

Project Title

Maintenace & support

Maintenance and Support
24 August 2020

Problem Statement

- Multiple projects working in (OpenMRS) Switzerland, Belgium, Nepal, India, and other countries.
- Modular architecture implemented.
- Modular implementation of multiple countries.
- Non-modular implementation of multiple countries.
- High complexity of the system.
- Multiple codebase and platforms to be managed.
- Documentation and training required.
- Periodic releases.

Strategy and Results

- Provides clear tracking and management of the available modules.
- Tools management, Configuration (Ansible), Jenkins, Jenkins, Travis CI, GitHub.
- Modularization of the system through the Service Bus.
- High quality of the system.
- Robust integration of all modules.
- Modular documentation.
- Modular training and support.
- Periodic releases.
- Participate in feasibility analysis.

Problem Statement

Migration

OpenMRS modular architecture
March 2020

Problem statement

Requirement to support multiple dependencies simultaneously.

Adaptations and integrations were "code bloat"

→ difficult to upgrade, customize and share developments

Strategy

Service Bus application to modular architecture

- make the design decoupled
- easy for modules to co-exist
- easy for modules to exchange procedures

Results

Efficiency of the system has been increased.

Community of users can now use the system.

Users can now have their own specific requirements.

→ flexibility of the system.

Formal Sector Module

Formal Sector Coverage
January 2021

Problem Statement

- Several projects developed by different partners for the formal sector.
- Very complex and difficult to maintain.
- Very difficult to integrate and share developments.
- The goal is to reuse the specific features (Greece and Italy) and integrate them into the formal sector module.

Strategy

Service Bus application to modular architecture

- make the design decoupled
- easy for modules to co-exist
- easy for modules to exchange procedures

Results

- Several modules have been developed to follow formal sector demands.
- Modular approach implemented in the system.
- Several modules have been integrated.
- Interactions between modules have been reduced.
- Interactions between modules have been reduced.
- Modular approach has been adopted.
- Modular approach has been adopted.

Formal Sector API

Healthix Solutions
Academic Qualifications module

Problem Statement

Connect OpenMRS Production and Pre-prod through OpenAPIs via standard interfaces.

Modular system.

Interoperability.

High quality.

Standardized exchange of data.

Strategy

Service Bus application to modular architecture

- make the design decoupled
- easy for modules to co-exist
- easy for modules to exchange procedures

Results

- Development of a single API layer that connects OpenMRS Production and Pre-prod through OpenAPIs via standard interfaces.
- Modular system.
- Interoperability.
- High quality.
- Standardized exchange of data.

Accident Insurance

AI-based Claim Categorization
20 August 2020

Problem Statement

- All claims are coming in a single format.
- Claims are coming from various sources.
- Claims are coming in various formats.
- Claims are coming in various formats.
- Claims are coming in various formats.

Strategy

Machine learning model to categorize the claims.

- Data collection and pre-processing.
- Feature engineering.
- Model selection.
- Model training.
- Model evaluation.
- Model deployment.

Results

- Development of a single API layer that connects OpenMRS Production and Pre-prod through OpenAPIs via standard interfaces.
- Modular system.
- Interoperability.
- High quality.
- Standardized exchange of data.

AI

AI-based Claim Categorization
20 August 2020

Problem Statement

- All claims are coming in a single format.
- Claims are coming from various sources.
- Claims are coming in various formats.
- Claims are coming in various formats.
- Claims are coming in various formats.

Strategy

Machine learning model to categorize the claims.

- Data collection and pre-processing.
- Feature engineering.
- Model selection.
- Model training.
- Model evaluation.
- Model deployment.

Results

- Development of a single API layer that connects OpenMRS Production and Pre-prod through OpenAPIs via standard interfaces.
- Modular system.
- Interoperability.
- High quality.
- Standardized exchange of data.

Strategy

Results

Further Reading

[Project: 2020.T1 Maintenance](#)

[Project: 2020.T3 Modularisation](#)

[Project: D1 Formal Sector Support](#)

[Project: D1 Formal Sector APIs](#)

[Project: Accident Insurance](#)

[Project: D1 AI Claims Adjudication](#)

Q&A

In D4.0 project
SwissPHR will be a
module to store
documents from mobile apps

Sticky
notes for
questions

Is the code
open source?

yes,
available
on GitHub

What will happen if
you're replacing the AI
model with another one
in another country or another
language?
And if you don't
have training data at the
time of a new deployment?
The re-
training will
be manual or
automated?
what is a
minimum data set
requirement for the
new model to be
trained?
The re-
training is
manual.
An end-to-end
process of collecting
and processing data
from a country or
region to train a
new model.
Data collection
and processing
are done by
Healthix Solutions
and the new
model is
provided to
the customer.