**Tabletop Reserve Database Schema**

**Overview**

This document provides a comprehensive reference of the Firebase Firestore database schema for the Tabletop Reserve application. The schema includes collections, documents, fields, data types, and relationships between different entities.

**Collections**

**Users**

Stores user accounts with admin role.

| **Field** | **Type** | **Description** |
| --- | --- | --- |
| name | String | Full name of the admin user |
| email | String | Email address of the admin user |
| role | String | User role, value is "admin" for this collection |
| createdAt | Timestamp | When the user was created |
| createdBy | String | Email or UID of the admin who created this user |

**Stores**

Stores shop owner accounts and shop information.

| **Field** | **Type** | **Description** |
| --- | --- | --- |
| storeName | String | Name of the shop |
| ownerName | String | Name of the shop owner |
| email | String | Email address of the shop owner |
| address | String | Street address of the shop |
| city | String | City/town of the shop |
| county | String | County of the shop |
| postCode | String | Postal code of the shop |
| phoneNumber | String | Contact phone number |
| role | String | User role, value is "shop" |
| isApproved | Boolean | Whether the shop is approved by an admin |
| registrationStatus | String | Status of registration: "pending", "approved", "rejected" |
| description | String | Shop description |
| shopType | String | Type of shop (e.g., "game-store", "dedicated-tables", "cafe") |
| website | String | Shop website URL |
| socialMedia | String | Social media links |
| amenities | Object | Available amenities |
| amenities.wifi | Boolean | Whether Wi-Fi is available |
| amenities.food | Boolean | Whether food is available |
| amenities.drinks | Boolean | Whether drinks are available |
| amenities.parking | Boolean | Whether parking is available |
| amenities.accessible | Boolean | Whether the location is wheelchair accessible |
| amenities.gameLibrary | Boolean | Whether a game library is available |
| paymentMethods | Object | Accepted payment methods |
| paymentMethods.cash | Boolean | Whether cash is accepted |
| paymentMethods.credit | Boolean | Whether credit cards are accepted |
| paymentMethods.debit | Boolean | Whether debit cards are accepted |
| paymentMethods.mobile | Boolean | Whether mobile payments are accepted |
| specialty | String | Shop's game specialty |
| createdAt | Timestamp | When the shop account was created |
| updatedAt | Timestamp | When the shop information was last updated |
| approvedAt | Timestamp | When the shop was approved |
| approvedBy | String | Email or UID of the admin who approved the shop |
| rejectedAt | Timestamp | When the shop was rejected |
| rejectedBy | String | Email or UID of the admin who rejected the shop |

**Tables**

Stores information about gaming tables available for reservation.

| **Field** | **Type** | **Description** |
| --- | --- | --- |
| shopId | String | Reference to the store document ID |
| tableNumber | Number | Identifier for the table in the shop |
| capacity | Number | Number of people that can be seated |
| tableType | String | Type of table (e.g., "standard", "rpg", "miniatures") |
| description | String | Description of the table |
| specializedGame | String | Specific game the table is set up for |
| features | Array<String> | List of table features |
| hourlyRate | Number | Cost per hour (optional) |
| minBookingHours | Number | Minimum booking duration |
| isActive | Boolean | Whether the table is available for booking |
| createdAt | Timestamp | When the table was added |
| updatedAt | Timestamp | When the table information was last updated |

**Reservations**

Stores booking information for tables.

| **Field** | **Type** | **Description** |
| --- | --- | --- |
| shopId | String | Reference to the store document ID |
| tableId | String | Reference to the table document ID |
| tableNumber | Number | Table number (denormalized for efficiency) |
| reservationTime | Timestamp | Date and time of the reservation |
| duration | Number | Duration in hours |
| status | String | Status: "pending", "confirmed", "completed", "canceled" |
| customerName | String | Name of the customer |
| customerPhone | String | Phone number of the customer |
| customerEmail | String | Email of the customer (optional) |
| partySize | Number | Number of people in the reservation |
| notes | String | Customer notes or requests |
| shopNotes | String | Internal notes for shop staff |
| createdAt | Timestamp | When the reservation was created |
| createdBy | String | Who created the reservation ("shop" or customer ID) |
| updatedAt | Timestamp | When the reservation was last updated |
| confirmedAt | Timestamp | When the reservation was confirmed |
| confirmedBy | String | Who confirmed the reservation |
| completedAt | Timestamp | When the reservation was marked as completed |
| completedBy | String | Who marked the reservation as completed |
| canceledAt | Timestamp | When the reservation was canceled |
| canceledBy | String | Who canceled the reservation |
| cancellationReason | String | Reason for cancellation |

**SystemSettings**

Stores application-wide settings.

Document ID: appSettings

| **Field** | **Type** | **Description** |
| --- | --- | --- |
| autoApproveShops | Boolean | Whether to automatically approve new shop registrations |
| emailNotifications | Boolean | Whether to send email notifications for new shop registrations |
| maintenanceMode | Boolean | Whether the shop registration is disabled for maintenance |
| requiredFields | Object | Fields required for shop registration |
| requiredFields.phone | Boolean | Whether phone number is required |
| requiredFields.address | Boolean | Whether complete address is required |
| requiredFields.owner | Boolean | Whether owner name is required |
| registrationTimeLimit | Number | Time limit for shop registration process (in minutes) |
| sessionTimeout | Number | Time in minutes before admin is logged out due to inactivity |
| twoFactorAuth | Boolean | Whether two-factor authentication is required for admin login |
| failedLoginLimit | Number | Number of failed login attempts before temporary lockout |
| updatedAt | Timestamp | When settings were last updated |
| updatedBy | String | Who updated the settings |

**Relationships**

1. **Stores to Tables**:
   * One-to-many relationship
   * A store can have multiple tables
   * Tables reference their parent store via shopId
2. **Tables to Reservations**:
   * One-to-many relationship
   * A table can have multiple reservations (at different times)
   * Reservations reference their table via tableId
3. **Stores to Reservations**:
   * One-to-many relationship
   * A store can have multiple reservations
   * Reservations reference their store via shopId

**Authentication**

Firebase Authentication is used for user management. Each user in Firebase Auth corresponds to either:

* A document in the Users collection (for admin users)
* A document in the Stores collection (for shop owners)

The document ID in the respective collection matches the Firebase Auth UID.

**Notes**

1. **Denormalization**: Some data is intentionally denormalized (duplicated) for efficiency, such as the tableNumber in the Reservations collection.
2. **Status Tracking**: Multiple timestamps are used to track the lifecycle of entities (created, updated, approved, completed, etc.).
3. **Document IDs**:
   * User/Store document IDs match Firebase Auth UIDs
   * Table and Reservation IDs are auto-generated by Firestore
   * SystemSettings uses a fixed document ID "appSettings"
4. **Security Rules**: Implement Firestore security rules to ensure:
   * Admins can access all collections
   * Shop owners can only access their own data
   * Shops can't modify approval status