**Technical Overview – Professional Live Streaming Platform with Monetization for Virtual Consultations**

**IMPORTANT**  
The goal is to ensure complete understanding of the architecture, business logic, payment flows, and implementation responsibilities.

**PROJECT OVERVIEW**

This is a complete and automated digital platform, developed for healthcare professionals (psychologists, nutritionists, physiotherapists, doctors, and other independent specialists) who want to:

• Conduct private online consultations with predefined durations;  
• Broadcast educational live content publicly to attract new clients;  
• Sell premium digital content (photos and videos);  
• Boost profile visibility through paid promotions;  
• Monetize their digital presence securely and at scale.

Live video streaming is powered by **NGINX-RTMP**, enabling real-time broadcasts using tools like **OBS Studio** or any RTMP-compatible software. The system supports simultaneous viewers with high stability and low latency.

User engagement and monetization are integrated through a flexible system that includes:

• Token-based transactions for services;  
• Subscription plans with recurring billing;  
• Escrow payments via **Stripe Connect**;  
• Split commissions via **Stripe** or **Mercado Pago**;  
• Credit logic to handle alternative payment flows securely.

The user experience is inspired by platforms link https://www.twitch.tv/),, offering real-time interaction, live streaming, and service proposal mechanisms.

**FUNCTIONAL MODULES AND USER FLOW**

**1. Public Streaming Module**

• Professionals host live educational broadcasts to engage their audience;  
• Users watch for free and can send private consultation proposals during the stream;  
• When offline, a short waiting video is shown (max. 15 seconds);  
• Admin moderation is supported by **AWS Rekognition**.

**2. Timed Private Consultation Module**

• Users send time-based proposals (e.g., R$30 for 5 minutes);  
• Professionals must accept or reject within 24 hours;  
• Accepted proposals trigger value hold via **Stripe escrow**;  
• Upon session completion, funds are split between the platform and the professional;  
• If the patient is absent, the professional receives the amount as a no-show guarantee, and the platform collects a “consultation room rental” fee, which is debited from the user’s balance;  
• Users must have enough balance to cover room rental before sending a proposal. If not, they’re prompted to buy tokens;  
• Sessions take place in private rooms with countdown timers (3, 5, or 7 minutes);  
• **Fallback via Mercado Pago**: without escrow, users need preloaded and validated credit balance.

**3. Premium Gallery Module**

• Professionals upload individual media (photos/videos);  
• Each item has its own pricing;  
• Access is granted upon payment (via Mercado Pago);  
• Gallery content must be updated every 30 days to remain visible.

**4. Subscription System**

• **Level 1 (Free):** Limited to public content only;  
• **Level 2 (Premium):** Access to proposals, private rooms, and media gallery;  
• **Level 3 (Boosted):** Includes profile highlighting and search priority;  
• Recurring billing via **Mercado Pago** with upgrade/downgrade options.

**5. Profile Boosting (Highlighting)**

• Professionals can pay to have their profiles featured on the homepage and search results;  
• Visibility lasts for 24 hours or according to the selected plan.

**6. Token & Credit System**

• Patients purchase tokens to use in private sessions;  
• Professionals buy credits to boost visibility or unlock extra features;  
• Internal wallet tracks balances, usage, and pending transfers.

**ADMINISTRATIVE AND MODERATION TOOLS**

• Manual user verification and approval (patients and professionals);  
• AWS Rekognition integration for automatic flagging of stream violations;  
• Creation and management of sub-admins with limited permissions;  
• Launch of segmented email campaigns for marketing or communication;  
• Full admin access to logs, disputes, and financial transactions;  
• Force session termination, account suspension, or content deletion.

**PAYMENT INFRASTRUCTURE & LOGIC**

**Stripe Connect (Primary Method)**

• All private session payments are processed via **Stripe**;  
• Funds are securely held in **escrow**;  
• After the session, funds are automatically split;  
• Webhooks control payout confirmation and notifications.

**Mercado Pago (Alternative Method)**

• Used when Stripe is unavailable or unsupported in the user’s region;  
• Users preload credits, which are blocked upon proposal submission;  
• Once the service is delivered, credits are released to the professional’s wallet;  
• If payment fails, the admin can review and block the user through the reporting flow.

The fallback flow simulates internal escrow to ensure security and automation.

**Additional Logic:**  
• Configurable commission percentage per transaction;  
• Admin panel controls to activate/deactivate gateways as needed;  
• Full ledger of blocked, released, and pending balances per user;  
• Daily cron jobs can be configured for transfers (in fallback via Mercado Pago).

**ADMIN PANEL FEATURES**

• Exportable financial reports by user, period, or transaction type;  
• Commission management by payment method or profile type;  
• Manual actions for disputes and withdrawals;  
• Visual editor for homepage banners, logos, and layout;  
• Control over profile highlighting and gallery media expiration.

**TASKS**

• Restore front-end based on validated snapshot;  
• Review and finalize Stripe escrow and split logic;  
• Simulate internal escrow with Mercado Pago fallback;  
• Implement token reservation system for room rental and profile boosting;  
• Integrate push notification system (Firebase for mobile, SNS for web);  
• Finalize visual admin panel for banners and logos;  
• Adjust front-end layout and responsiveness;  
• Apply automatic media expiration rule (30 days);  
• Sync admin panel fields with front-end;  
• Test profile highlight functionality;  
• Perform full UX audit and performance optimization.

**MOBILE MODULE / PWA**

• App built with **Flutter WebView** for Android/iOS;  
• Includes splash screen, custom icon, and offline fallback;  
• Supports push notifications and geolocation.

**🔧 TECHNOLOGIES USED**

• **Backend:** PHP (Laravel)  
• **Frontend:** HTML, CSS, JS  
• **Admin Panel:** Laravel Breeze + Spatie ACL  
• **Database:** MySQL  
• **Authentication:** Laravel Sanctum  
• **Storage:** Amazon S3  
• **Streaming & Moderation:** NGINX RTMP + AWS Rekognition  
• **Payments:** Stripe Connect (Escrow/Split) + Mercado Pago (Fallback)  
• **Deployment:** AWS EC2 + GitHub + SSL

**FINAL DELIVERABLES**

• 100% functional and stable source code;  
• Technical documentation and installation instructions;  
• User and admin usage guides;  
• Narrated video showcasing platform and admin panel usage;  
• Web publishing + delivery of mobile binaries (via PWA).