

A diagram of a software application

Description automatically generated

A diagram of a computer program

Description automatically generated

Yellow section - for read only apps

For transactional also add the blue also

Welcome back!

In this video tutorial, I will discuss with you about the architectural overview of Abap RESTful Application

Programming model.

Let's start.

The Abap RESTful application programming model consists of three main layers.

Data modeling and behavior.

Business service provisioning and service consumption.

The Data Modeling and Behavioral Layer contains domain specific business objects that are defined with

CDs, views, and transactional behavior.

The Business Service provisioning layer consists of projection views with their projection behavior

to focus on a particular aspect of data model, which is exposed as a business service through the data

protocol.

Service consumption layer allows you to consume all types of OData services as well as OData fig API.

Now let's investigate how the development flow looks in more detail.

The database layer is defined by the definition of dictionary tables.

And are the basis of any data model.

The core data services are the CDS views on the top of dictionary tables.

In the CDS layer, the semantic data model is defined by CDS views, and it can be used to manipulate

data that is persisted in the database table.

Behavior definition is a crucial aspect of any CDS data model, as it defines the action that can be

performed on the data, such as creating, updating, and deleting the records.

The behavior implementation provides the implementation of a behavior.

This can be achieved through two different approaches managed and unmanaged.

In case of managed approach.

Framework automatically implements operations such as create, update and delete.

In case of unmanaged approach, developers must implement all the behavior.

This involves coding for the necessary operations, including create, update and delete.

Projection data model is a subset of fields of the underlying data model that are relevant for the application.

UI annotations would be the part of projection view.

The projection behavior definition provides the behavior for the projection CDS view.

It is a subset of operations defined for the underlying data model.

Service definition describes which CDS entities of a data model are to be exposed as a business service.

Service bindings allows you to bind service definitions to the client server communication protocol,

such as OData.

The service bindings is used to start the SAP Fiori Element application for you, which makes the application

visible on the UI.

To develop a read only application, you need to carry out the steps containing this highlighted result.

To develop a full transactional application with steps in this, highlighted regions are required in

the edition.

You are able to change the data consistency of the existing instance of entity by adding validations.

Action can be used as a part of the business logic to execute non-standard operations such as status

change.

To develop additional custom logic such as validations and actions.

The step in the highlighted regions are need to be carried out.

The main intention of this video lecture is to make you familiar with the different terminologies in

the RESTful application programming model.

If you didn't understand any point at this moment, don't worry, because in the upcoming videos, we

will create a Fury application using RESTful application programming model, and we will follow the

architectural steps that has given in this diagram.

So that will help you understand each step in detail.

So that's it for now.

See you in next video lecture.