

# PartyPulse Design Document

**Date:** 10/22/2024

**Version:** 1.0

**Authors:** Caden Parsons, Y'Vin Kpa(we filled in his info), Daisy Ibuoka

## Use-Case Model

### Actors:

- **User (Gamer)** - Caden Parsons
- **Provider (Party Space Creator)** - Y'Vin Kpa
- **SysAdmin (Administrator)** - Daisy Ibuoka

### Use Cases:

#### User (Gamer)

1. **Create Profile**
  - Description: User creates a profile to join the platform.
2. **Modify Profile**
  - Description: User updates their profile information.
3. **Find PartySpaces**
  - Description: User searches for available PartySpaces based on game preferences.
4. **Join PartySpace**
  - Description: User joins a selected PartySpace.
5. **Report Issue**
  - Description: User reports a bug or issue within the app.

#### Provider (Party Space Creator)

1. **Create Party Space**
  - Description: Provider creates a new PartySpace for a game.
2. **Manage Party Members**
  - Description: Provider manages members in their PartySpace, including banning users.
3. **Manage Party Settings**
  - Description: Provider updates settings for their PartySpace.

## SysAdmin (Administrator)

### 1. Manage User Access

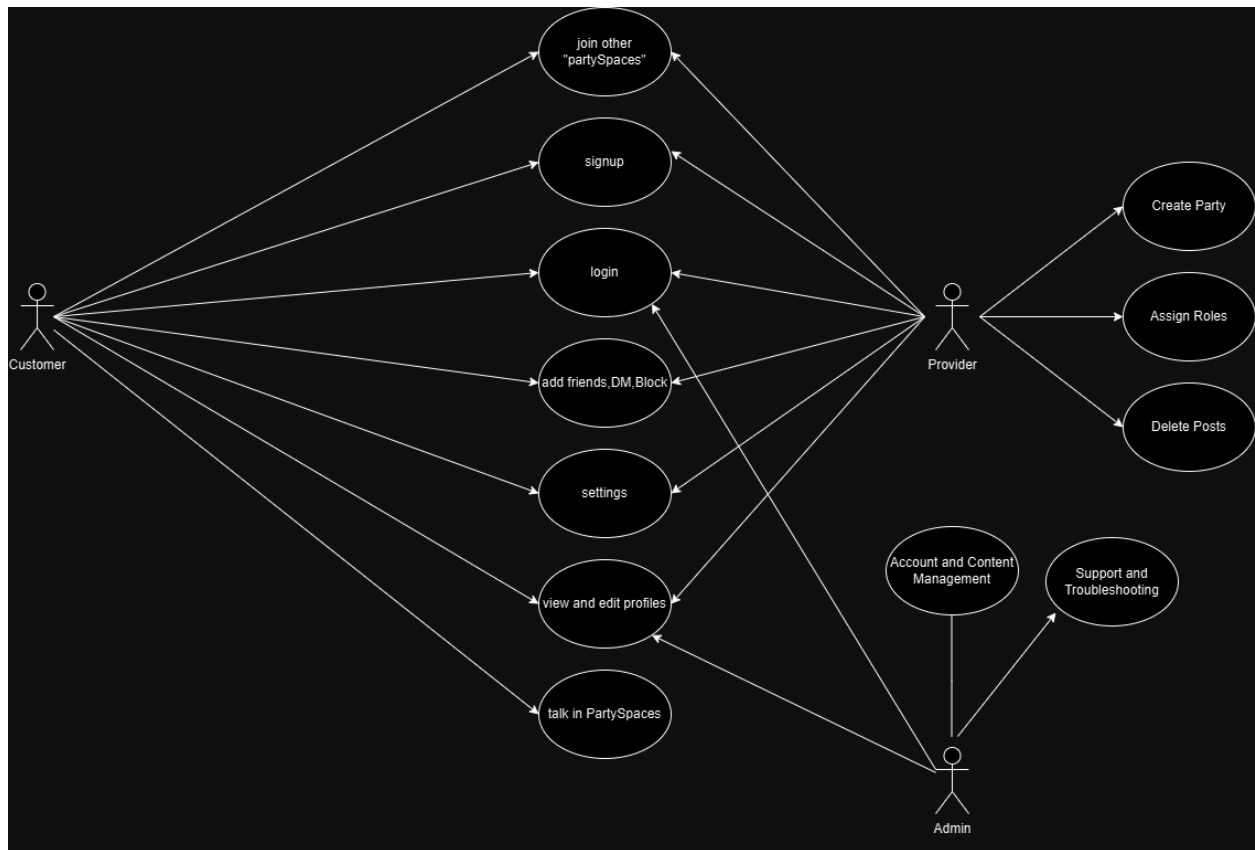
- Description: Admin reviews and bans users for inappropriate behavior.

### 2. Moderate Content

- Description: Admin removes flagged profiles and inappropriate reviews.

### 3. View Analytics

- Description: Admin views usage statistics of the application.



## 2. State Machine Diagram

### User (Gamer) State Diagram

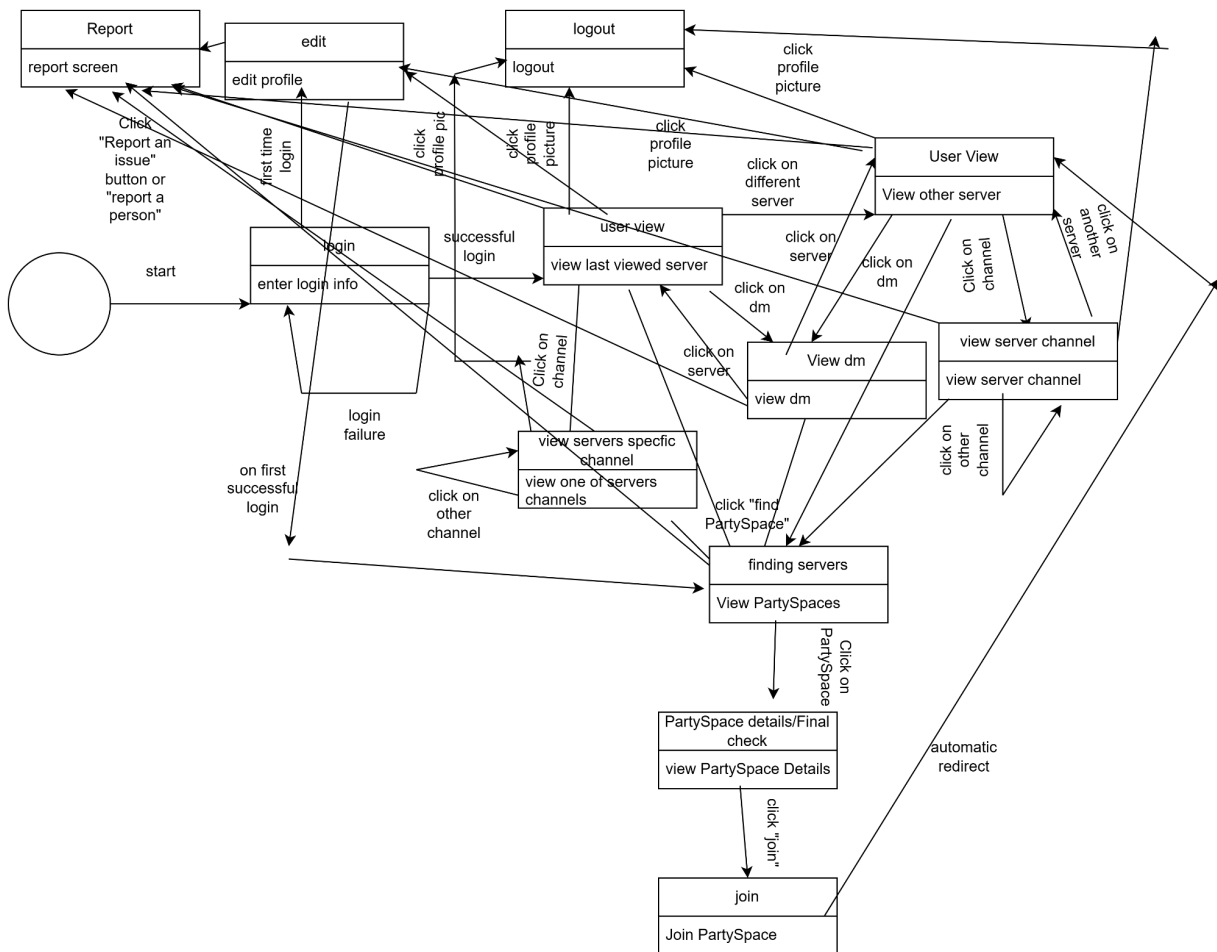
States:

1. Login
2. Profile Creation
3. Profile Editing

4. **PartySpace Search**
5. **Join Confirmation**
6. **Report Issue**

### Transitions:

- **Login → Profile Creation:** User logs in and is prompted to create a profile.
- **Profile Creation → Profile Editing:** User completes profile creation and navigates to edit their profile.
- **Profile Editing → PartySpace Search:** User saves changes and searches for PartySpaces.
- **PartySpace Search → Join Confirmation:** User selects a PartySpace to join.
- **Report Issue → PartySpace Search:** User reports an issue and returns to PartySpace search.



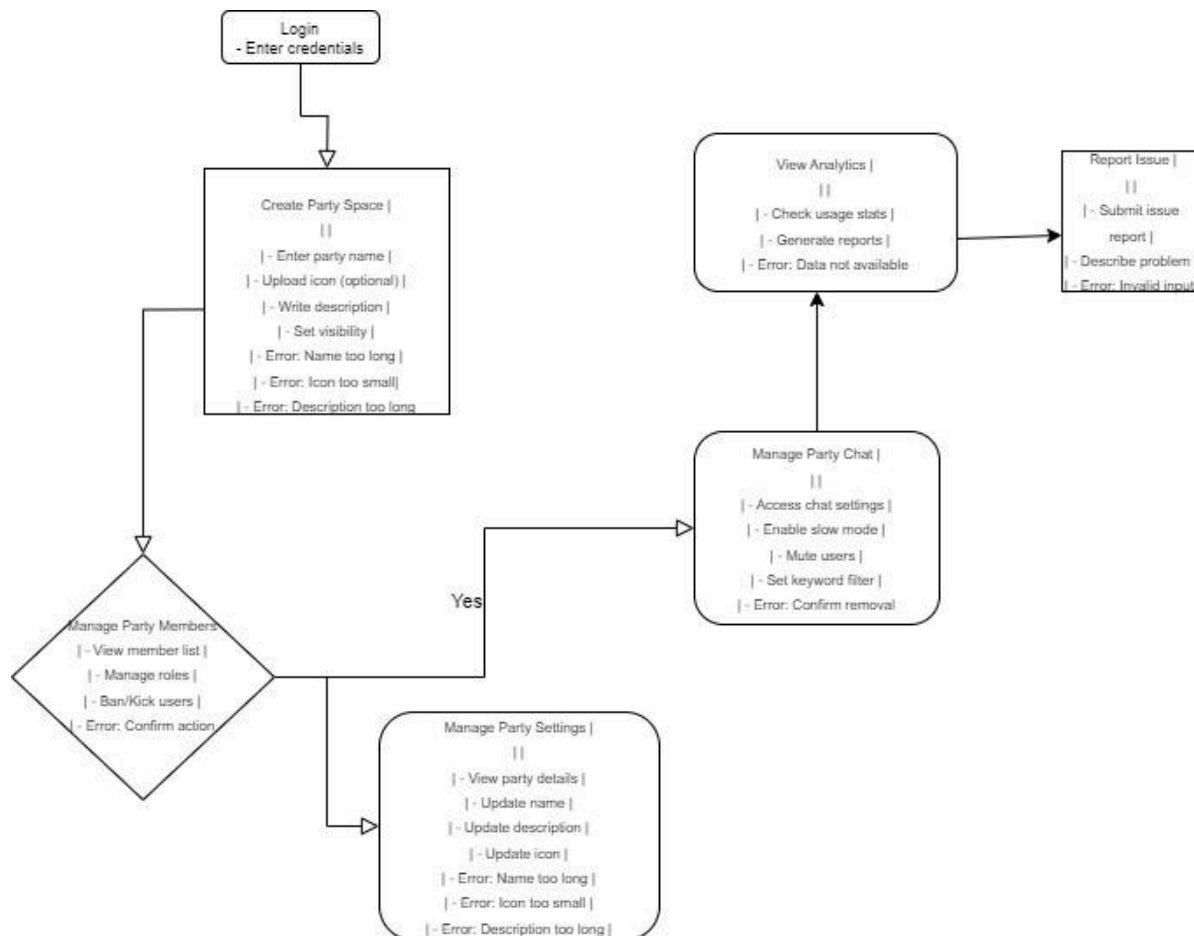
## State Machine Diagram for Provider (Party Space Creator)

### States:

- Login
- Create Party Space
- Manage Party Members
- Manage Party Settings
- View Analytics
- Report Issue

### Transitions:

- Login → Create Party Space: User logs in and creates a new Party Space.
- Create Party Space → Manage Party Members: User creates a Party Space and then manages its members.
- Manage Party Members → Manage Party Settings: User navigates to change settings for the Party Space.
- Manage Party Settings → View Analytics: User checks analytics after updating settings.
- Report Issue → View Analytics: User reports an issue and returns to view analytics.



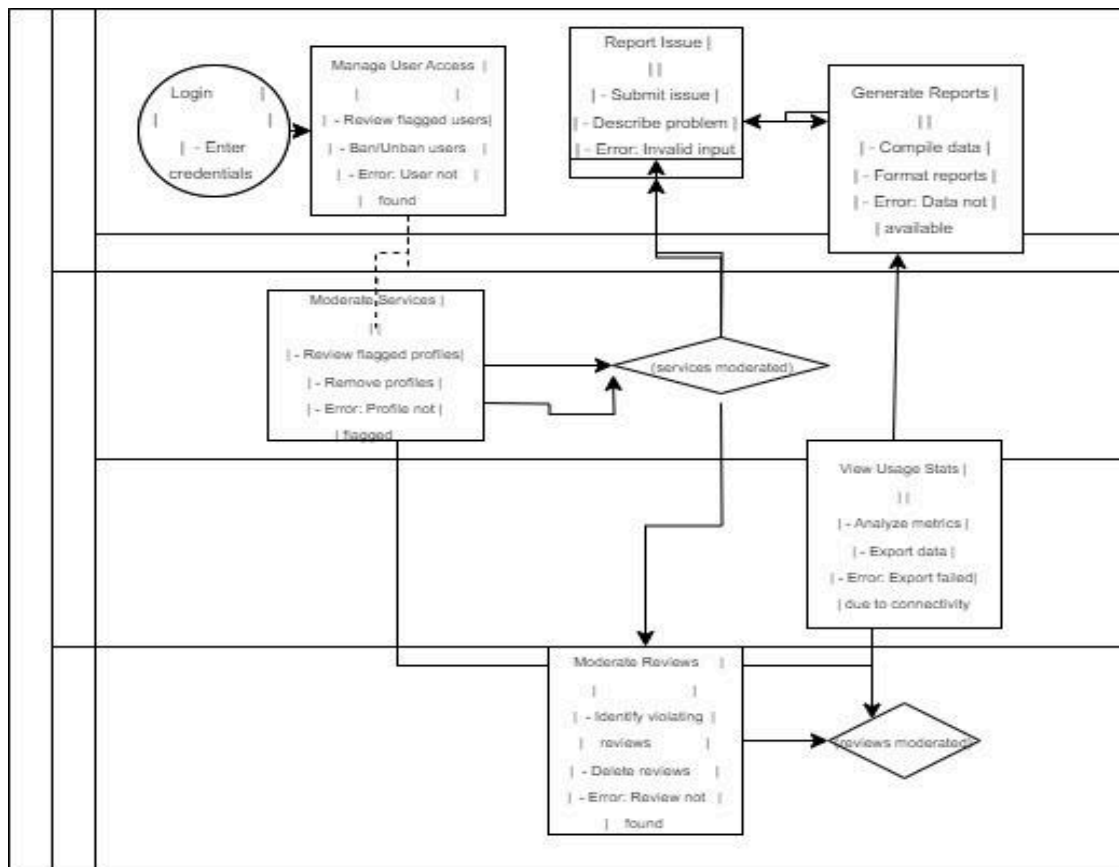
## State Machine Diagram for SysAdmin (Administrator)

### States:

- Login
- Manage User Access
- Moderate Content
- View Analytics
- Generate Reports
- Report Issue

### Transitions:

- Login → Manage User Access: Admin logs in and reviews user access.
- Manage User Access → Moderate Content: Admin reviews and moderates content.
- Moderate Content → View Analytics: Admin checks usage statistics after moderation.
- View Analytics → Generate Reports: Admin generates reports based on analytics.
- Report Issue → View Analytics: Admin reports an issue and returns to analytics.



### 3. Database Schema

#### Schema Diagram

##### Tables:

##### 1. Users

Field	Type	Constraints
user_id	INT	PK
username	VARCHAR	UNIQUE
password	VARCHAR	
profile_info	TEXT	

##### 2. PartySpaces

Field	Type	Constraints
party_id	INT	PK
creator_id	INT	FK for Users(user_id)
game_title	VARCHAR	
description	TEXT	
visibility	ENUM	('public', 'private')

##### 3. Memberships

Field	Type	Constraints
membership_id	INT	PK
user_id	INT	FK for Users(user_id)

Field	Type	Constraints
membership_id	INT	PK
party_id	INT	FK to PartySpaces(party_id)
role	ENUM	('member', 'admin', 'moderator')

#### 4. Reports

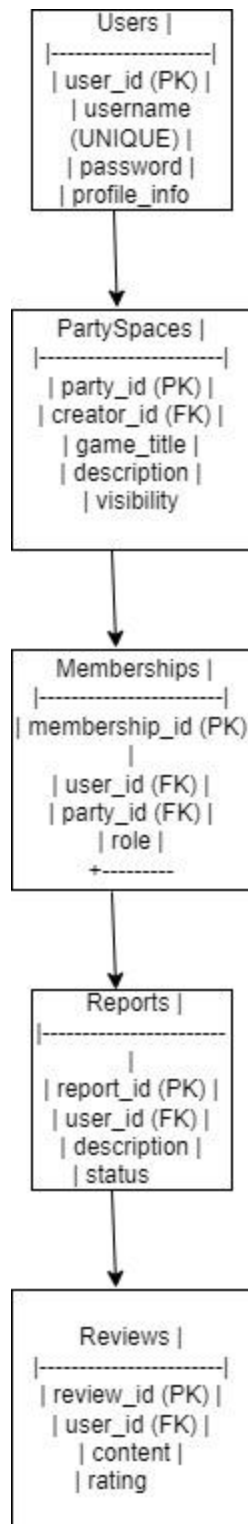
Field	Type	Constraints
report_id	INT	PK
user_id	INT	FK for Users(user_id)
description	TEXT	
status	ENUM	('pending', 'resolved')

#### 5. Reviews

Field	Type	Constraints
review_id	INT	PK
user_id	INT	FK for Users(user_id)
content	TEXT	
rating	INT	

### Relationships

- **Users** can create multiple **PartySpaces**.
- **PartySpaces** can have multiple **Memberships**.
- **Users** can report issues and provide **Reviews**.
- **Admins** can manage **Reports** based on user activity.





## 4. Software Architecture

### MVC Architecture Diagram

- **Models:**
  - **UserModel**
  - **PartySpaceModel**
  - **MembershipModel**
  - **ReportModel**
  - **ReviewModel**
- **Controllers:**
  - **UserController**
  - **PartySpaceController**
  - **MembershipController**
  - **ReportController**
  - **ReviewController**
- **Views:**
  - **UserProfileView**
  - **PartySpaceView**
  - **MembershipView**
  - **ReportView**
  - **ReviewView**

### Relationships

- The **UserController** interacts with **UserModel** to handle user-related actions.
- The **PartySpaceController** manages **PartySpaceModel** for creating and managing PartySpaces.
- **MembershipController** links users to PartySpaces via **MembershipModel**.
- **ReportController** oversees user reports through **ReportModel**.
- **ReviewController** handles reviews through **ReviewModel**.

