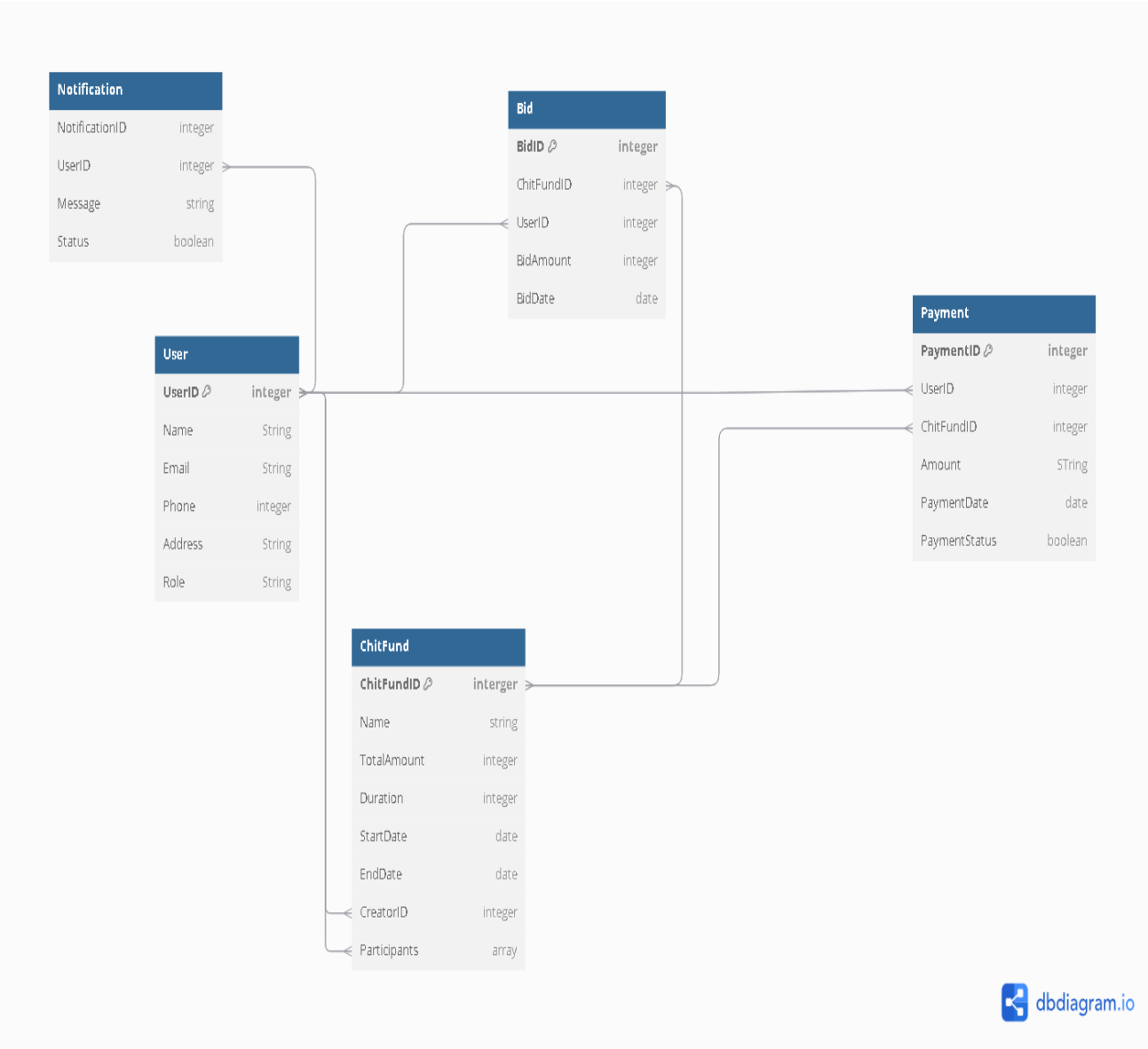
User Satisfaction Score: Achieve an average satisfaction score of 4 out of 5 from participants and chit fund creators.

System Uptime: Maintain a system uptime of 99.9% to ensure reliability and availability for users.

10. UML Diagram



11. Schema Diagram



12. API DOCUMENTATION

User Service

- **POST** /api/users - Create a new user
- **GET** /api/users/{user_id} - Retrieve user details by UserID
- **PUT** /api/users/{user_id} - Update user information
- **DELETE** /api/users/{user_id} - Delete a user by UserID

ChitFund Service

- **POST** /api/chitfunds - Create a new chit fund
- **GET** /api/chitfunds/{chit_fund_id} - Retrieve chit fund details by ChitFundID
- **PUT** /api/chitfunds/{chit_fund_id} - Update chit fund information
- **DELETE** /api/chitfunds/{chit_fund_id} - Delete a chit fund by ChitFundID

Payment Service

- **POST** /api/payments - Create a new payment
- **GET** /api/payments/{payment_id} - Retrieve payment details by PaymentID
- **PUT** /api/payments/{payment_id} - Update payment information
- **DELETE** /api/payments/{payment_id} - Delete a payment by PaymentID

Bid Service

- **POST** /api/bids - Create a new bid
- **GET** /api/bids/{bid_id} - Retrieve bid details by BidID
- **PUT** /api/bids/{bid_id} - Update bid information
- **DELETE** /api/bids/{bid_id} - Delete a bid by BidID

Notification Service

- **POST** /api/notifications - Create a new notification
- **GET** /api/notifications/{notification_id} - Retrieve notification details by NotificationID
- **PUT** /api/notifications/{notification_id} - Update notification status or message
- **DELETE** /api/notifications/{notification_id} - Delete a notification by NotificationID