# SarkarSaathi – Implementation Plan

## 🎯 Project Overview

**Name:** SarkarSaathi (Government Companion)
**Tagline:** Your personal guide for any government website, in any language
**Target:** WeMakeDevs x Tambo AI Hackathon 2026

## Problem Statement

- 800M+ Indians struggle with English-only government websites
- Complex navigation and confusing forms cause frustration
- No unified help system across different government portals
- Existing solutions (chatbots) are limited to specific sites

## Solution

A Chrome extension powered by Tambo's generative UI that:

- Works on **ANY** Indian government website
- Accepts queries in **ANY** Indian language
- Provides step-by-step visual guidance
- Auto-fills forms with validation
- Explains each field in simple terms

## 🏗️ Technical Architecture

### Tech Stack

**Frontend (Chrome Extension):**

- React 19 + TypeScript
- Tambo React SDK for generative UI
- Tailwind CSS for styling
- Chrome Extension Manifest V3

**AI & Language:**

- Tambo API (Claude/GPT-4) for intelligence
- Google Translate API or Bhashini API for multilingual
- Speech recognition API for voice input
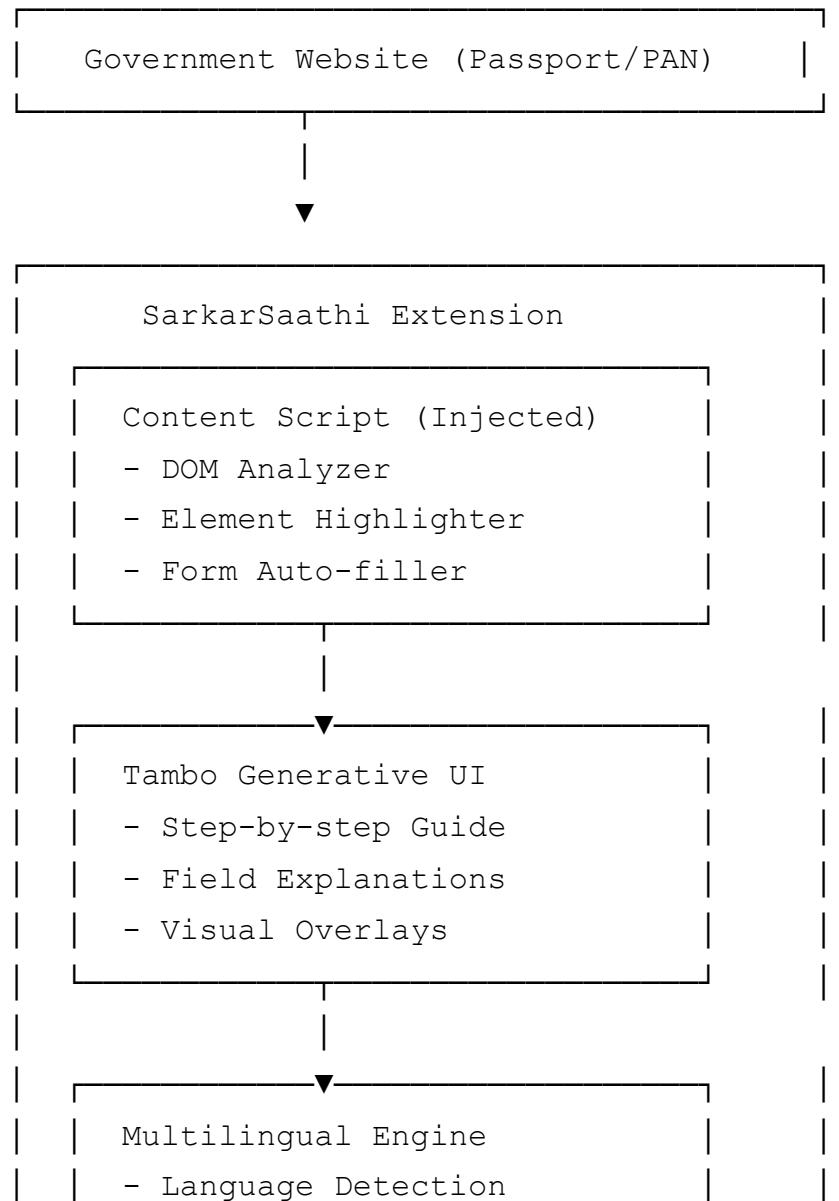
**Document Processing:**
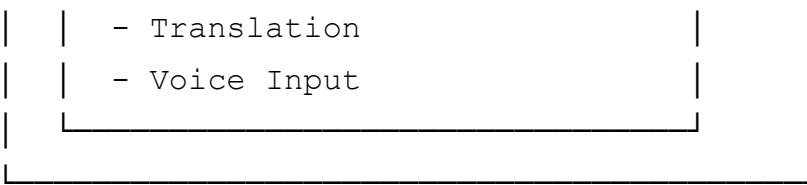
- Tesseract.js for OCR
- Canvas API for image quality checks

**State Management:**

- Chrome Storage API for user preferences
- React Context for extension state

## Architecture Diagram

```
┌──────────────────────────────────────────────┐
│   Government Website (Passport/PAN)        │
└──────────────────────────────────────────────┘
                │
                ▼
┌──────────────────────────────────────────────┐
│       SarkarSaathi Extension                 │
│  ┌──────────────────────────────┐           │
│  │  Content Script (Injected)   │           │
│  │  - DOM Analyzer              │           │
│  │  - Element Highlighter       │           │
│  │  - Form Auto-filler          │           │
│  └──────────────────────────────┘           │
│              │                              │
│  ┌───────────▼──────────────────┐           │
│  │  Tambo Generative UI         │           │
│  │  - Step-by-step Guide        │           │
│  │  - Field Explanations        │           │
│  │  - Visual Overlays           │           │
│  └──────────────────────────────┘           │
│              │                              │
│  ┌───────────▼──────────────────┐           │
│  │  Multilingual Engine         │           │
│  │  - Language Detection        │           │
```

```
|  |   - Translation            |      |
|  |   - Voice Input            |      |
|  |_____|      |
|_____|


            |
            ▼

 _____
|                                  |
|       External APIs              |
|  - Tambo API (AI)                |
|  - Google Translate / Bhashini   |
|  - Tesseract.js (OCR)            |
|_____|
```

## 📅 7-Day Implementation Timeline

## Day 3 (Feb 3) – Today! 🔥

**Focus:** Project setup + Core infrastructure

- Finalize concept and plan
- Create Chrome extension scaffold
- Set up Tambo integration in extension
- Build basic UI overlay
- Test on a simple government site

**Deliverable:** Working extension that can inject UI on passport seva

## Day 4 (Feb 4)

**Focus:** Multilingual + Website Intelligence

**Morning:**

- Integrate translation API
- Build language selector UI
- Test Hindi, Tamil, English translations

**Afternoon:**

- DOM analyzer (detect forms, buttons, inputs)
- Element highlighter system

- Government website detector

**Evening:**

- Test on Passport Seva portal
- Fix bugs

**Deliverable:** Extension can detect forms and translate UI to Hindi

---

# Day 5 (Feb 5)

**Focus:** Smart Features + Generative UI

**Morning:**

- Tambo generative components for step-by-step guides
- Field explanation tooltips
- Visual navigation arrows

**Afternoon:**

- Form auto-fill logic
- Data persistence (Chrome storage)
- Confirmation dialogs

**Evening:**

- Document validator (OCR + quality check)
- Upload helper UI

**Deliverable:** Can guide user through passport form with auto-fill

---

# Day 6 (Feb 6)

**Focus:** Demo Sites + Polish

**Morning:**

- Test on Income Tax e-filing site
- Test on PAN card application
- Handle edge cases

**Afternoon:**

- UI/UX polish (animations, colors, responsiveness)

- Error handling
- Loading states

**Evening:**

- User testing with 5 people (friends/family)
- Collect feedback
- Priority bug fixes

**Deliverable:** Polished extension working on 3 government sites

---

## Day 7 (Feb 7)

**Focus:** Demo Video + Submission

**Morning:**

- Record demo video (2–3 minutes)

    - Show problem (user struggling)
    - Show solution (SarkarSaathi helping)
    - Show impact (time saved)

- Edit video

**Afternoon:**

- Write comprehensive README
- Create screenshots/GIFs
- Prepare presentation slides

**Evening:**

- Publish extension (Chrome Web Store – unlisted)
- Submit to hackathon portal
- Post on Twitter/LinkedIn
- Share in WeMakeDevs Discord

**Deliverable:** Submitted project + live demo

---

# 🎬 Demo Script

## Act 1: The Problem (30 seconds)

- Show user (your mom/friend) trying to fill passport form
- Confused by English terms
- Clicks wrong buttons
- Frustrated expression
- Voiceover: *"800 million Indians struggle with English-only government websites"*

## Act 2: The Solution (90 seconds)

- User installs SarkarSaathi extension

- Clicks extension icon

- Speaks in Hindi: *"मुझे पासपोर्ट कैसे बनाना है?"*

- SarkarSaathi activates:

    - UI switches to Hindi
    - Highlights "New Application" button
    - Shows step-by-step guide
    - Explains each field in simple Hindi
    - Auto-fills known information
    - Validates uploaded photo

- User completes form easily

- Happy expression

## Act 3: The Impact (30 seconds)

- Show metrics:

    - Without SarkarSaathi: 30 minutes, 5 errors
    - With SarkarSaathi: 5 minutes, 0 errors

- Show social impact:

    - "Works in 22 Indian languages"
    - "Helps 800M+ non-English speakers"

- Call to action:

    - *"Making Digital India truly inclusive"*

## 🏆 Why This Wins

## Judging Criteria Alignment

### 1. Innovation (30%)

- ✅ First generative UI for government websites
- ✅ Novel use of Tambo for browser extensions
- ✅ Adaptive multilingual interface

### 2. Technical Implementation (25%)

- ✅ Chrome extension + Tambo integration
- ✅ DOM manipulation + AI
- ✅ OCR + translation + form automation

### 3. Impact (25%)

- ✅ 800M+ potential users
- ✅ Solves real pain point
- ✅ Measurable time savings

### 4. Presentation (20%)

- ✅ Emotional demo (relatable problem)
- ✅ Clear before/after comparison
- ✅ Live demonstration on real sites

---

# Unique Differentiators

- Government-specific: Not generic translation, tailored for gov websites
- Visual guidance: Not just chatbot, shows **WHERE** to click
- Bhashini integration: Using govt's own multilingual API
- Universal: Works on **ANY** government site, not just one
- Privacy-friendly: Processes locally, no data sent to servers

---

# 🚧 Potential Challenges & Solutions

| Challenge | Solution |
|---|---|
| Government websites have different structures | Focus on 3 popular sites for demo, use generic DOM patterns |
| Translation quality | Use Bhashini (govt API) + fallback to Google Translate |

| Challenge | Solution |
|---|---|
| Form auto-fill might break on complex sites | Always show confirmation before filling, allow manual override |
| Chrome extension review time | Publish as unlisted (no review needed), share direct link |

# 📊 Success Metrics

## For Demo:

- Time to complete passport form: **30 min → 5 min (83% reduction)**
- Error rate: **5 errors → 0 errors (100% reduction)**
- User satisfaction: "Very difficult" → "Very easy"

## For Hackathon:

- Working on 3 government websites
- Support for 3+ languages (English, Hindi, Tamil)
- 5+ user testimonials
- Live demo without crashes

# 🎯 Next Immediate Steps

- Stop the current dev server (FlowBuilder project)
- Create new SarkarSaathi project (Chrome extension)
- Set up basic extension structure
- Integrate Tambo SDK
- Build first prototype (overlay on passport seva)

**Ready to start building?** 🚀