



# Cerințe Proiect - Chat AI Legislativ pentru Primării








## Obiectivul Proiectului

Dezvoltarea unui **chat widget AI legislativ** pentru primării și instituții publice din România.

**Faza inițială:** Focus pe **taxe și impozite** - răspunsuri la întrebări despre legislația fiscală locală.

**Extensii viitoare:** Sistemul se poate adapta pentru alte departamente:

-  **GIS** - informații cadastrale, urbanism
-  **Regii locale** - apă, canalizare, salubritate
-  **Spații verzi** - autorizații, regulamente
-  **Asistență socială** - beneficii, proceduri
-  **Administrație generală** - proceduri, documente

Widget-ul se integrează pe site-urile primăriilor și oferă răspunsuri instant din documentele oficiale ale instituției.

---



## Funcționalități Cerute

### 1. Chat Widget (Frontend)

- **Buton chat** în dreapta jos pe site-uri web
- **Popup chat** care se deschide la click
- **Interfață simplă** pentru conversații (mesaje, input, send button)
- **Responsive design** (funcționează pe mobile și desktop)
- **Integrare ușoară** - 1 linie de cod JavaScript pe site

### 2. Backend AI

- **API REST** pentru procesarea mesajelor
- **Răspunsuri în română** la întrebări despre legislația fiscală
- **Citarea surselor** - să specifice din ce document vine informația
- **Upload documente** - posibilitatea de a adăuga noi PDF-uri cu legi/regulamente
- **Multi-tenant** - suport pentru mai multe primării cu documente diferite

### 3. Administrare

- **Upload PDF-uri** cu legislație fiscală
  - **Configurare per primărie** (documente specifice pentru fiecare)
  - **Istoric conversații** pentru analiză
- 

## **Specificații Tehnice**

### **Stack Tehnologic Obligatoriu:**

Frontend Widget: HTML + CSS + JavaScript (vanilla, fără framework)

Backend: Python + FastAPI

Database: PostgreSQL

AI Model: Llama 3.2 3B prin Ollama

RAG System: LangChain + Chroma

Embeddings: sentence-transformers (pentru română)

### **Cerințe Hardware Inițiale:**

- **Dezvoltare:** Laptop cu 8GB+ RAM
- **Testare:** Llama 3.2 3B local prin Ollama
- **Producție:** Upgrade path la server dedicat

### **Cerințe Software:**

- Python 3.11+
  - PostgreSQL 15+
  - Ollama pentru model serving
  - Cross-browser compatibility (Chrome, Firefox, Safari, Edge)
- 

## **Structura Livrabilelor**

### **1. Chat Widget**

```

└─ widget/
   └─ fiscal-chat-widget.js  # Widget complet (single file)
   └─ fiscal-chat-widget.css # Stiluri
   └─ demo.html             # Pagină de test
   └─ integration-guide.md  # Documentație integrare
```

### **Cerințe widget:**

- Funcționează standalone (nu depinde de alte librării)
- Dimensiune max 100KB
- Configurabil prin parametri (culori, poziție, text)
- Loading states și error handling

## 2. Backend API

```
└─ backend/  
  └─ main.py          # FastAPI application  
  └─ models/          # SQLAlchemy models  
  └─ api/              # API endpoints  
  └─ services/         # Business logic  
  └─ ai/               # AI/RAG implementation  
  └─ requirements.txt  # Dependencies
```

### Endpoints minime:

```
POST /api/chat          # Procesare mesaj chat  
POST /api/documents/upload # Upload PDF-uri  
GET  /api/health        # Health check  
GET  /api/municipalities # Lista primării
```

## 3. Database Schema

```
sql  
  
-- Primării  
municipalities (id, name, domain, created_at)  
  
-- Conversații  
conversations (id, municipality_id, session_id, created_at)  
messages (id, conversation_id, role, content, sources, created_at)  
  
-- Documente  
documents (id, municipality_id, filename, category, processed_at)
```

## 4. AI Components

- **Document processing:** PDF → text chunks pentru RAG
- **Vector storage:** Embeddings în Chroma
- **Query processing:** Intent recognition + context retrieval

- **Response generation:** Structured prompts pentru Llama
- 

## Criterii de Acceptanță

### Funcțional:

- ✓ Widget se integrează pe orice site cu 1 linie de cod
- ✓ Răspunde corect la întrebări despre taxe/impozite din documentele uploadate
- ✓ Citează sursa informației (document, articol, pagină)
- ✓ Suportă mai multe primării cu documente diferite
- ✓ Timp de răspuns sub 10 secunde
- ✓ Interfață user-friendly și intuitivă

### Tehnic:

- ✓ Backend rulează stabil pe laptop cu 8GB RAM
- ✓ Suportă minimum 10 utilizatori concurenți
- ✓ Upload și procesare automată PDF-uri
- ✓ Validare și sanitizare input-uri
- ✓ Error handling complet
- ✓ Logs pentru debugging

### Documentație:

- ✓ README cu instrucțiuni de instalare
  - ✓ Documentație API
  - ✓ Ghid de integrare widget
  - ✓ Exemple de configurare pentru primării
- 

## Etape de Dezvoltare

### Faza 1: MVP

- ☐ Setup Ollama + Llama 3.2 3B
- ☐ Backend FastAPI cu endpoint /chat
- ☐ Widget basic funcțional
- ☐ RAG simplu cu 5-10 PDF-uri test
- ☐ PostgreSQL setup
- ☐ Demo funcțional pe localhost

### Faza 2: Production Ready

- ☐ Multi-tenant support
- ☐ Upload documents API
- ☐ Widget styling și configurare
- ☐ Error handling și validări
- ☐ Docker setup pentru deployment
- ☐ Documentație completă

### Faza 3: Testing & Deployment

- ☐ Testing comprehensiv
- ☐ Performance optimization
- ☐ Security review
- ☐ Production deployment
- ☐ Training și handover

---

### Rezultat Final Așteptat

Un **produs funcțional** care:

- Pentru cetățeni:** Buton chat pe site primărie → întrebare despre taxe → răspuns instant cu surse
- Pentru primării:** Reducerea apelurilor telefonice + servicii 24/7
- Pentru dezvoltatori:** Cod curat, documentat, ușor de mentinut și extins

#### Demo scenario:

Cetățean accesează site primărie  
→ Click pe chat bot (dreapta jos)  
→ Scrie: "Care e taxa pe clădiri în 2024?"  
→ AI răspunde: "Conform HCL 45/2024, taxa pe clădiri este 0.2% din valoarea cadastrală. Sursa: Anexa 3, pag. 15"  
→ Cetățean mulțumit, nu sună la telefon

**Scalabilitate viitoare:** Odată implementat pentru taxe și impozite, același sistem se poate extinde rapid pentru:

- Departamentul GIS (urbanism, cadastru)
- Regiile locale (utilități)
- Asistența socială (beneficii)
- Alte servicii publice

Arhitectura modulară permite adăugarea de noi domenii fără refactorizare majoră.

