

PRODUCT REQUIREMENT DOCUMENT (PRD)

1 Product Overview

Product Name (Working):

Meetra AI (*naam change ho saka hai*)

Category:

Video Conferencing + Agentic AI Collaboration Platform

Vision:

Zoom jaisa reliable video calling app, lekin:

- Better UI/UX
- Lightweight
- AI-assisted meetings
- Future-proof (AI-resistant nahi, AI-native)

Target Users:

- Remote teams
- Freelancers
- Online teachers / coaches
- Sales & support teams
- Hackathons / startups

2 Core Problems (Why This Product?)

Zoom Problems	Meetra AI Solution
Boring UI	Modern, clean UI
No smart AI help	Built-in Agentic AI
Manual meeting notes	Auto summaries
No action tracking	AI task extraction
Heavy client	Lightweight web-first

3 Core Features (MVP)

Video Conferencing

- 1:1 calls
- Group calls (up to 50 in MVP)
- HD video & audio
- Screen sharing
- Mute / unmute
- Camera on/off

Chat

- In-meeting chat
- Emoji reactions
- File sharing (basic)

Authentication

- Email + password
 - Google login
 - Guest join (link based)
-

4 UI / UX Features (Major Differentiator)

UI Principles

- Minimal
- Calm colors
- Zero clutter
- 1-click actions

UX Improvements

- Floating controls
- Auto-hide toolbar
- Speaker highlight
- Grid + focus modes

Branding

- Custom logo
 - Soft gradient theme
 - Dark / light mode
-

5 🔥 Agentic AI Feature (CORE DIFFERENTIATOR)

AI Meeting Copilot (Agentic)

What it does:

- Listens to meeting (audio → text)
- Understands discussion
- Acts independently (agent)

AI Capabilities (MVP)

-  Live meeting summary
-  Action item extraction
-  Key decisions list
-  “Ask AI about meeting” chat

Example:

“AI, what were the action items for Donia?”

AI replies:

- Donia: Create PRD document
- Ali: Setup backend repo
- Sarah: Design logo

6 Future AI Enhancements (Post-MVP)

- Auto follow-up emails

- Speaker sentiment analysis
 - Meeting effectiveness score
 - Voice commands
 - CRM sync agent
 - Calendar scheduling agent
-

7 Tech Stack (Recommended & Scalable)

Frontend

- Next.js 14 (App Router)
- Tailwind CSS
- Shadcn UI
- WebRTC (video)
- Zustand / Redux

Backend

- Node.js (NestJS / Express)
- WebSockets
- REST + WebRTC signaling

AI Layer

- Gemini / OpenAI / DeepSeek
- Whisper / AssemblyAI (speech-to-text)

- Agent framework (LangGraph / custom)

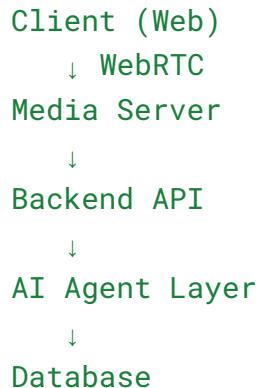
Database

- PostgreSQL (Supabase / Neon)
- Redis (real-time states)

Infra

- Docker
 - AWS / GCP / Fly.io
 - TURN/STUN servers
-

8 System Architecture (High Level)



9 Functional Requirements

Must Have (MVP)

- Create meeting

- Join via link
- Video/audio stable
- AI summary after meeting
- Secure auth

Should Have

- Screen share
- Chat
- Recording

Nice to Have

- Breakout rooms
 - Polls
 - Whiteboard
-

10 Non-Functional Requirements

- Latency < 300ms
 - 99.9% uptime
 - GDPR compliant
 - End-to-end encryption (future)
 - Scalable to 10k users
-

11 Dos & Don'ts (Very Important)

DOs

- Modular code
- Feature flags
- AI opt-in only
- Clear logs
- Rate limits

DON'Ts

- Don't auto-record without consent
 - Don't overload UI
 - Don't hardcode AI keys
 - Don't couple AI with core video logic
-

12 Security Requirements

- JWT authentication
 - Encrypted signaling
 - Secure WebRTC channels
 - Role-based access
 - Meeting passwords
-

Development Phases

Phase 1 – Planning

- Finalize PRD
- Wireframes
- Architecture

Phase 2 – Core Video

- WebRTC
- Calls
- Chat

Phase 3 – AI Agent

- Speech to text
- Summary agent
- Action extractor

Phase 4 – UI Polish

- Animations
- Branding
- Performance

Phase 5 – Testing & Launch

- Load testing
- Security testing

- Beta release
-

14 Success Metrics (KPIs)

- Avg meeting duration
 - AI usage rate
 - Retention
 - Churn
 - Feedback score
-

15 Documentation Files Structure

```
/docs
├── PRD.md
├── ARCHITECTURE.md
├── AI_AGENT_SPEC.md
├── SECURITY.md
├── API_SPEC.md
└── UI_GUIDELINES.md
```

16 Monetization (Optional)

- Free tier (40 min)
- Pro (AI features)
- Teams
- Enterprise

