Zero-to-Hero: EGD Onboarding with swiyu

Replacing physical post office visits and clunky legacy protocols with a 30-second, fully digital Electronic Health Dossier (EGD) activation using the federal e-ID and modern FHIR standards.

The Problem: EPD's "Paper & SOAP" Bottleneck

1. **Friction in Onboarding:** Creating an EPD historically required physical presence or postal mail to guarantee identity.
2. **Missing Citizen APIs:** EPD infrastructure relies heavily on heavy IHE XDS.b protocols. There is no lightweight, patient-facing REST API in production.
3. **Complex Consent:** Managing who sees what is overly technical and disjointed for the average citizen.

The 2026 Architecture (Our Solution)

We are prototyping the missing modern interfaces, utilizing the upcoming EGD standards and the federal e-ID.

- **MyEGD (Web App):** The modern, citizen-facing frontend.
- **swiyu (e-ID):** Federated identity wallet for passwordless, secure authentication (OIDC4VP).
- **Mock EGD API:** A modern FHIR facade implementing CH:MHD, CH:PDQm, and CH:PPQm.
- **EPD Bridge (HCP Client):** Existing clinical software to prove the loop works for the doctor.

Phase 2: Real-World Doctor Lookup & Consent

The patient defines data sharing rules inside MyEGD.

1. HPD / MedReg Lookup: The patient searches for their doctor. MyEGD queries a mock of the **Swiss Healthcare Provider Directory (HPD)** via the doctor's GLN (Global Location Number).

2. Primary Care Consent (CH:PPQm):

Patient grants "Standard Access" to the selected doctor's GLN. MyEGD generates a doctor-specific FHIR `Consent` resource (`CH:PPQm` Template `301`, without delegation) in the EGD backend.

3. Secondary Use (Research):

A separate opt-in records research/HFG intent as a distinct consent artifact outside the standard CH:PPQm template set.

Phase 3: Closing the Loop (Doctor's View)

To prove the ecosystem works, we simulate the healthcare provider using the **EPD Bridge**.

- **Demographic Search:** Doctor searches for the patient. EPD Bridge queries the Mock EGD API via **CH:PDQm** and resolves the EPR-SPID.
- **Access Check:** EPD Bridge requests the patient's documents via **CH:MHD**.
- **Validation:** The Mock EGD API checks its state. Because the patient granted access to the doctor's GLN in Phase 2, the medical documents are successfully returned.

Technical Standards & APIs

Powering this modern architecture requires adherence to federal eHealth standards.

Identity & Trust

- **swiyu e-ID:** `OIDC4VP` (OpenID Connect for Verifiable Presentations).

Access & Authorization

- **IHE IUA / OAuth2:** Token-based authorization for MyEGD and EPD Bridge API.

EGD / FHIR Profiles (HL7 FHIR R4, CH EPR FHIR v5.0.0)

- **CH:PDQm:** Patient Demographics (EPR-SPID resolution & JIT creation).
- **CH:PPQm:** Privacy Policy Management (patient-to-doctor consent policies).
- **CH:MHD:** Mobile access to Health Documents (document retrieval).

Directories

- **CH:Provider / HPD:** Healthcare Provider Directory for accurate GLN lookup.

> ***Note:*** *Persistent medical-document storage is* ***out of scope*** *for this MVP. We focus on authentication, identity, and consent routing.*

Hackathon MVP Scope

What we are building live to demonstrate this flow:

1. **swiyu Handshake:** OIDC4VP integration extracting the EPR-SPID.
2. **JIT Dossier Creation:** Instantly initializing an empty EGD plus the onboarding policy (`CH:PPQm` Template `201`).
3. **HPD Integration:** Searching and selecting a doctor by GLN.
4. **Consent UX:** Generating valid FHIR `Consent` JSON payloads for onboarding (`201`) and doctor access (`301`).
5. **HCP Loop Closure:** Adapting the EPD Bridge to successfully query our Mock API and fetch documents based on the newly minted consent.