Agentic AI for Life Events

Agentic Government Service Agents to transform interactions with public services.

Understanding AI Agents

Agentic systems combine intelligence with action, giving models the ability to employ tools, make decisions, and adapt to new situations.

Can agentic systems navigate complex government systems to handle a life event - e.g. the Birth of a Child?

Autonomous Task Execution

Agents decompose goals into subtasks, execute them in order, handle errors, and verify completion.

Multi-System Orchestration

Navigate different portals, understand forms, coordinate between agencies, and maintain context.

Intelligent Document Processing

Extract information from documents, validate accuracy, match to requirements, and handle various formats.

Adaptive Problem Solving

Interpret errors, research solutions, try alternatives, and escalate to humans when necessary.

Opportunity

Life events require citizens to navigate complex processes involving multiple government agencies and significant time and effort.

We propose designing an agentic AI system to automate live-event interactions.

Key Questions

1. Can AI agents reliably manage multi-agency government interactions?
2. Can we ensure security and compliance while enabling automation?
3. What's the optimal technical architecture using modern agent frameworks?
4. How do we integrate digital identity and payment systems?

Multi-Agency Coordination	Key Technologies	Timeline
Integration across government departments	LangChain, LangGraph LlamaIndex (?), Airflow (?)	Three months to demo (end of August)

LangChain and LangGraph

LangChain

Industry standard for creating LLM-powered applications.

Framework for building agents that can reason, use tools, and maintain memory across interactions.

- **Chains:** Composable units of functionality that process data sequentially
- **Agents:** Use LLMs to decide which tools to use and in what order
- **Tools:** Discrete capabilities like web search, API calls, or document processing
- **Memory:** Various strategies for maintaining context (conversation, summary, vector-based)

LangGraph

Extends LangChain with graph-based workflow capabilities for modeling processes with multiple paths and decision points.

- **State Machines:** Model complex government workflows as state graphs
- **Conditional Routing:** Different paths based on citizen circumstances
- **Parallel Execution:** Handle independent tasks simultaneously
- **Checkpointing:** Save and resume long-running processes

Intelligent Document Processing

Government processes involve extensive document handling (e.g. birth certificates, identification, proof of residence, medical records).

Requirement to extract text from PDFs and images, parse structured data, fill out forms and track status.

Intelligent Retrieval Information Extraction

Semantic search across uploaded documents, using contextual meaning and intent rather than keywords.

Pull specific data from documents (names, dates, addresses) to auto-populate forms and cross-reference information

Document Understanding and Validation

Extract relevant structured information from various formats.

Check if uploaded documents meet requirements, verify authenticity markers, and ensure all required fields are present.

Knowledge Integration and Compliance Checking

Combine information from multiple sources enabling agents to make informed decisions based on comprehensive information.

Ensure documents meet legal requirements and verify document dependencies.

Specialised tooling: LlamaIndex?

Digital Identity Integration

Can your agent use your digital ID on your behalf?

Current digital ID systems (myID) are designed for human interaction (enter code from app, approve on phone).

No public API for agents to authenticate on your behalf - yet.

Mock Authentication (For Prototype):

- Simulate the authentication flow and focus on the agent workflows assuming authentication is solved. Document how real integration will work.
- *Google Digital Credentials API:* a new web platform API that allows websites to selectively request verifiable information about the user through digital credentials such as a driver's license or a national identification card stored in a digital wallet.

Payment Integration

Integration with cards / bank transfers. Agents handle all payment flows, understanding fee structures and calculating costs upfront. Confirmations automatically collected, stored, and associated with applications.

For prototype: Use payment orchestration platforms like Stripe or Square that provide unified APIs across payment methods.

Agent Interoperability - A2A and MCP

Multiple agents may autonomously discover each other and collaborate.

Google's A2A protocol provides an open standard for agent interoperability regardless of underlying framework (e.g. LangChain).

A2A complements Anthropic's **Model Context Protocol (MCP)**, which standardises AI integration with integrate with external resources like databases, APIs, file systems, and other services.

Agent Discovery

Agents publish capabilities through "Agent Cards" describing their identity, abilities, and endpoints. Any A2A-compliant agent can discover what other agents can do.

Task Management

Define how agents create, delegate, and manage tasks across boundaries, enabling long-running processes.

Flexible Communication

Multiple modalities for agent interaction - text, forms, files, or structured data.

Opaque Execution

Agents collaborate without exposing internal state, memory, or tools, preserving intellectual property while enabling cooperation.

General Life Event Use Case

Document Collection

Identify required documents, process uploads, extract information, validate completeness

Identity Verification

Verify parent identity, link baby to parents, establish authorization

Multi-Agency Registration

E.g. Medicare, Centrelink and Registry of Births, Deaths, and Marriages

Payment Processing

Calculate fees, process payments, track receipts

Status Management

Monitor progress, handle exceptions, notify parents

Development Plan

Month 1

- Set up development environment (Python, Docker, VS Code, Git repo)
- Build simple LangChain agent that can have conversations
- Study Australian government service processes
- Implement basic document processing (extract text from PDFs)
- Get agent to complete one simple task end-to-end

Deliverable: Working chatbot that can simulate submitting a birth registration to one agency

Month 2

- Implement LangGraph workflow for birth registration process
- Add state management (track what's been completed)
- Connect to (mock) government agencies
- Build simple interface for testing
- Add error handling and retry logic
- Implement status tracking across agencies

Deliverable: Agentic system that can coordinate registration across multiple agencies with status updates

Month 3

- Add document upload and validation
- Implement (mock) payment flow
- Add conversation memory and context
- Implement basic A2A protocol for agent communication
- Test user flows
- Write technical documentation
- Presentation and live demo

Deliverable: Complete prototype with documentation and compelling demonstration