

CoreVia

Business Analysis Document

iOS → Android Migration Blueprint

Tarix: 2026-02-24 | Versiya: 1.0

Məqsəd: iOS (SwiftUI) appinin 1:1 Android (Kotlin + Jetpack Compose) versiyasını yaratmaq

1. ÜMUMİ MƏLUMAT

Parametr	Dəyər
App Adı	CoreVia
Platforma	iOS (mövcud) → Android (yaradılacaq)
Backend	FastAPI (Python) - Hetzner Server
API Base URL	https://api.corevia.life
Direct IP	http://89.167.53.205
Database	PostgreSQL (corevia_db)
iOS Fayl Sayı	111 Swift fayl
Dillər	Azərbaycan <input type="checkbox"/> , İngilis <input type="checkbox"/> , Rus <input type="checkbox"/>
User Types	Client (Tələbə), Trainer (Müəllim)
Package Name	life.corevia.app

2. TEXNOLOGİYA STACK-İ

iOS (Mövcud)

Texnologiya	Məqsəd
SwiftUI	UI Framework
MVVM + Manager	Arxitektura
async/await	Asinxron əməliyyatlar
CoreML (YOLO v8 + EfficientNet)	AI Food Detection
Keychain	Token saxlama
NavigationStack	Naviqasiya
StoreKit	In-App Purchase
URLSession	Networking

Android (Yaradılacaq)

Texnologiya	Məqsəd
Kotlin	Proqramlaşdırma dili
Jetpack Compose	UI Framework
MVVM + Repository	Arxitektura
Kotlin Coroutines + Flow	Asinxron əməliyyatlar
TensorFlow Lite	AI Food Detection
EncryptedSharedPreferences	Token saxlama
Navigation Compose	Naviqasiya
Google Play Billing	In-App Purchase
Retrofit + OkHttp	Networking
Hilt	Dependency Injection
Coil	Şəkil yükləmə

Texnologiya	Məqsəd
CameraX	Kamera
kotlinx.serialization	JSON parsing

3. NAVİQASIYA STRUKTURU

3.1 Client (Tələbə) - 6 Tab

Tab	Ekran	İkon
Tab 1	Home (Ana Səhifə)	<input type="checkbox"/>
Tab 2	Workouts (Məşqlər)	<input type="checkbox"/>
Tab 3	Food (Qidalanma / AI Kalori)	<input type="checkbox"/>
Tab 4	Chat (Mesajlar)	<input type="checkbox"/>
Tab 5	Analytics (Statistika)	<input type="checkbox"/>
Tab 6	Profile (Profil)	<input type="checkbox"/>

3.2 Trainer (Müəllim) - 6 Tab

Tab	Ekran	İkon
Tab 1	Trainer Home (Dashboard)	<input type="checkbox"/>
Tab 2	Training Plans (Məşq Planları)	<input type="checkbox"/>
Tab 3	Meal Plans (Qidalanma Planları)	<input type="checkbox"/>
Tab 4	Chat (Mesajlar)	<input type="checkbox"/>
Tab 5	Trainer Hub (Məhsul/Session)	<input type="checkbox"/>
Tab 6	Profile (Profil)	<input type="checkbox"/>

3.3 Ortaq Ekranlar

Login / Register / Forgot Password / OTP / Onboarding (yalnız Client) / Marketplace / Live Sessions / Social Feed / Daily Survey / Settings / Premium / Trainer Browse (yalnız Client)

4. EKРАНLAR - TAM SİYAHİ (43 ekran)

4.1 Auth Ekranları

Screen 1: LoginScreen ☐ HAZIR

Parametr	Dəyər
iOS Fayl	LoginView.swift
Android Fayl	ui/auth/LoginScreen.kt
Status	<input type="checkbox"/> HAZIR
API	POST /api/v1/auth/login

Elementlər: Dil seçici (AZ/EN/RU), Logo (glow), User type (Client/Trainer), Email, Password, Şifrəni unutdum link, Login düyməsi (gradient), Qeydiyyat linki, OTP verification (6 rəqəm), 60s geri sayım

Screen 2: RegisterScreen

Parametr	Dəyər
iOS Fayl	RegisterView.swift
Android Fayl	ui/auth/RegisterScreen.kt
Status	<input type="checkbox"/> Yazılmalı
API	POST /api/v1/auth/register

Elementlər: User type seçimi, Ad, Email, Şifrə, Şifrə təsdiq, Qeydiyyat düyməsi, Giriş linki

Screen 3: ForgotPasswordScreen

Parametr	Dəyər
iOS Fayl	ForgotPasswordView.swift
Android Fayl	ui/auth/ForgotPasswordScreen.kt
Status	<input type="checkbox"/> Yazılmalı
API	POST forgot-password + reset-password

3 Step: Email daxil et → OTP kodu daxil et → Yeni şifrə təyin et

4.2 Onboarding (yalnız Client)

Screen 4: OnboardingScreen

Parametr	Dəyər
iOS Fayl	OnboardingView.swift
Android Fayl	ui/onboarding/OnboardingScreen.kt
API	GET /api/v1/onboarding/options + POST complete

4 Step: Fitness Goal → Fitness Level → Bədən info (yaş, çəki, boy) → Trainer tipi (optional)

4.3 Home Ekranları

Screen 5: HomeScreen (Client)

Elementlər: Salamlama ('Salam, {ad}!'), daily stats kartları (məşq vaxtı, kalori), survey prompt, progress bar, bugünkü məşqlər preview, AI recommendations, pull-to-refresh

API: GET /api/v1/workouts/ + GET /api/v1/ai/recommendations + GET /api/v1/survey/daily/today

Screen 6: TrainerHomeScreen (Trainer)

Elementlər: Profil şəkli header, 2x2 stats grid (abunəçilər, aktiv tələbələr, gəlir, planlar), tələbə proqresi, quick actions

API: GET /api/v1/trainer/stats

4.4 Workout Ekranları (Client)

Screen 7: WorkoutScreen

Siyahı + filtr (kateqoriya, tarix). Kartlar: başlıq, kateqoriya, müddət, kalori, status. FAB → AddWorkout

API: GET /api/v1/workouts/ + DELETE /api/v1/workouts/{id} + PATCH .../toggle

Screen 8: AddWorkoutScreen

Inputlar: Başlıq, kateqoriya (strength/cardio/flexibility/endurance), müddət, kalori, qeydlər, tarix

API: POST /api/v1/workouts/

4.5 Food / AI Kalori Ekranları

Screen 9: FoodScreen

Qida siyahısı + nutrition xülasəsi. Filtr: breakfast/lunch/dinner/snack. AI kamera düyməsi

API: GET /api/v1/food/ + DELETE /api/v1/food/{id}

Screen 10: AddFoodScreen

Kamera/qaleriya şəkil, AI analiz düyməsi, manuel giriş (ad, kalori, protein, karb, yağ, tip, qeydlər)

API: POST /api/v1/food/ + POST .../image + POST .../analyze

Screen 11: AICalorieResultScreen

Aşkar edilmiş yeməklər, hər biri üçün kalori/makro/porsiya/confidence. Toplam xülasə. Saxla düyməsi

ML Pipeline: YOLO v8 → EfficientNet → USDA DB (Android: TensorFlow Lite)

Screen 12: AICalorieHistoryScreen

Analiz tarixçəsi siyahısı. Pagination. Gündəlik toplanmalar.

4.6 Chat Ekranları

Screen 13: ConversationsScreen

Söhbət siyahısı. Trainer bölməsi. Son mesaj, vaxt, oxunmamış badge. Client: Premium lazım, Trainer: pulsuz

API: GET /api/v1/chat/conversations + GET /api/v1/chat/limit

Screen 14: ChatDetailScreen

Mesaj tarixçəsi (reverse list), göndər/al bubble-lar, input field, göndər düyməsi, oxunma indikatoru

API: GET /api/v1/chat/history/{userId} + POST /api/v1/chat/send

4.7 Trainer Browse (Client)

Screen 15: TrainerBrowseScreen

Axtarış, filtr (ixtisas, rating, qiymət). Trainer kartları: şəkil, ad, ixtisas, təcrübə, rating, qiymət

API: GET /api/v1/users/trainers + POST /api/v1/users/assign-trainer/{id}

4.8 Training Plan Ekranları (Trainer)

Screen 16: TrainingPlanScreen

Plan siyahısı + filtr (tələbə, plan tipi). Silmə/redaktə. FAB → AddTrainingPlan

Screen 17: AddTrainingPlanScreen

Başlıq, tip (weight_loss/weight_gain/strength_training), exercises (əlavə et: ad, set, rep, müddət), tələbə seç, qeydlər

API: GET/POST/PUT/DELETE /api/v1/plans/training

4.9 Meal Plan Ekranları (Trainer)

Screen 18: MealPlanScreen + Screen 19: AddMealPlanScreen

Plan siyahısı. Yeni plan: başlıq, kalori hədəfi, yemək items (ad, kalori, makro, tip), tələbə, qeydlər

API: GET/POST/PUT/DELETE /api/v1/plans/meal

4.10 Marketplace Ekranları

Screen 20: MarketplaceScreen

Məhsul siyahısı. Tip filtri: workout_plan, meal_plan, training_program, ebook, video_course. Pagination

API: GET /api/v1/marketplace/products

Screen 21: ProductDetailScreen

Cover şəkil, başlıq, təsvir, qiymət, satıcı info, rating, reviews, satın al düyməsi

API: GET .../products/{id} + GET .../reviews + POST /api/v1/marketplace/purchase

Screen 22: WriteReviewScreen

Ulduz rating (1-5), şərh text, saxla düyməsi. API: POST /api/v1/marketplace/reviews

4.11 Trainer Hub Ekranları (Trainer)

Screen 23: TrainerHubScreen

Hub dashboard, quick stats, alt bölmələrə naviqasiya, quick actions

Screen 24: TrainerProductsScreen

Trainer-in öz məhsulları siyahısı, tipə görə filtr, yarat/sil/redaktə

Screen 25: CreateProductScreen

Məhsul tipi, başlıq, təsvir, qiymət, cover şəkil upload, published toggle

API: POST /api/v1/marketplace/products

Screen 26: TrainerSessionsScreen

Session siyahısı, status filtri (scheduled/live/completed/cancelled), yarat/sil

Screen 27: CreateLiveSessionScreen

Başlıq, təsvir, session tipi (group/one_on_one/open), max iştirakçı, çətinlik, müddət, tarix/vaxt, public/private, ödənişli/pulsuz, qiymət, exercises

API: POST /api/v1/live-sessions

4.12 Live Session Ekranları

Screen 28: LiveSessionListScreen

Session siyahısı, filtr (çətinlik, tip). Kartlar: başlıq, trainer, qiymət, iştirakçılar, tarix. Pagination

API: GET /api/v1/live-sessions

Screen 29: LiveSessionDetailScreen

Başlıq, təsvir, trainer info, vaxt, çətinlik, max iştirakçı, workout planı, qoşul düyməsi, iştirakçılar

API: GET .../live-sessions/{id} + POST .../join + GET .../participants

Screen 30: LiveWorkoutScreen

Canlı session: geri sayım, cari hərəkət, poza aşkarlama feedback, real-time iştirakçılar, performans

WebSocket: wss://api.corevia.life/api/v1/live-sessions/ws/{id}?token=...

4.13 Social Ekranları

Screen 31: SocialFeedScreen

Post siyahısı (feed). Hər postda: müəllif info, mətn, şəkil, like sayı, şərh sayı, like düyməsi, şərh düyməsi

API: GET /api/v1/social/feed + POST .../like + DELETE .../like

Screen 32: CreatePostScreen

Post tipi (workout/meal/progress/achievement/general), mətn, şəkil, workout/meal bağla, public/private

API: POST /api/v1/social/posts + POST .../image

Screen 33: CommentsScreen

Şərhlərin siyahısı (müəllif, mətn, vaxt), şərh input, göndər düyməsi

API: GET .../comments + POST .../comments

4.14 Daily Survey

Screen 34: DailySurveyScreen

Sliders: enerji(1-5), yuxu saati(0-24h), yuxu keyfiyyəti(1-5), stress(1-5), əzələ ağrısı(1-5), əhval(1-5), su stəkanları(0-30), qeydlər

Suallar backend-dən gəlir (multi-language dəstəyi)

API: GET /api/v1/survey/questions + POST /api/v1/survey/daily + GET .../today

4.15 Analytics Ekranları

Screen 35: AnalyticsDashboardScreen

Cari həftə statistikası, çəki trendi (line chart), məşq trendi, qidalanma trendi, workout streak, 30 günlük toplanmalar

API: GET /api/v1/analytics/dashboard

Screen 36: OverallStatsScreen

Toplam məşq sayı, toplam dəqiqə, toplam yandırılan kalori, orta gündəlik kalori, consistency faizi

4.16 Profil Ekranları

Screen 37: ProfileScreen

User type-a görə fərqli content. Profil şəkli, statistikalar, edit düyməsi, settings, logout

Client: ad, email, yaş, çəki, boy, hədəf. Trainer: ad, ixtisas, təcrübə, bio, qiymət, verification

API: GET /api/v1/auth/me

Screen 38: EditProfileScreen

Client: ad, yaş, çəki, boy, hədəf + şəkil upload. Trainer: ad, ixtisas, təcrübə, bio, qiymət + şəkil upload

API: PUT /api/v1/users/profile + POST /api/v1/uploads/profile-image

4.17 Settings, Premium, Digər

Screen 39: SettingsScreen

Dil seçici (AZ/EN/RU), tema (light/dark), bildiriş ayarları, gizlilik, hesab silmə, çıxış

Screen 40: PremiumScreen

Üstünlüklər: limitsiz chat, ətraflı analitika, prioritet dəstək. Android: Google Play Billing Library

Screen 41: TrainerVerificationScreen

Verification foto upload, Instagram handle, status göstəricisi. API: POST /api/v1/auth/verify-trainer

Screen 42: RouteTrackingScreen

Google Maps, start/stop, məsafə, müddət, kalori. API: POST/GET/DELETE /api/v1/routes/

Screen 43: TrainerContentScreen

Content siyahısı, yarat/sil. API: GET /api/v1/content/my + POST /api/v1/content/

5. API ENDPOINTLƏRİ - TAM SİYAHİ (92)

5.1 Authentication (8)

#	Method	Endpoint	Auth
1	POST	/api/v1/auth/login	Xeyr
2	POST	/api/v1/auth/register	Xeyr
3	GET	/api/v1/auth/me	Bəli
4	POST	/api/v1/auth/refresh	Xeyr
5	POST	/api/v1/auth/refresh-claims	Bəli
6	DELETE	/api/v1/auth/delete-account	Bəli
7	POST	/api/v1/auth/forgot-password	Xeyr
8	POST	/api/v1/auth/reset-password	Xeyr

5.2 User/Profile (6)

#	Method	Endpoint	Auth
9	PUT	/api/v1/users/profile	Bəli
10	POST	/api/v1/uploads/profile-image	Bəli
11	GET	/api/v1/users/trainers	Bəli
12	POST	/api/v1/users/assign-trainer/{id}	Bəli
13	DELETE	/api/v1/users/unassign-trainer	Bəli
14	GET	/api/v1/users/my-students	Bəli

5.3 Workouts (5)

#	Method	Endpoint	Auth
15	POST	/api/v1/workouts/	Bəli
16	GET	/api/v1/workouts/	Bəli
17	PUT	/api/v1/workouts/{id}	Bəli
18	DELETE	/api/v1/workouts/{id}	Bəli
19	PATCH	/api/v1/workouts/{id}/toggle	Bəli

5.4 Food (6)

#	Method	Endpoint	Auth
20	POST	/api/v1/food/	Bəli
21	GET	/api/v1/food/	Bəli
22	PUT	/api/v1/food/{id}	Bəli
23	DELETE	/api/v1/food/{id}	Bəli
24	POST	/api/v1/food/{id}/image	Bəli
25	POST	/api/v1/food/analyze	Bəli

5.5 Training Plans (5)

#	Method	Endpoint	Auth
26	POST	/api/v1/plans/training	Beli
27	GET	/api/v1/plans/training	Beli
28	PUT	/api/v1/plans/training/{id}	Beli
29	DELETE	/api/v1/plans/training/{id}	Beli
30	PUT	/api/v1/plans/training/{id}/complete	Beli

5.6 Meal Plans (5)

#	Method	Endpoint	Auth
31	POST	/api/v1/plans/meal	Beli
32	GET	/api/v1/plans/meal	Beli
33	PUT	/api/v1/plans/meal/{id}	Beli
34	DELETE	/api/v1/plans/meal/{id}	Beli
35	PUT	/api/v1/plans/meal/{id}/complete	Beli

5.7 Chat (4)

#	Method	Endpoint	Auth
36	GET	/api/v1/chat/conversations	Beli
37	GET	/api/v1/chat/history/{userId}	Beli
38	POST	/api/v1/chat/send	Beli
39	GET	/api/v1/chat/limit	Beli

5.8 Social (8)

#	Method	Endpoint	Auth
40	GET	/api/v1/social/feed	Beli
41	POST	/api/v1/social/posts	Beli
42	POST	/api/v1/social/posts/{id}/image	Beli
43	POST	/api/v1/social/posts/{id}/like	Beli
44	DELETE	/api/v1/social/posts/{id}/like	Beli
45	DELETE	/api/v1/social/posts/{id}	Beli
46	GET	/api/v1/social/posts/{id}/comments	Beli
47	POST	/api/v1/social/posts/{id}/comments	Beli

5.9 Live Sessions (7+WS)

#	Method	Endpoint	Auth
48	GET	/api/v1/live-sessions	Beli
49	POST	/api/v1/live-sessions	Beli
50	GET	/api/v1/live-sessions/{id}	Beli
51	POST	/api/v1/live-sessions/{id}/join	Beli
52	GET	/api/v1/live-sessions/{id}/participants	Beli
53	DELETE	/api/v1/live-sessions/{id}	Beli

#	Method	Endpoint	Auth
54	WS	/api/v1/live-sessions/ws/{id}?token=	Bəli

5.10 Marketplace (8)

#	Method	Endpoint	Auth
55	GET	/api/v1/marketplace/products	Bəli
56	GET	/api/v1/marketplace/products/{id}	Bəli
57	POST	/api/v1/marketplace/products	Bəli
58	DELETE	/api/v1/marketplace/products/{id}	Bəli
59	GET	/api/v1/marketplace/.../reviews	Bəli
60	POST	/api/v1/marketplace/reviews	Bəli
61	POST	/api/v1/marketplace/purchase	Bəli
62	GET	/api/v1/marketplace/my-purchases	Bəli

5.11 Digər Endpointlər (30)

#	Method	Endpoint	Auth
63-66	CRUD	/api/v1/trainer/{id}/reviews	Bəli
67-71	CRUD	/api/v1/content/	Bəli
72-75	G/P	/api/v1/survey/...	Bəli
76-79	CRUD	/api/v1/routes/	Bəli
80	GET	/api/v1/analytics/dashboard	Bəli
81	GET	/api/v1/ai/recommendations	Bəli
82-84	G/P	/api/v1/onboarding/...	Qarışıq
85-86	POST	/api/v1/premium/...	Bəli
87	GET	/api/v1/trainer/stats	Bəli
88	POST	/api/v1/auth/verify-trainer	Bəli
89-91	G/P	/news/...	Bəli

6. ƏSAS DATA MODƏLLƏRİ (Kotlin)

AuthModels

```
data class LoginRequest(val email: String, val password: String)

data class RegisterRequest(
    val name: String, val email: String,
    val password: String, val userType: String
)

data class AuthResponse(
    val accessToken: String, val refreshToken: String,
    val tokenType: String
)

data class UserResponse(
    val id: String, val name: String, val email: String,
    val userType: String, val profileImageUrl: String?,
    val isActive: Boolean, val isPremium: Boolean,
    val age: Int?, val weight: Double?, val height: Double?,
    val goal: String?, val trainerId: String?,
    val specialization: String?, val experience: Int?,
    val rating: Double?, val pricePerSession: Double?,
    val bio: String?, val verificationStatus: String?
)
```

WorkoutModels

```
data class Workout(
    val id: String, val title: String,
    val category: String, // strength, cardio, flexibility, endurance
    val duration: Int, val caloriesBurned: Int?,
    val notes: String?, val date: String,
    val isCompleted: Boolean
)
```

FoodModels

```
data class FoodEntry(
    val id: String, val name: String, val calories: Int,
    val protein: Double?, val carbs: Double?, val fats: Double?,
    val mealType: String, val date: String, val imageUrl: String?
)

data class AICalorieResult(
    val foods: List<DetectedFood>, val totalCalories: Double,
    val totalProtein: Double, val totalCarbs: Double,
    val totalFat: Double, val confidence: Double
)
```

ChatModels

```
data class ChatConversation(
    val userId: String, val userName: String,
    val userProfileImage: String?, val lastMessage: String?,
    val unreadCount: Int
)

data class ChatMessageResponse(
    val id: String, val senderId: String,
    val receiverId: String, val message: String,
    val isRead: Boolean, val createdAt: String
)
```

MarketplaceModels

```
data class MarketplaceProduct(
    val id: String, val sellerId: String,
    val productType: String, val title: String,
    val description: String, val price: Double,
    val currency: String, val coverImageUrl: String?,
    val averageRating: Double?, val reviewCount: Int?
)
```

LiveSessionModels

```
data class LiveSession(  
    val id: String, val trainerId: String,  
    val title: String, val sessionType: String,  
    val maxParticipants: Int, val difficultyLevel: String,  
    val durationMinutes: Int, val scheduledStart: String,  
    val status: String, val isPaid: Boolean,  
    val price: Double, val registeredCount: Int?  
)
```

SocialModels

```
data class SocialPost(  
    val id: String, val userId: String,  
    val postType: String, val content: String?,  
    val imageUrl: String?, val likesCount: Int,  
    val commentsCount: Int, val isPublic: Boolean,  
    val isLiked: Boolean, val author: PostAuthor?  
)
```

SurveyModels

```
data class DailySurveyRequest(  
    val energyLevel: Int, val sleepHours: Double,  
    val sleepQuality: Int, val stressLevel: Int,  
    val muscleSoreness: Int, val mood: Int,  
    val waterGlasses: Int, val notes: String?  
)
```

7. ANDROID PAKET STRUKTURU

```
life.corevia.app/  
+-- CoreViaApp.kt, MainActivity.kt  
+-- data/  
| +-- remote/ (ApiService, AuthInterceptor, TokenRefreshAuthenticator)  
| +-- local/ (TokenManager, PreferencesManager)  
| +-- repository/ (Auth, Workout, Food, Plan, Chat, Social, ...)  
| +-- model/ (Auth, User, Workout, Food, Chat, Marketplace, ...)  
+-- di/ (AppModule, NetworkModule, RepositoryModule)  
+-- ui/  
| +-- navigation/AppNavigation.kt  
| +-- theme/ (Color, Type, Theme)  
| +-- components/ (Button, TextField, Card, Loading, ...)  
| +-- auth/ (Login, Register, ForgotPassword + ViewModels)  
| +-- onboarding/ (Onboarding + ViewModel)  
| +-- home/ (Home, TrainerHome + ViewModels)  
| +-- workout/ (Workout, AddWorkout + ViewModels)  
| +-- food/ (Food, AddFood, AICalorie + ViewModels)  
| +-- chat/ (Conversations, ChatDetail + ViewModels)  
| +-- trainers/ (TrainerBrowse + ViewModel)  
| +-- plans/ (TrainingPlan, MealPlan + ViewModels)  
| +-- marketplace/ (Marketplace, ProductDetail, Review + VMs)  
| +-- trainerhub/ (Hub, Products, Sessions + ViewModels)  
| +-- livesession/ (List, Detail, LiveWorkout + ViewModel)  
| +-- social/ (Feed, CreatePost, Comments + ViewModels)  
| +-- survey/ (DailySurvey + ViewModel)  
| +-- analytics/ (Dashboard, Stats + ViewModel)  
| +-- profile/ (Profile, EditProfile + ViewModel)  
| +-- settings/ (Settings + ViewModel)  
| +-- premium/ (Premium + ViewModel)  
| +-- route/ (RouteTracking + ViewModel)  
| +-- content/ (TrainerContent + ViewModel)  
+-- util/ (Constants, Extensions, DateUtils, NetworkResult, Localization)
```

8. İNKİŞAF PLANI

Phase	Tapşırıqlar	Prioritet
Phase 1: İnfrastruktur	NetworkModule, TokenManager, ApiService, NetworkResult, Navigation	Yüksək
Phase 2: Auth	Login (hazır), Register, ForgotPassword, AuthRepository	Yüksək
Phase 3: Core	Onboarding, Home (Client/Trainer), TabBar, Profile	Yüksək
Phase 4: Workout/Food	Workout CRUD, Food CRUD, AI Calorie (TF Lite)	Orta
Phase 5: Plans/Chat	TrainingPlan, MealPlan, Conversations, ChatDetail	Orta
Phase 6: Market/Social	Marketplace, Social Feed, Trainer Browse	Orta
Phase 7: Advanced	Live Sessions (WS), Trainer Hub, Analytics, Survey, Routes, Premium	Aşağı

9. XÜLASƏ

Kategoriya	Say
Toplam Ekranlar	43
Toplam API Endpointlər	92 (91 + 1 WebSocket)
Data Modəlləri	30+
Repository-lər	16
ViewModel-lər	25+
Dillər	3 (AZ, EN, RU)

Kategoriya	Say
User Types	2 (Client, Trainer)
iOS Fayl Sayı	111
Android Təxmini Fayl Sayı	~120-130

Bu sənəd CoreVia iOS appinin tam analizinə əsaslanır və Android versiyasının 1:1 yaradılması üçün blueprint kimi istifadə olunacaq.