Chào bạn, tôi là "Call .Net".

Tôi hiểu sự sốt ruột và mong muốn "thực chiến" ngay của bạn. Bạn không cần lý thuyết suông nữa, bạn cần **"Cầm tay chỉ việc"**.

Dưới đây là bản kế hoạch triển khai **Level Technical (Kỹ thuật chi tiết)** dành cho **1 Server FusionPBX** để bạn (Backend Lead) và team bắt tay vào làm ngay chiều nay.

### PHẦN 1: SETUP SERVER & MÔI TRƯỜNG (Làm ngay hôm nay)

**Mục tiêu:** Đến cuối ngày, bạn phải có trang Login của FusionPBX và API sẵn sàng.

#### 1. Mua VPS/Cloud

* **Cấu hình:** 4 Core CPU, 4GB RAM, 50GB SSD (Debian 12 Bookworm).
* **Firewall (Mở port ngay lập tức):**
  + TCP: 80, 443 (Web/API), 22 (SSH).
  + TCP/UDP: 5060, 5061 (SIP Signaling).
  + UDP: 16384 - 32768 (RTP - Quan trọng nhất để có tiếng).
  + TCP: 8021 (ESL - Để .NET kết nối vào điều khiển. **Lưu ý:** Chỉ whitelist IP của server .NET, không mở public).

#### 2. Cài FusionPBX (Dùng Script chuẩn)

SSH vào server Debian 12 (quyền root) và chạy đúng 2 dòng lệnh này:

Bash

# 1. Tải script cài đặt chính chủ  
wget -O - https://raw.githubusercontent.com/fusionpbx/fusionpbx-install.sh/master/debian/resources/finish.sh | sh  
  
# 2. Ngồi chờ khoảng 15-20 phút.  
# Sau khi xong, nó sẽ hiện ra USERNAME / PASSWORD admin trên màn hình. HÃY COPY LẠI NGAY.

#### 3. Bật Module API (Quan trọng cho .NET)

Mặc định FusionPBX có thể chưa bật App API.

1. Đăng nhập Web Admin (https://IP-Server).
2. Vào menu **Advanced** -> **Menu Manager**.
3. Tìm nhóm **Advanced**, edit và đảm bảo menu **API** được hiển thị.
4. Vào **Advanced** -> **API** để lấy **API Key** (Nếu chưa có thì tạo mới). Code .NET sẽ dùng Key này.

### PHẦN 2: THIẾT KẾ DATABASE (.NET SIDE)

Bạn copy đoạn SQL này chạy vào SQL Server của dự án .NET. Đây là cấu trúc Mapping cần thiết.

SQL

-- 1. Bảng quản lý Khách hàng (Tenants)  
CREATE TABLE Tenants (  
 Id INT IDENTITY(1,1) PRIMARY KEY,  
 Name NVARCHAR(100) NOT NULL,  
 DomainName NVARCHAR(100) NOT NULL, -- Ví dụ: tenant1.call.nextx.vn  
 Status INT DEFAULT 1, -- 1: Active, 0: Locked  
   
 -- QUAN TRỌNG: Mapping với FusionPBX  
 FusionDomainUuid UNIQUEIDENTIFIER NOT NULL, -- UUID của domain bên Fusion  
 ApiKey NVARCHAR(255) NULL, -- Key riêng của tenant (nếu cần)  
 CreatedAt DATETIME DEFAULT GETDATE()  
);  
  
-- 2. Bảng quản lý Máy lẻ (Extensions)  
CREATE TABLE Extensions (  
 Id INT IDENTITY(1,1) PRIMARY KEY,  
 TenantId INT NOT NULL,  
 ExtensionNumber VARCHAR(20) NOT NULL, -- 1001, 1002...  
 Password VARCHAR(50) NOT NULL, -- SIP Password  
 DisplayName NVARCHAR(100),  
   
 -- QUAN TRỌNG: Mapping với FusionPBX  
 FusionExtensionUuid UNIQUEIDENTIFIER NULL,   
   
 FOREIGN KEY (TenantId) REFERENCES Tenants(Id)  
);  
  
-- 3. Bảng Lịch sử cuộc gọi (CDR - Call Detail Records)  
-- Dữ liệu này sẽ được đồng bộ từ FusionPBX về  
CREATE TABLE CallLogs (  
 Id BIGINT IDENTITY(1,1) PRIMARY KEY,  
 FusionCallUuid UNIQUEIDENTIFIER NOT NULL, -- ID duy nhất của cuộc gọi  
 TenantId INT,  
 CallerNumber VARCHAR(50), -- Người gọi  
 DestinationNumber VARCHAR(50), -- Người nghe  
 Direction VARCHAR(20), -- inbound/outbound/local  
 StartTime DATETIME,  
 AnswerTime DATETIME,  
 EndTime DATETIME,  
 Duration INT, -- Tổng thời gian (giây)  
 BillSec INT, -- Thời gian đàm thoại thực tế (để tính tiền)  
 RecordingUrl NVARCHAR(500), -- Link file ghi âm  
 HangupCause VARCHAR(50) -- Lý do tắt máy (NORMAL\_CLEARING, BUSY...)  
);

### PHẦN 3: TÍCH HỢP API (.NET -> FUSIONPBX)

FusionPBX REST API thường nằm ở đường dẫn: https://<domain>/app/api/<resource>

Auth: Header Authorization: Bearer <API\_KEY>

#### 1. Tạo Tenant (Domain)

* **Logic .NET:** Khi tạo khách hàng mới -> Gọi API tạo Domain trên Fusion.
* **Endpoint:** POST /app/api/domains
* **Payload:**  
  JSON  
  {  
   "domain\_name": "khachA.call.nextx.vn",  
   "domain\_enabled": "true"  
  }
* **Response:** Nhận về domain\_uuid. Lưu ngay vào DB Tenants.FusionDomainUuid.

#### 2. Tạo Extension

* **Logic .NET:** User tạo máy lẻ 1001 -> Gọi API tạo Extension trên Fusion.
* **Endpoint:** POST /app/api/extensions
* **Payload:**  
  JSON  
  {  
   "domain\_uuid": "<UUID\_CUA\_KHACH\_A>",  
   "extension": "1001",  
   "password": "PasswordSipSieuKho123!",  
   "accountcode": "khachA",  
   "effective\_caller\_id\_name": "Nhan Vien A",  
   "effective\_caller\_id\_number": "1001",  
   "enabled": "true"  
  }
* **Response:** Nhận về extension\_uuid. Lưu vào DB.

### PHẦN 4: REAL-TIME & CALL CONTROL (ESL)

Đây là phần "ăn tiền" để bạn làm tính năng giám sát, popup cuộc gọi.

* **Thư viện .NET:** Cài gói NuGet NEventSocket (hoặc Freeswitch.Esl).

**Code mẫu C# (Service chạy ngầm - BackgroundService):**

C#

using NEventSocket;  
using NEventSocket.Channels;  
  
public class FreeswitchMonitorService : BackgroundService  
{  
 protected override async Task ExecuteAsync(CancellationToken stoppingToken)  
 {  
 // Kết nối vào cổng ESL của FusionPBX (Port 8021)  
 // Password mặc định thường là "ClueCon" (hãy đổi nó trong file autoload\_configs/event\_socket.conf.xml)  
 using var socket = await InboundSocket.Connect("192.168.1.10", 8021, "ClueCon");  
  
 // Đăng ký nhận sự kiện: Cuộc gọi đến (CHANNEL\_CREATE) và Kết thúc (CHANNEL\_HANGUP\_COMPLETE)  
 await socket.SubscribeEvents(EventName.ChannelCreate, EventName.ChannelHangupComplete);  
  
 Console.WriteLine("Connected to FreeSWITCH ESL!");  
  
 // Vòng lặp lắng nghe  
 while (!stoppingToken.IsCancellationRequested)  
 {  
 var eventMessage = await socket.ReceiveEvent();  
   
 if (eventMessage.EventName == EventName.ChannelCreate)  
 {  
 var caller = eventMessage.GetHeader("Caller-Caller-ID-Number");  
 var dest = eventMessage.GetHeader("Caller-Destination-Number");  
 var uuid = eventMessage.GetHeader("Unique-ID");  
   
 // TODO: Bắn SignalR lên Web để hiện Popup "Có cuộc gọi đến"  
 Console.WriteLine($"Cuộc gọi mới: {caller} -> {dest} (UUID: {uuid})");  
 }  
 }  
 }  
}

### PHẦN 5: ROADMAP CHI TIẾT (LỊCH TRÌNH 2 TUẦN ĐẦU)

Đây là checklist "Must-Do" để bạn chạy được luồng cơ bản (MVP).

#### **TUẦN 1: KẾT NỐI & KHỞI TẠO**

* **Thứ 2 (Hôm nay):**
  + [ ] Mua Server, Cài Debian 12 + FusionPBX.
  + [ ] Login thành công Web Admin FusionPBX.
  + [ ] Tạo DB SQL Server theo cấu trúc ở Phần 2.
  + [ ] Dựng khung Project .NET (Clean Arch).
* **Thứ 3: API Wrapper**
  + [ ] Viết FusionCallService.cs trong .NET.
  + [ ] Implement hàm CreateDomainAsync (Test bằng Postman trước).
  + [ ] Implement hàm CreateExtensionAsync.
  + [ ] Test: Chạy code .NET -> Kiểm tra trên Web Fusion xem domain/ext có hiện ra không.
* **Thứ 4: SIP & Trunking**
  + [ ] Mua 1 đầu số SIP Trunk (VNPT/FPT/CMC) loại test.
  + [ ] Cấu hình Trunk trên giao diện FusionPBX (Phần này nên làm tay trên GUI Fusion trước để hiểu tham số).
  + [ ] Cấu hình Outbound Route (Gọi ra) trên FusionPBX.
  + [ ] Dùng Zoiper (Softphone) đăng nhập Extension 1001 -> Gọi thử ra số di động của bạn. **Mục tiêu: Có chuông và có tiếng.**
* **Thứ 5: Call Control (ESL)**
  + [ ] Viết BackgroundService kết nối ESL port 8021.
  + [ ] Gọi điện thử -> Xem Log Console của .NET có nhảy thông tin không.
  + [ ] Tích hợp SignalR: Khi có cuộc gọi -> Bắn notify ra frontend.
* **Thứ 6 + 7: Frontend & Review**
  + [ ] Làm trang danh sách Extension (CRUD).
  + [ ] Làm nút "Click-to-Call" trên Web (Click -> .NET gửi lệnh ESL originate -> Phone đổ chuông).

#### **TUẦN 2: HOÀN THIỆN & BILLING**

* **Thứ 2:** Xử lý Inbound Route (Gọi vào đổ chuông Extension).
* **Thứ 3:** Viết Job đồng bộ CDR (Call Logs) từ PostgreSQL của Fusion về SQL Server để xem lịch sử.
* **Thứ 4:** Tính năng ghi âm (Tìm hiểu cách lấy link file .wav từ FusionPBX trả về API).
* **Thứ 5:** Security Check (Fail2Ban, đổi port mặc định).
* **Thứ 6:** Demo nội bộ toàn công ty dùng thử.

### PHẦN 6: SRS & SDS (DẠNG NGẮN GỌN CHO DEV)

#### 1. SRS (Yêu cầu phần mềm - Functional)

* **FR-01 (Provisioning):** Hệ thống phải tự động tạo Tenant và Extension trên PBX khi User thao tác trên Web CRM. Độ trễ < 2s.
* **FR-02 (Authentication):** Extension được tạo ra phải có password ngẫu nhiên mạnh (12 ký tự, có ký tự đặc biệt) để tránh bị hack cước.
* **FR-03 (Real-time):** Khi có cuộc gọi đến, Web CRM phải hiện popup trong vòng 500ms chứa thông tin người gọi.
* **FR-04 (History):** Log cuộc gọi phải được đồng bộ về hệ thống chính chậm nhất 5 phút sau khi cuộc gọi kết thúc.

#### 2. SDS (Thiết kế hệ thống - Technical)

* **Mô hình:** Microservices-ready.
* **Giao thức:**
  + Data Config: REST API (HTTPS).
  + Call Event: Socket TCP (ESL protocol).
* **Tech Stack:**
  + App: .NET 8.
  + PBX: FusionPBX 5.x (Debian 12).
  + Proxy: Nginx (Reverse Proxy cho API).
  + DB: SQL Server (Main), PostgreSQL (PBX Internal).

### Lời cuối

Bạn đã có đủ "gạch đá, xi măng" (Code SQL, C#, Script cài đặt). Việc bây giờ là mở Terminal lên và chạy lệnh cài FusionPBX ngay lập tức. Đừng chờ đợi gì nữa! Chúc dự án thành công!