



# Project Title: Auto-Architect AI

## *Automated Software Planning & Resource Estimation System*

**Version:** 1.0.0

**Date:** February 03, 2026

**Prepared By:** Business Automation LTD

## 1. Executive Summary

**Auto-Architect AI** is an advanced Multi-Agent SaaS platform designed to revolutionize the software onboarding process. By leveraging Large Language Models (LLMs) and a "Human-in-the-Loop" architecture, the system transforms raw client requirements into comprehensive technical documentation, resource planning, and cost analysis reports within minutes. Unlike generic AI tools, this system integrates a real-time **Cost Intelligence Engine** to provide financial transparency regarding development tools and infrastructure.

## 2. Problem Statement

In the traditional software development lifecycle (SDLC), the initial "Discovery Phase" faces significant challenges:

- **Time-Consuming:** Creating a detailed PRD/SRS manually takes 7–14 days.
- **Resource Gap:** Clients often underestimate the need for paid tools (e.g., Cloud, AI licenses), leading to budget overruns.
- **Inconsistency:** Communication gaps between Sales, Analysts, and Developers result in vague requirements.

## 3. Solution Overview

We propose an **Agentic AI Workflow** orchestrated by **LangGraph**, consisting of 5 specialized AI Agents. The system ingests raw client inputs (text/files), analyzes requirements, allows human validation, and outputs a professional project blueprint.


## Key Differentiators:

1. **Human-in-the-Loop:** Critical validation gate to prevent AI hallucinations.
2. **Cost Intelligence:** Automated calculation of 3rd-party tool costs (ROI analysis).
3. **Role-Based Processing:** Specialized agents for Analysis, Architecture, Finance, and Management.

## 4. System Architecture & Workflow

The system operates on a **Sequential & Parallel Processing Model**:

### Phase 1: Ingestion & Analysis

- **Input:** Client provides project details via Chat or File Upload (PDF/Docx).
- **Agent 1 (The Analyst):** Extracts functional requirements, user stories, and constraints.
-  **The Human Gate:** The workflow pauses. The Admin reviews the extracted feature list via a dashboard, edits if necessary, and clicks "**Approve**".

### Phase 2: Technical & Financial Planning (Parallel Execution)

Once approved, the data flows to three distinct agents:

- **Agent 2 (The Architect):** Determines the Technology Stack (e.g., MERN, Python), Database Schema, and Cloud Architecture.
- **Agent 3 (The Finance Expert):** Queries the internal `Tools_Pricing_DB` to recommend necessary tools (Jira, GitHub Copilot, AWS) and calculates estimated monthly costs.
- **Agent 4 (The Project Manager):** Estimates development time, team composition, and creates a Sprint-based timeline.

### Phase 3: Documentation & Delivery

- **Agent 5 (The Technical Writer):** Aggregates data from all previous agents.
- **Output:** Generates a branded PDF containing the Scope, Tech Stack, Cost Breakdown, and Timeline.

## 5. Detailed Agent Specifications

Agent Name	Role	Core Responsibility	Model/Logic
Agent Alpha	Business Analyst	Requirement Extraction & Gap Analysis.	GPT-4o / Prompt Eng.
Agent Beta	Solution Architect	Tech Stack Selection & System Design.	Claude 3.5 Sonnet
Agent Gamma	Cost & Tools Expert	Tool Recommendations & Budget Estimation.	RAG (Retrieval Augmented Generation)
Agent Delta	Project Manager	Timeline estimation & Resource Allocation.	Custom Logic + LLM
Agent Epsilon	Tech Writer	Final Report Compilation & Formatting.	GPT-4o

## 6. Technology Stack

### Backend & AI

- **Framework:** Python (FastAPI) - for high-performance API.
- **Orchestration:** LangGraph - for stateful multi-agent workflows and human-interrupt logic.
- **LLMs:** OpenAI GPT-4o (Logic) & Anthropic Claude 3.5 Sonnet (Coding/Architecture).
- **Database:** PostgreSQL (User Data) + Vector DB (Knowledge Base).

### Frontend

- **Framework:** Next.js 14 (React) - for a responsive Client/Admin Dashboard.
- **Visualization:** Mermaid.js (for rendering live flowcharts).
- **Styling:** Tailwind CSS.

Infrastructure

- **Hosting:** AWS or Vercel.
- **Authentication:** Clerk / NextAuth.

7. The "Cost Intelligence" Module (Sample Output)

One of the core deliverables in the final documentation will be the Financial Breakdown:

Table 1: Infrastructure & Tools Investment

Tool Category	Recommended Solution	Monthly Cost (Est.)	Business Impact (ROI)
AI Assistant	GitHub Copilot Business	\$19 / user	Increases dev velocity by ~40%.
Project Mgmt	Jira Software	\$7.75 / user	Ensures agile tracking & transparency.
Cloud Hosting	AWS (EC2 + RDS)	~\$65.00	Scalable infrastructure with 99.9% uptime.
Security	Snyk	Free / \$25	Vulnerability scanning for secure code.

8. Implementation Roadmap

- **Milestone 1: Core Logic (Weeks 1-3)**
  - Setup LangGraph environment.
  - Develop Analyst and Architect Agents.
- **Milestone 2: Human Loop & Finance (Weeks 4-5)**
  - Implement the "Interrupt" logic for human review.
  - Build the Tools\_Pricing\_Database.
- **Milestone 3: UI & Integration (Weeks 6-7)**
  - Develop the Admin Dashboard.
  - PDF Generation Integration.
- **Milestone 4: Beta Launch (Week 8)**
  - Testing with dummy client data.

## 9. Conclusion

**Auto-Architect AI** will transform the agency's pre-sales and planning process. By automating the repetitive task of documentation while keeping human oversight for accuracy, we ensure a **90% reduction in planning time** and **100% transparency in cost estimation** for our clients.

