USE CASE
From Rigid Workflows to Real-Time Orchestration: How Agentic AI and Composable APIs Can Transform Healthcare Automation

Revolutionizing Healthcare Workflows with Agentic AI and Composable APIs: A Vision from Karl Fankhauser

Healthcare technology is at a critical inflection point, where the complexity of enterprise systems is limiting the efficiency and adaptability of medical professionals. The rigid workflows of Electronic Medical Record (EMR) systems require physicians to navigate cumbersome interfaces, manually input data, and wait for system-driven responses before taking action. But what if these workflows could be dynamically composed in real time, driven by physician intent rather than system constraints?

In this exclusive use case analysis, Karl Fankhauser, a leading expert who is spearheading the use of emerging technologies to promote affordability and accessibility to healthcare, explores how Agentic AI and composable APIs could fundamentally reshape how healthcare providers interact with their systems. Complementing the insights shared in our broader discussion on Composable APIs and Agentic AI in Enterprise Systems, this use case dives into a real-world scenario where an AI agent intelligently orchestrates workflows—reducing friction, increasing efficiency, and ensuring compliance with medical protocols.

Through the lens of Dr. X's patient encounter, this piece illustrates the challenges of current healthcare workflows and envisions how an AI-powered agent could transform the process—automating orders, verifying prescription constraints, and intelligently handling referrals based on clinical data. While fully autonomous workflow creation may still be on the horizon, the potential for AI-driven, API-enabled automation represents a huge leap toward composable and adaptive healthcare systems.

Let's explore how Agentic AI can bridge the gap between physician decision-making and system execution—ushering in a new era of composable, API-driven healthcare automation.

Current workflow use case

Dr. X, a primary care physician, has a visit with a patient who has a pre-existing chronic pain condition. As the patient describes their symptoms Dr. X starts to develop a possible diagnosis.   To finalize his diagnosis Dr. X wants to order some images to rule out some concerns and some lab tests as well. Also, the patient in real pain, so Dr. X wants to prescribe an opioid. To follow protocol, this means checking existing prescriptions and fulfillments of the same or other opioids – potentially at outside pharmacies. If the tests come back as he suspects he wants to send the results to his colleague Dr. Y who is a specialist in the suspected disease. Finally, Dr. X thinks that the patient should see an interventional pain specialist.

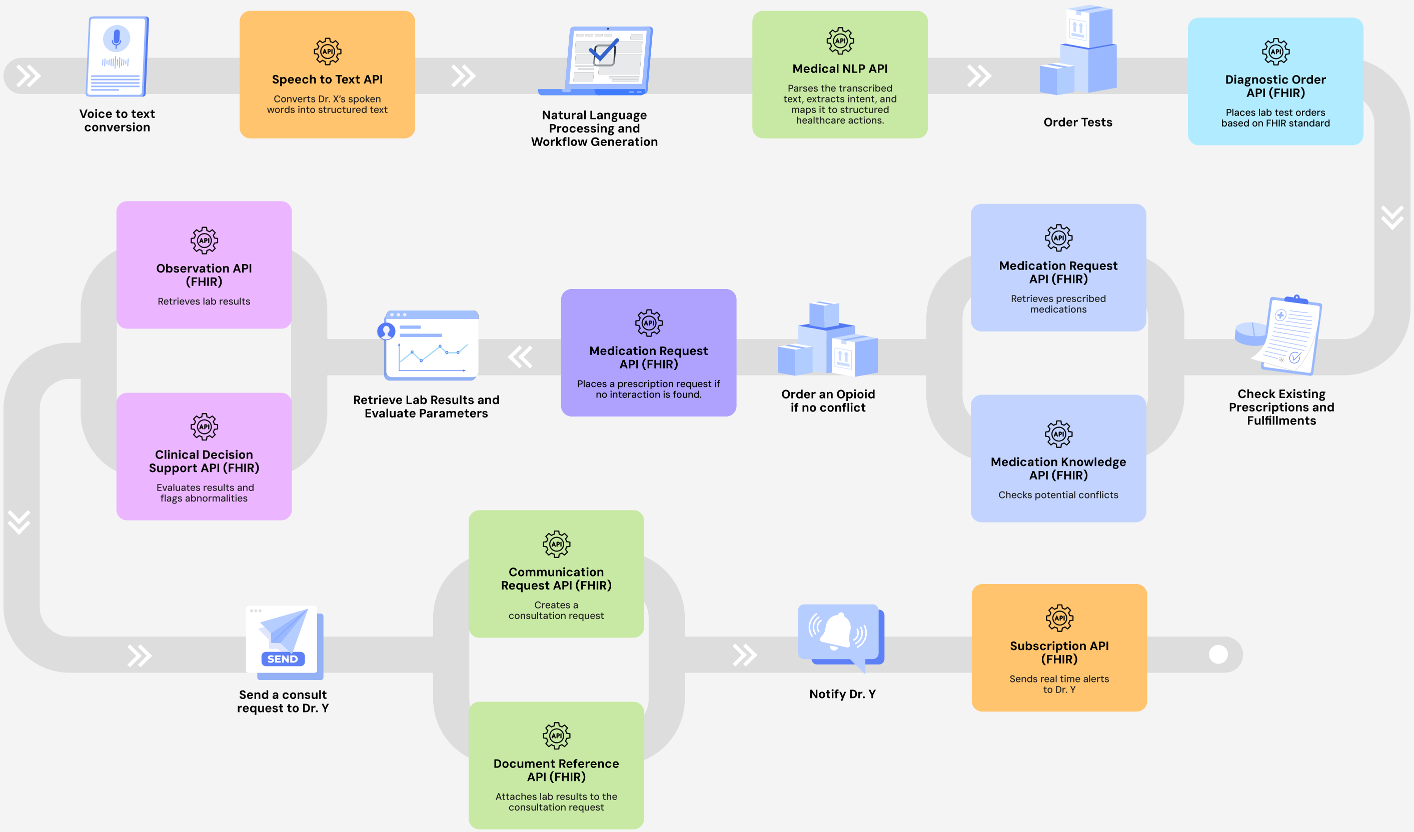
Summary

Dr. X needs to place two orders (images and labs), order one 1 prescription (that has preconditions), test that precondition at internal and external partner organizations, create two referrals with one of those being conditional on the outcome of the orders, in addition to the regular capturing of his encounter notes. There are also starts, waits and restarts in this flow.

An AI Agent Leveraging APIs for On-the-fly Workflow Creation

Revised Use Case

Dr. X sees his patient and begins to suspect a possible diagnosis. He speaks his desired workflow – order a battery of tests, check existing prescriptions and fulfillments and order an opioid if there is no conflict, if the results come back with certain parameters, send a consult request to Dr. Y along with the results. An agent converts the voice to text and the text to a flow that it implements making calls to existing APIs – some of which are FHIR standard APIs.



Voice to text conversion

Speech to Text API
Converts Dr. X's spoken words into structured text

Natural Language Processing and Workflow Generation

Medical NLP API
Parses the transcribed text, extracts intent, and maps it to structured healthcare actions.

Order Tests

Diagnostic Order API (FHIR)
Places lab test orders based on FHIR standard

Check Existing Prescriptions and Fulfillments

Medication Request API (FHIR)
Retrieves prescribed medications

Medication Knowledge API (FHIR)
Checks potential conflicts

Order an Opioid if no conflict

Medication Request API (FHIR)
Places a prescription request if no interaction is found.

Retrieve Lab Results and Evaluate Parameters

Observation API (FHIR)
Retrieves lab results

Clinical Decision Support API (FHIR)
Evaluates results and flags abnormalities

Send a consult request to Dr. Y

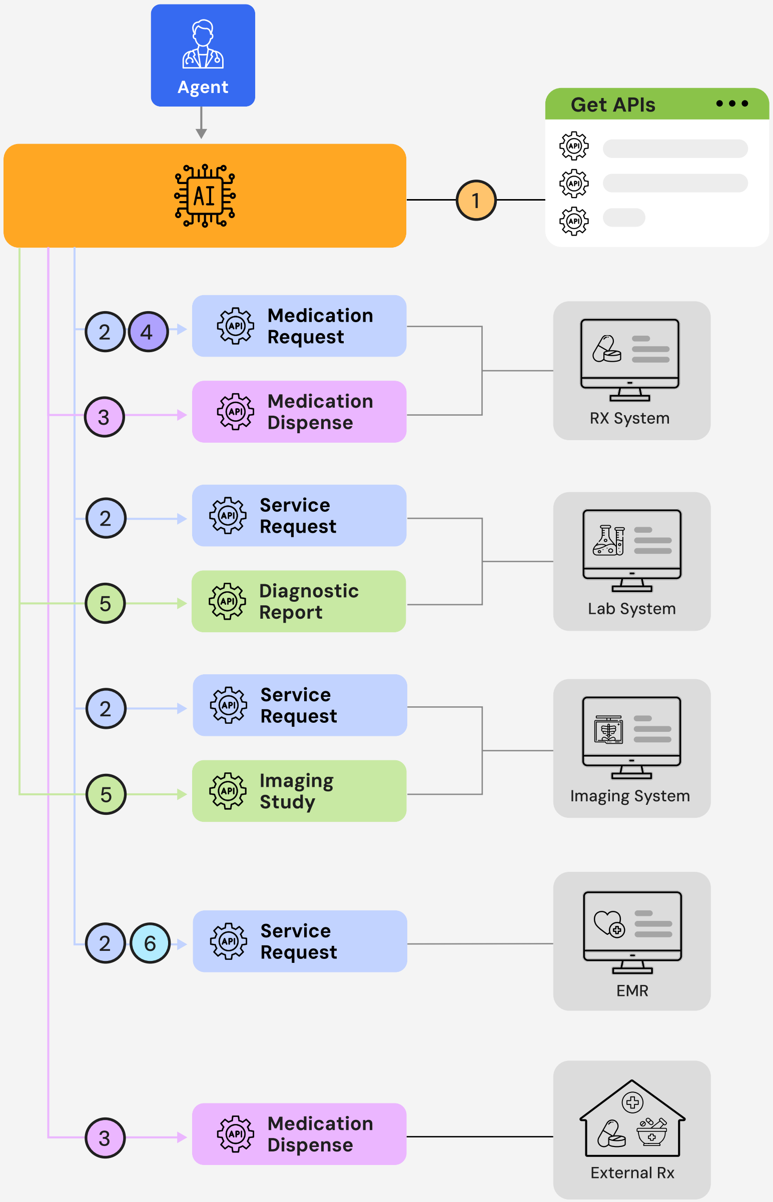
Communication Request API (FHIR)
Creates a consultation request

Document Reference API (FHIR)
Attaches lab results to the consultation request

Notify Dr. Y

Subscription API (FHIR)
Sends real time alerts to Dr. Y

If we map out our use case, we have the agent parsing the natural language into a workflow. Then it starts a series of API calls to implement this workflow:



1. As a first step the Agent discovers APIs that it can use to accomplish the various tasks
2. Next, it can launch several requests concurrently: making a service request for the labs, the images, the pain management referral and the pain medication. However, the pain medication request should return some indication that the request cannot be met without meeting the preconditions of checking other prescriptions first.
3. The agent can check internal and external pharmacies to see what has been dispensed.
4. Once that has been satisfied a new request can be sent that also includes the details on how the constraints are satisfied.
5. Something needs to trigger the Agent to go and get the Lab diagnostic report and the imaging report when they're ready. The Agent needs to parse this data and look for the guardrails indicated by Dr. X and,
6. If the results are within the parameters specified another referral should be created for Dr. X's colleague.

Moving workflow creation entirely to the end user is more visionary than practical today. There are many good reasons to have predefined flows.

- They should enforce best practices.
- Predefined flows have likely been rigorously tested – it is unlikely that something unexpected will happen in the flow.
- If an agent were to do something unexpected, in a healthcare setting, or a powerplant, lives are at stake.

There are certainly some organizational issues that need to be overcome (maybe I don't want all doctors creating flows) and probably some legal issues but let's put these questions aside and use this is use case to illustrate some of the more technical limitations and opportunities that need to be addressed to make this kind of composability possible.

While we may get to this state someday, significantly reducing internal and external IT development costs, there is probably an interim state where Agents are "developed" and tested before being introduced into production environments.

About Karl Fankhauser

Karl Fankhauser is a huge advocate of the University of Colorado where he received his MS in CS/IS. Since then, Karl has worked primarily in healthcare software and IT, with some time spent in manufacturing and telecom as well. As the Chief Architect at CareScience, he architected and delivered the solution that would later become the model for the Health Information Exchange used across the United States. This solution leveraged AI and APIs in the healthcare space when few were even thinking about these technologies. During a long tenure at Kaiser Permanente, Karl worked as Principal Architect in the privacy and security space, various functional areas in the health and as a Principal Architect and Director in Enterprise Architecture where he added care delivery and corporate services dimensions to his background. Many of the architectural decisions promoted by Karl led to huge savings and revenue gains. Currently, Karl is starting Advanced Healthcare Automation a company that focuses on using emerging technologies to promote affordability and accessibility to healthcare. Some of the technologies this new company is researching and developing includes APIs (of course), Agentic AI and drones.

You can reach out at Karl@advancedhealtcareautomation.com

About digitalML

digitalML are creators of the ignite Platform for large enterprises. One Holistic Catalog to capture, rationalize, govern, and manage the extended lifecycles of all your API types. Connected to your existing IT landscape and supplying multiple tailored API consumption experiences. Leverage ignite alongside support from digitalML's experts over every step of your API maturity journey to deliver greater efficiency and API economy benefits.

Learn more at digitalml.com