Product Requirements Document (PRD)

Project Title: AI Todo App - Intelligent Multilingual Task Management

Developer: Vibha Rawan

Date: August 05, 2025

Deployment Link:

GitHub Repo:

## 1. Overview

The AI Todo App is an intelligent multilingual task management application that combines traditional task management with AI-powered translation capabilities. Users can:

* Create, manage, and organize tasks with priorities, statuses, and tags
* Translate tasks into 6 supported languages (English, Spanish, French, German, Italian, Hindi)
* Use a hybrid translation system (predefined dictionary + GPT-4 AI fallback)
* Filter and search tasks with advanced capabilities
* View comprehensive analytics and productivity insights
* Experience a fully localized interface in their preferred language

This app was built using Streamlit with custom CSS styling and integrates OpenAI's GPT-4o-mini API for intelligent translations while using predefined dictionaries for common phrases to optimize costs.

## 2. Objectives

* **Primary Goal:** Create a comprehensive task management solution that breaks language barriers
* **AI Integration:** Implement hybrid translation system using both dictionary lookup and GPT-4 AI
* **User Experience:** Provide intuitive, multilingual interface with real-time analytics
* **Performance:** Optimize translation costs through intelligent caching and fallback systems
* **Deployment:** Successfully deploy on Vercel.

## 3. Target Users

* **Multilingual Professionals:** Teams working across different languages who need task coordination
* **International Students:** Students managing coursework and projects in multiple languages
* **Global Remote Workers:** Distributed teams requiring task management with translation support
* **Language Learners:** Users wanting to practice task management in different languages

## 4. Functional Requirements

### Core Task Management (Create Task)

* **Create Task:** Users can add tasks with title, description, priority (High/Medium/Low), and tags
* **Task Status Management:** Three-state system (Todo, In Progress, Completed) with visual indicators
* **Task Organization:** Tag-based categorization with comma-separated input
* **Task Deletion:** Remove tasks with confirmation to prevent accidental deletion
* **Priority System:** Color-coded priority levels (High, Medium, Low)

### Multilingual Interface (Manage Task)

* **6 Language Support:** English, Spanish, French, German, Italian, Hindi
* **Complete UI Translation:** All interface elements translated
* **Language Switching:** Real-time language switching without data loss

### Analytics & Insights (Analytics)

* **Task Statistics:** Completion rates, status breakdowns, priority distributions
* **Translation Analytics:** API usage tracking, cache efficiency metrics

## 5. Design Requirements

### Visual Design

* **Modern Streamlit Interface:** Clean, professional layout
* **Responsive Layout:** Three-column grid system with collapsible sidebar
* **Color-coded System:** Consistent color scheme for priorities and statuses
* **Card-based Design:** Individual task cards with shadows and rounded corners

### User Experience

* **Tab-based Navigation:** Three main tabs (Create, Manage, Analytics)
* **Sidebar Utilities:** Quick stats, actions, and bulk operations
* **Progressive Disclosure:** Expandable sections for detailed information
* **Intuitive Icons:** Clear iconography for all actions and states
* **Loading States:** Visual feedback during API calls and processing

### Accessibility

* **Semantic Structure:** Proper heading hierarchy and content organization
* **Keyboard Navigation:** Full keyboard accessibility for all features
* **Screen Reader Support:** Proper ARIA labels and descriptions
* **High Contrast:** Sufficient color contrast for all text elements

## 6. API Details

### Primary Translation API

**API Used:** OpenAI GPT-4o-mini  
**Endpoint:** https://api.openai.com/v1/chat/completions  
**Method:** POST  
**Authentication:** Bearer token (API key)

**Request Configuration:**

* **Model:** gpt-4o-mini-2024-07-18
* **Max Tokens:** 150 per translation
* **Temperature:** 0.3 (for consistent results)
* **System Prompt:** Professional translator instructions

### Fallback System

**Primary:** Predefined translation dictionary with 25+ UI elements and 17+ task phrases per language  
**Secondary:** GPT-4 API for complex or unique content

### Error Handling

* **API Failures:** Graceful fallback to original text
* **Rate Limiting:** Built-in request tracking and management
* **Connection Issues:** User-friendly error messages with retry options

## 7. Technical Stack

**Frontend Framework:** Streamlit 1.28+  
**Programming Language:** Python 3.8+  
**Styling:** Custom CSS injection via st.markdown  
**AI Integration:** OpenAI Python SDK  
**Data Storage:** Streamlit session\_state (client-side)  
**Environment Management:** python-dotenv for configuration  
**Deployment:** Streamlit Cloud  
**Version Control:** GitHub with full documentation

# 8. Deployment

- Hosted live on Streamlit Cloud  
- GitHub repository publicly accessible with full source code and documentation

# 9. Testing Notes

- Manually tested task creation, completion, and translation flows  
- Tested all major languages in lang\_map  
- Error handling added for failed API responses

# 10. Development Time Tracking

Streamlit setup: ~1 hour  
Implementing task logic: ~1.5 hours  
Translation logic + API integration: ~1.5 hours  
Testing + bug fixes: ~1 hour  
Deployment & GitHub setup: ~0.5 hour  
Loom video creation: ~0.5 hour  
Total: ~7 hours