Summary Document – AI Solutions for Nonprofits

Overview: This repository contains two complementary projects aimed at accelerating digital transformation for non-profit organisations using AI. Both projects were developed as part of an AI enablement initiative to empower nonprofits: one addresses *specific challenge analysis and solution design*, and the other assesses *overall organisational readiness* for AI and digital projects. Together, these tools guide nonprofits from understanding their readiness level to implementing targeted tech solutions, forming a holistic AI/digital transformation journey for a nonprofit. The content is designed to be accessible to both technical and non-technical stakeholders, highlighting practical value while also noting technical implementation where relevant.

Task 1 — AI Architect for Nonprofit Solutions

Target Users

This tool is designed for nonprofit program managers, operations leads, and executive directors who need clear, tailored guidance on technology choices without hiring a full technical team. It also serves advisors and consultants who steer digital transformation or system selection across multiple clients, as well as smaller organisations that lack in-house capacity but still need to make confident, timely decisions.

Ethical Considerations

We minimise risk by keeping inputs problem-oriented and discouraging the sharing of personally identifiable information. Each recommendation comes with a rationale and explicit assumptions, so teams don't over-index on AI output, and every response contains a "verify before purchase/implementation" note.

Prompts are tuned to surface open-source and low-cost options alongside commercial tools to support equity across organisation sizes and regions. Human oversight is explicit: the tool frames options and trade-offs, while decisions remain with staff. Language and structure are intentionally plain so mixed technical/non-technical teams can act together.

Technical Approach

A lightweight Flask API orchestrates a modular prompt pipeline and tags each session with a short, human-readable problem ID for traceability. Analysis runs on GPT-4 using nonprofit-tuned

prompts and supports two modes: a fast single-screen Q&A for quick scoping, and an enhanced dialogue that gathers context before recommending options.

We evaluated local LLMs (e.g., Llama-3) for cost and data control, and serverless vs. containerised deployment;endors such as Claude or Gemini for guardrails and reasoning style, as well as adding retrieval-augmented generation. The current stack optimises speed, portability, and minimal infrastructure while keeping those alternatives open.

Creative Features (Beyond Requirements)

Beyond the baseline, the Architect offers a dual-mode workflow so busy teams can either move quickly or pause for a short consultative exchange. The auto-generated problem identifier lets organisations track multiple threads without a heavy ticketing system. Outputs are intentionally actionable—each recommendation pairs a rationale with concrete first steps and low/medium/high-effort paths—and the text is easy to paste into briefs, proposals, or issue trackers.

AI-Assisted Development

We used LLM assistance to prototype prompts, tighten error messages, and generate test cases that reflect common nonprofit scenarios. Code-suggestion tools accelerated boilerplate routes and schema scaffolding, and iterative LLM critiques helped remove jargon so the guidance reads as practical advice rather than generic tech marketing.

Task 2 — AI & Digital Transformation Readiness Assessment Tool

Target Users

The assessment is aimed at executive and strategy leaders, digital/IT managers, and board committees who need a shared, comparable picture of their organisation’s readiness. It also helps funders and intermediaries understand grantee baselines and supports capacity-building teams that want to benchmark progress over time without imposing heavy reporting burdens.

Ethical Considerations

We collect organisation-level practices rather than personal data and explain the purpose and storage up front. Scoring follows a four-tier rubric—“Needs Foundation,” “Emerging,” “Established,” and “Advanced”, to emphasise growth pathways rather than rigid labels. We document how weights and thresholds work and pair every numeric result with a short narrative explaining why the score looks the way it does.

Writing is plain language with optional examples and visuals, and results can be downloaded for offline sharing or board packets.

Technical Approach

The UI is built in React with progress tracking and input validation, backed by a Node/Express API that runs a category-weighted scoring engine to produce both numeric and narrative results. Immediately after submission, users see charts by dimension and an overall tier as well as AI-generated insights tailored to their responses (via GPT-4). Results can be exported as Markdown or PDF for distribution. We used faster no-/low-code options (Replit and Cursor) for code generation and improvement. The chosen stack balances developer velocity, UX quality, and extensibility, for example, adding RAG or administrative dashboards later.

Creative Features (Beyond Requirements)

The framework spans six dimensions: leadership, people, data, technology, governance, and change-enablement—so the picture stays holistic rather than tool-centric. The four-tier scale sets clear milestones while leaving room for nuance. AI commentary reads like an expert coach, highlighting quick wins, sequencing larger moves, and making trade-offs explicit. Exportable summaries make it easy to brief boards, funders, and cross-functional teams.

AI-Assisted Development

For evidence gathering and design, we used GPT-5 Deep Research to review existing capacity and readiness instruments and to synthesise the methodology and full questionnaire, mapping dimensions, weights, and scoring rules, checking for bias and clarity, and extracting best-practice language. We then refined long-form questions with examples and branching logic, wrote result-narration prompts that surface tailored insights and assumptions, and used code-assist tools for component scaffolding, validation patterns, and test-data generation.

How the Two Tools Work Together

Begin with the readiness assessment to establish a shared baseline and agree on priorities. Once a focus area is selected, the Architect turns that priority into concrete solution paths with rationale, effort estimates, and first steps. Over time, teams can re-take the assessment, compare results, and use the Architect again to design the next wave of improvements—creating a simple loop from diagnosing needs to implementing practical change.