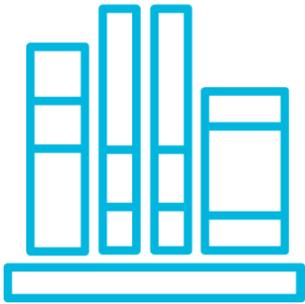


openEHR Technical Course

Welcome to study the openEHR Technical Course!

The course has been prepared in co-operation between Rosaldo Oy and Andraž Koželj. Andraž has been responsible for preparing the Course materials and is also acting as the facilitator during this course. He has been working as an independent consultant, but he is currently working as a Software Development Lead at Better.



If you study this Course together with a team, active communication with the other participants as well as with the facilitator is encouraged during your study. Chatting, commenting and asking questions will help you digest and analyze the topics from several viewpoints. If you study this Course alone, you have the chance to address your comments and questions to the facilitator.

You can also receive a certificate for completion of the Course.

The Course will familiarize you with the core technical elements of openEHR and with the openEHR architecture, where data are separated from the applications. During the Course you will learn how to utilize the Clinical Knowledge Manager when designing your own data model. Furthermore, you will get to know the openEHR API and will start working with the data. You will also learn how to create a modern user interface with the help of low code tools.



TRAINING CONTENT

The training is instructed as follows:

The Course is structured in 5 parts:

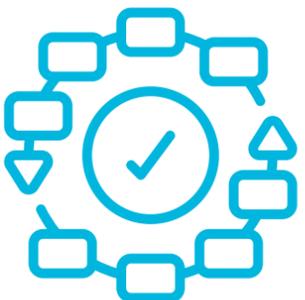
Part 1 Introduction to the Course.

Part 2 Introduction to openEHR discusses the openEHR concepts in general.

Part 3 Practical work focuses on practical work with the openEHR API. The practical work processes are applicable to any openEHR environment. The practical examples have however been prepared using the development tools from a company called Better.

Part 4 Building a model-based user interface is about utilizing Better Studio Form Builder in building applications.

Part 5 Takeaways from the Course.



openEHR Technical Course



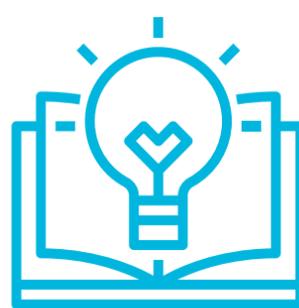
Part 1: Introduction to the Course

The content of the Course and the facilitator are introduced in this part. You will also find general instructions regarding navigation and active participation in the Cloned eLearning environment.

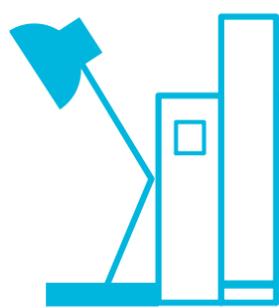
Part 2: Introduction to openEHR

In this part you will learn about the openEHR technical core elements and their role in the openEHR architecture. These elements are the building blocks for a system, where data are separated from the applications.

You will also be guided how to find the archetypes in the Clinical Knowledge Manager and how to inspect and download them in order to be able to design your own data model with the Archetype Designer tool.



This part consists of the following modules.

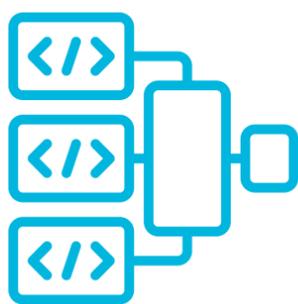
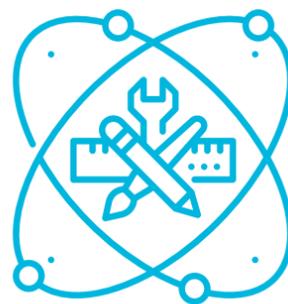


openEHR basics

In this module you will get to know what openEHR is and what it means to base your solution on a standardized open specification. We explain the basics of the underlying reference model that defines the data types and structures used throughout the system. Understanding the reference model and how openEHR defines the information model is paramount to understanding the rest of the openEHR-related approach.

How does it all work?

In this module we compare openEHR to the traditional approach for applications to work with data and to address classical technical challenges - such as data model governance and interoperability. This module will tell you about the openEHR core elements, their role and their interaction with each other. In the openEHR environment data input, storage and output are clearly separated. We will also explain the benefits of this approach.



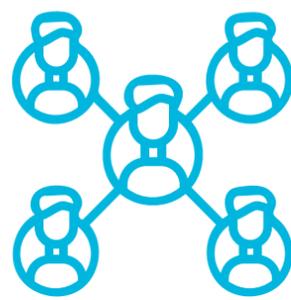
Archetypes

This module tells you about Archetypes. They belong to the openEHR core elements and define the possible care-related clinical content that you can incorporate in the solution. You will learn about archetypes - what they are, where you can find them, how they are designed and maintained and why they are important.

openEHR Technical Course

Templates and data model design

This module explains how templates are built by bundling archetypes into a data model and how they can be fine-tuned to your needs. Templates represent your data model for specific use cases. In this module we will present the modelling process in practice and will design a simple template, simultaneously explaining the tools and other necessary details along the way.



Compositions

This module explains how data are stored in the openEHR based repository and how the clinical data documents (compositions) look like. We will present the data validation and composition versioning process. You will also learn about the internal data structure and its meaning.

Archetype Query Language

This module will discuss Archetype Query Language (AQL). AQL is the most commonly used tool for retrieving data from an openEHR repository. We will compare AQL to SQL and explain the differences in syntax and usage. You will get to understand the basic principles when querying for structured data and get hints to avoid the pitfalls.



Part 3: Practical work with the openEHR API / typical workflows

In this part you will get to know the openEHR API and start working with the data. We will show how to manage EHRs, how to upload the template we have designed, how to store and update a composition, and how to query the data. Furthermore, we will create a demographic entry for a patient, search for patients and create a patient entry using an HL7 ADT message. Finally, we will manage and query the terminology on a local terminology server and export the clinical data using the ETL module.



This part consists of the following modules.

openEHR API - interacting with the repository

In this module you will be introduced to the main repository API endpoints that allow us to manage templates, EHRs and their folders and compositions. We will use them to create or retrieve different resources. We will also show practical examples addressing e.g. the following topics:

- how to create a new EHR,
- how to retrieve a specific composition,
- how to execute a query.



openEHR Technical Course

Demographic service



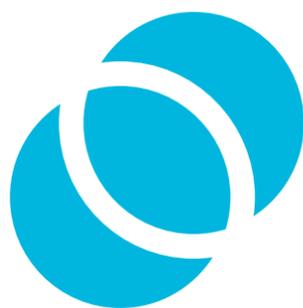
In this module you will learn about the demographic service. You will learn that the openEHR repository focuses on clinical data and addresses privacy concerns by design. Accordingly, it does not contain any demographic or administrative data. These data are typically handled in parallel and offered through different interfaces - HL7 ADT messages, the FHIR API or similar. The demographic service can substitute the master patient index and in this module you will learn how to use it at its best.

Terminology server - handling external value sets

In the openEHR environment terminologies are managed with value-sets that provide shared information to be included in clinical documents. They can be local or provided by external data sources - terminology servers. In this part we will have a closer look into typical terminology server capabilities and will use it when querying for data.



ETL- Data export



In this module open data export is discussed. The EHR repository is flexible by design, separating the database model from the information model. When large amounts of data are needed for research, trials or business intelligence purposes, a separate relational data warehouse may be preferred. The ETL process is used to extract and transform openEHR into a relational model of your own choice.

Part 4: Building a model-based User Interface

The traditional development model starts by building a form for collecting the data. In the openEHR development we start with the model and build a form on top of it. In this part you will get to understand how the openEHR-based data model helps you create a modern interface with low code tools.



This part consists of the following modules.

Form Builder basics - building a User Interface



openEHR templates contain all the necessary information for building data collecting forms based on their content. In this module we will explore the relationship between the data model and user interface, use generic form elements to enrich the form, play with layouting options and get familiar with Form Builder tools basics.

openEHR Technical Course

Form Builder - smart forms

When building forms with complex content, you can benefit a lot from the ability to add logic to the form and to respond to the data being entered. Dependencies and scripting will be presented, together with the concept of variables, to enable data sharing and responding to changes. We can access the data from the repository with AQL and present it on forms as well. In this module we will teach you how to use the Form Builder Tool for creating smart forms.



Form Builder - working with external data



A typical healthcare institution will have data scattered across multiple data servers and services. Most systems expose data via different APIs, the REST API being the most common. Forms can interact with APIs in order to gather data or send it somewhere else.

In this module we will interact with an API service to gather content, use terminology server to populate a form and present data in tables.

Form Renderer - deploy your own forms

Form Renderer is the library that renders the forms designed with Form Builder on the target device. In this module you will learn how to embed a form into any web application, configure it how you like, interact with it and store the gathered data in the openEHR repository.



Part 5: Takeaways from the training

This concludes the most important takeaways from the training. As mentioned, you can also get a certificate of completion!

