

GrouMapアプリ 詳細設計仕様書

=====

目次

1. 要件定義
2. データベース設計 (Firestore)
3. UI/UX設計
4. 技術仕様
5. Cloud Functions仕様
6. Firebase Security Rules
7. テスト戦略
8. 権限制限実装仕様
9. レベル・XPシステム仕様
10. 実装雛形生成仕様
11. 実行計画書

=====

1. 要件定義

=====

機能要件

アプリ種別:

1. ユーザーアプリ
 - 店舗検索・発見機能
 - ポイント・スタンプ収集機能
 - クーポン取得・利用機能
 - 投稿閲覧機能
2. 店舗用アプリ
 - 店舗情報管理
 - 投稿・クーポン作成機能
 - 分析ダッシュボード
 - スタッフ管理機能
 - 会社管理者機能 (isCompanyAdmin: true の場合)
 - 全社お知らせ作成・管理
 - フィードバック管理
 - メンテナンス告知

権限管理システム:

- 一般ユーザー: 基本機能のみ
- 店舗オーナー: 自店舗管理 + staffIds による複数スタッフ管理
- 会社管理者: 全社機能 + isCompanyAdmin: true フラグ

=====

2. データベース設計 (Firestore)

=====

コレクション構造:

/users/{uid}

```
{
  "displayName": "string",
  "bio": "string",
  "address": {
    "prefecture": "string",
    "city": "string",
    "addressLine": "string"
  }
}
```

```

    },
    "photoUrl": "string",
    "level": "number",           // ユーザーレベル (1~100)
    "currentXP": "number",       // 現在の総経験値
    "levelXP": "number",         // 現在レベルでの経験値
    "nextLevelXP": "number",     // 次レベルまでに必要な経験値
    "stampTotal": "number",
    "authProvider": "email|google|apple",
    "gender": "male|female|other|unknown",
    "ageBand": "teens|20s|30s|40s|50s|60s+|unknown",
    "readAnnouncements": ["string"],
    "referralCode": "string",
    "invitedByCode": "string",
    "storeReferralCode": "string",
    "accountCreatedAt": "timestamp",
    "lastLoginAt": "timestamp",
    "isCompanyAdmin": "boolean"
  }
}

/stores/{storeId}
{
  "name": "string",
  "iconUrl": "string",
  "heroImageUrl": "string",
  "categories": ["string"],
  "location": "GeoPoint",
  "timezone": "Asia/Tokyo",
  "hours": {
    "regular": {
      "mon": [{"open": "09:00", "close": "18:00"}],
      "tue": [{"open": "09:00", "close": "18:00"}]
    },
    "exceptions": [
      {
        "date": "2024-12-25",
        "isClosed": true,
        "hours": [{"open": "10:00", "close": "16:00"}]
      }
    ]
  },
  "isTemporarilyClosed": "boolean",
  "plan": "basic|premium",
  "planLimit": "number",
  "staffIds": ["string"],
  "social": {
    "instagram": "string",
    "x": "string",
    "facebook": "string"
  },
  "websiteUrl": "string",
  "statusMeta": {
    "lastComputedIsOpen": "boolean"
  }
}

```

=====

3. UI/UX設計

=====

ユーザーアプリ画面構成:

HomeScreen（ホーム） - 完全画面構成：

画面構成（上から順番）：

1. アプリヘッダー
 - 左上： アプリアイコン & サービス名「GrouMap」
 - 右上： 通知ベルアイコン（未読時は赤い丸表示）
2. メンテナンス告知バナー（条件付き表示）
 - システムメンテナンス時のお知らせ表示
3. レベル・XPカード表示
 - レベル表示（1~100Lv）： 現在のユーザーレベル
 - XPゲージ： 次レベルまでの経験値プログレスバー
 - QRコード： ユーザー識別用
 - 輝度調整ボタン： 画面輝度変更
4. ポイント・スタンプ表示
 - 左： ポイント数表示（更新ボタン付き）
 - 右： ゴールドスタンプ数表示（更新ボタン付き）
5. ルーレットチャレンジ
 - スロットマシンチャレンジへのエントリーボタン
 - ログイン時のみ表示、1日1回制限
 - ゴールドグラデーション背景
6. 4x2メニューグリッド
7. XP詳細表示
8. クーポン横スライド表示
9. 店舗からのお知らせ（投稿横スライド表示）

ユーザーアプリ 4x2メニューグリッド構成：

[ポイント履歴]	[スタンプ]	[バッジ]	[店舗一覧]
[ランキング]	[友達紹介]	[店舗紹介]	[フィードバック]

店舗用アプリ 4x3メニューグリッド構成：

[ポイント発行履歴]	[ポイント利用履歴]	[クーポン管理]	[投稿管理]
[顧客分析]	[売上データ]	[スタッフ管理]	[プラン・契約情報]
[運営からのお知らせ]	[フィードバック送信]	[空き]	[空き]

4. レベル・XPシステム仕様

レベルシステム概要：

レベル範囲：

- 最小レベル： 1（新規ユーザーの初期値）
- 最大レベル： 100（カンスト）
- レベルアップ： XP（経験値）を獲得してレベルを上げる

XP獲得ルール：

1. ポイント獲得時のXP
 - 1ポイント獲得 = 2XP付与
2. ポイント利用時のXP
 - 1ポイント利用 = 1XP付与
3. バッジ獲得時のXP
 - 初回来店バッジ： 50XP
 - 常連バッジ： 200XP

- 探索バッジ: 150XP
- スタンプマスターバッジ: 300XP

4. 特別行動でのXP

- 毎日ログイン: 10XP
- スロットチャレンジ参加: 25XP
- 友達招待成功: 100XP

レベルアップ特典:

レベル到達報酬:

- レベル5: 100ポイント + ブロンズ探索バッジ
- レベル15: 500ポイント + 特別クーポン
- レベル50: 2500ポイント + ベテランバッジ
- レベル100: 10000ポイント + レジェンドバッジ + GrouMapマスター称号

=====

5. 権制限実装仕様

=====

Premium機能制限実装:

UI表示制御:

- クーポン管理: Premium限定
- 投稿管理: Premium限定
- 基本機能でロック表示、タップでアップグレード誘導

機能アクセス制御:

クーポン作成時のプラン制限チェック

```
if (store?.plan != StorePlan.premium) {
    throw PremiumRequiredException('クーポン作成にはPremiumプランが必要です');
}
```

会社管理者権限実装:

UI表示制御:

- 管理者メニュー表示 (isCompanyAdmin: true 時のみ)
- お知らせ作成・管理機能
- フィードバック管理機能

機能アクセス制御:

会社管理者権限確認

```
if (user?.isCompanyAdmin != true) {
    throw InsufficientPermissionException('会社管理者権限が必要です');
}
```

=====

6. 技術仕様

=====

必須パッケージ:

dependencies:

```
# Firebase関連
firebase_core: ^2.24.2
firebase_auth: ^4.15.3
cloud_firestore: ^4.13.6
firebase_storage: ^11.5.6
cloud_functions: ^4.6.6
firebase_messaging: ^14.7.10
```

```
# UI・ナビゲーション
flutter_riverpod: ^2.4.9
go_router: ^12.1.3
google_maps_flutter: ^2.5.0

# 画像処理
image_picker: ^1.0.4
flutter_image_compress: ^2.1.0

# ユーティリティ
intl: ^0.18.1
uuid: ^4.2.2
fl_chart: ^0.66.0
```

Firestore Storage設計:

```
storage/
├── users/{uid}/
│   └── avatars/{fileName}           # ユーザープロフィール画像
├── stores/{storeId}/
│   ├── posts/{postId}/
│   │   └── images/{fileName}       # 投稿画像
│   ├── menus/{menuItemId}/
│   │   └── images/{fileName}       # メニュー画像
│   └── hero/{fileName}             # 店舗ヒーロー画像
└── temp/                           # 一時保存領域
```

7. Firebase Security Rules

Firestore Rules:

```
// プラン・Premium機能チェック
function isPremiumStore(storeId) {
  return get(/databases/(database)/documents/stores/(storeId)).data.plan == 'premium';
}
```

// スタッフ権限チェック

```
function isStoreStaff(storeId, uid) {
  return uid in get(/databases/(database)/documents/stores/(storeId)).data.staffIds;
}
```

// 会社管理者チェック

```
function isCompanyAdmin(uid) {
  return get(/databases/(database)/documents/users/(uid)).data.isCompanyAdmin == true;
}
```

Storage Rules:

```
// ユーザーアバター
match /users/{uid}/avatars/{fileName} {
  allow read, write: if request.auth.uid == uid
    && request.resource.contentType.matches('image/*');
}
```

// 店舗画像

```
match /stores/{storeId}/{allPaths=**} {
  allow read: if true;
  allow write: if isStoreStaff(storeId, request.auth.uid)
    && request.resource.contentType.matches('image/*');
```

```
}
```

```
=====
```

8. Cloud Functions仕様

```
=====
```

主要Functions:

1. grantPoints: ポイント付与処理
2. redeemPoints: ポイント利用処理
3. createOrUpdatePost: 投稿作成・更新
4. redeemCoupon: クーポン利用処理
5. markAnnouncementAsRead: お知らせ既読処理
6. aggregateDailyMetrics: 日次集計処理
7. incrementPostView: 投稿閲覧数カウント

```
=====
```

9. 実装雛形生成仕様

```
=====
```

Dartデータモデル実装:

UserModel:

@freezed

```
class UserModel with _$UserModel {  
  const factory UserModel({  
    required String uid,  
    required String displayName,  
    @Default(1) int level,  
    @Default(0) int currentXP,  
    @Default(0) int levelXP,  
    @Default(100) int nextLevelXP,  
    @Default(false) bool isCompanyAdmin,  
  }) = _UserModel;  
}
```

StoreModel:

@freezed

```
class StoreModel with _$StoreModel {  
  const factory StoreModel({  
    required String storeId,  
    required String name,  
    @Default(StorePlan.basic) StorePlan plan,  
    @Default([]) List<String> staffIds,  
  }) = _StoreModel;  
}
```

```
=====
```

10. テスト戦略

```
=====
```

単体テスト:

- データモデルのシリアライゼーション
- Repository・Serviceロジック
- ユーティリティ関数
- バリデーション処理

ウィジェットテスト:

- UI状態変化の検証
- ユーザーインタラクション

- isCompanyAdmin フラグによるUI切替
- 営業時間の深夜跨ぎ判定

統合テスト:

- 認証フロー (E2E)
- クーポン利用フロー (E2E)
- 画像アップロードフロー
- Cloud Functions連携

=====

11. 実行計画書 (Implementation Roadmap)

=====

開発フェーズ概要:

Phase 1: プロジェクト初期化・基盤構築 (3-4日)

- プロジェクト構成
- Firebase初期設定
- コア機能
- 開発環境

Phase 2: 認証・ユーザー管理システム (4-5日)

- AuthService クラス実装
- ユーザーモデル
- 認証画面
- ルーティング基盤

Phase 3: データモデル・Repository層 (5-6日)

- データモデル (Freezed + json_serializable)
- Repository層 (CRUD)
- サービス層
- 状態管理 (Riverpod)

Phase 4: 基本UI・ナビゲーション (4-5日)

- 共通ウィジェット
- ユーザーアプリ基本画面
- 店舗用アプリ基本画面
- 画面遷移・アプリ種別切り替え

Phase 5: コア機能実装 (7-8日)

- 地図機能
- 店舗機能
- 投稿・クーポン基本機能
- ポイント・スタンプシステム

Phase 6: 高度な機能・管理者機能 (5-6日)

- 会社管理者機能
- 分析ダッシュボード
- 高度なクーポン機能
- プッシュ通知

Phase 7: テスト・最適化・デプロイ準備 (4-5日)

- テスト実装
- パフォーマンス最適化
- Firebase本番環境
- アプリストア準備

=====

実装注意事項

=====

セキュリティ要件:

1. **Firestore Security Rules**: 全てのデータアクセスで適切な権限チェック
2. **認証状態管理**: 未認証ユーザーの適切なハンドリング
3. **入力値検証**: クライアント・サーバー双方での検証
4. **画像アップロード**: ファイルタイプ・サイズ制限の実装
5. **API率制限**: Cloud Functionsでの適切な制限設定

パフォーマンス要件:

1. **画像圧縮**: 1MB以下、品質80%での圧縮必須
2. **遅延読み込み**: 長いリストでのpagination実装
3. **キャッシュ戦略**: `cached_network_image` での適切なキャッシング
4. **Firestore最適化**: 複合インデックス・クエリ最適化
5. **オフライン対応**: Firestore Offline Persistence活用

エラーハンドリング要件:

1. **ネットワークエラー**: ユーザーフレンドリーなメッセージ表示
2. **権限エラー**: 適切な権限要求・説明
3. **バリデーションエラー**: フィールド単位の詳細エラー表示
4. **Cloud Functions**: 適切なHTTPSエラーコード返却
5. **ログ記録**: エラー分析のための適切なログ出力

=====

以上がGrouMapアプリの詳細設計仕様書です。
この仕様に基づいて、段階的に実装を進めてください。