COMP9322
Software Service Design and Engineering
S1 - 2018
Week 7 – Exploring A Basic Solution for Assignment 2

Case Study in Capsifi Jalapeno Framework

To document and manage different data sources, data sets, data processing programs and services, the organization decides to leverage a business and IT architecture modelling tool. It captures relevant aspects of the IT infrastructure. Clients can explore and use existing infrastructure modelled in the tool to create analytic process for house price prediction.

We identify 4 main resource types to be modelled:

1. Objects that represent information shared between different services: Data Sources, Dataset Type, Data Set, Data Processing Program.
2. Services available for operations: Import Dataset, Execute Model, Download Dataset, Visualize Dataset.
3. Graphical user interface templates that facilitate users to define particular inputs for import, execute and export.
4. End-to-end process that leverage services, interface templates and information objects to conduct prediction.

Let's explore how these resources can be modelled in Capsifi business architecture modelling tool available at: https://unsw.capsifi.com/. Each team will be given a username and password to access this.

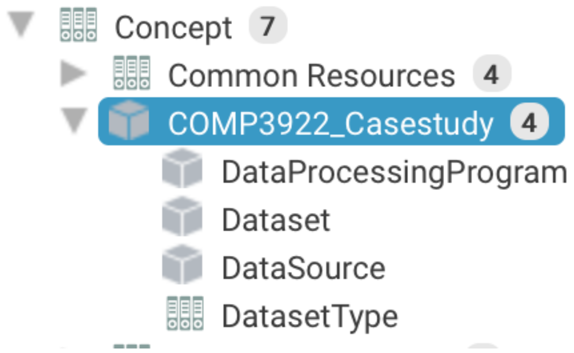
You can use it to design and model your own information objects, services, interactions and digital interactions.
Use the "SELECT A MODEL" dropdown list provided in the home page to switch between models.
Example of the case study is implemented in the Demo model. You can inspect and observe it.

Each team can access an empty model named as "UNSW_Teaching<teamNumber>" through "SELECT A MODEL" dropdown list. You can use that to design and implement the case study with your information objects, services, GUIs and processes.

Note: It is recommended to use Google Chrome browser.

1. Modelling Information Objects

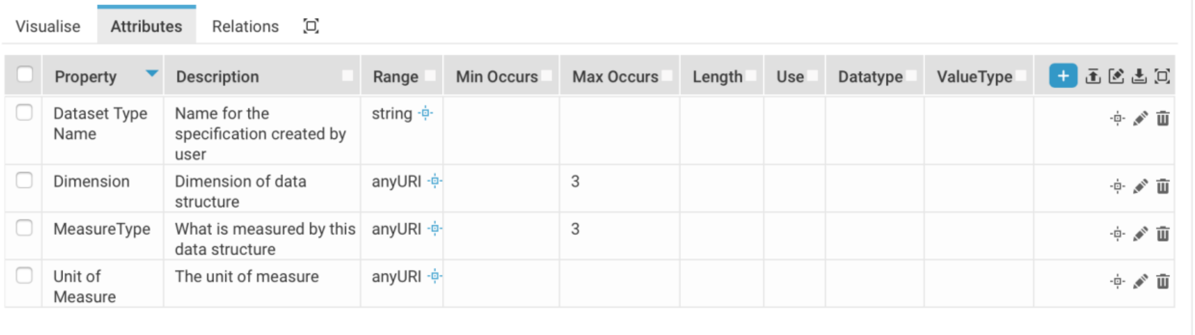
To model information objects, we can use Capsifi Information Modelling Cell. Navigate to Information Modelling -> Get started -> Manage Concepts.
You can select or create a root and model your information concepts under that.
For this example, information objects are created in “Demo” model, under root “Concept/ COMP3922_Casestudy”, as shown in the image below.



- Concept 7
  - Common Resources 4
  - COMP3922_Casestudy 4
    - DataProcessingProgram
    - Dataset
    - DataSource
    - DatasetType

We give a name, description and a type for each concept.

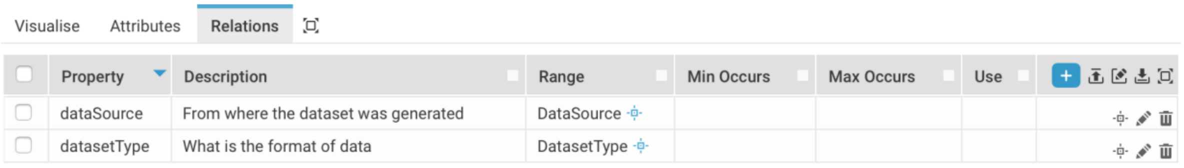
Further, you can add different attributes to these concepts when you scroll down the “Manage Concept” tab and select “Attributes” tab. There you can add different properties to your concept, with property name, description, range etc. Shown below is the set of properties for “DatasetType” example concept, which are useful to define what is measured by a data set and in what dimensions.



☐	Property	Description	Range	Min Occurs	Max Occurs	Length	Use	Datatype	ValueType
☐	Dataset Type Name	Name for the specification created by user	string						
☐	Dimension	Dimension of data structure	anyURI		3				
☐	MeasureType	What is measured by this data structure	anyURI		3				
☐	Unit of Measure	The unit of measure	anyURI						

You can select “Relations” tab next to “Attributes” tab and represent relationships between different concepts.

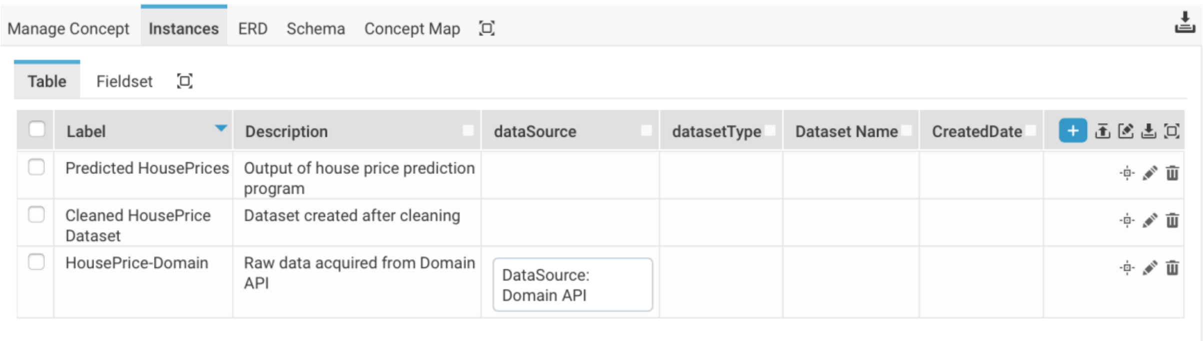
For example look at the figure below where a “Dataset” is defined to have a relation to “DataSource” and “DatasetType”, to represent the origin of Data set and required format.



☐	Property	Description	Range	Min Occurs	Max Occurs	Use
☐	dataSource	From where the dataset was generated	DataSource			
☐	datasetType	What is the format of data	DatasetType			

Similarly add other necessary attributes and relations for the concepts: Dataset, Dataset Type, DataSource, Data Processing Program.

You can use the “Instances” tab to enter instances of each concept such as the available data sets as shown below.



□	Label	Description	dataSource	datasetType	Dataset Name	CreatedDate	+
□	Predicted HousePrices	Output of house price prediction program					
□	Cleaned HousePrice Dataset	Dataset created after cleaning					
□	HousePrice-Domain	Raw data acquired from Domain API	DataSource: Domain API				

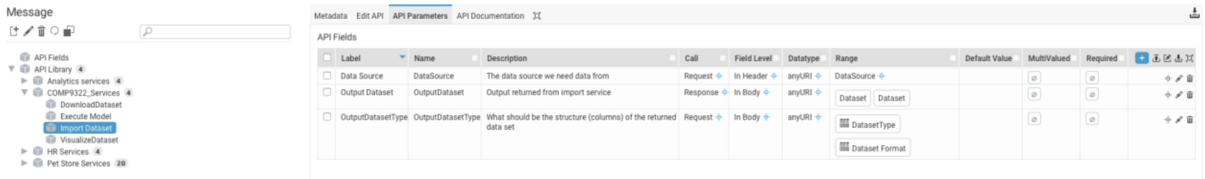
2. Modelling Services

Now let’s see how to model import, execute and export services within Capsifi framework. On the home page, navigate to Digital Interactions -> Get Started -> API Management. Under the root “API Library” you can create a parent (host) under which all your services can be modelled.
Please note that this API Management interface supports REST API modelling and document generation in OpenAPI (Swagger) format from the services/APIs you model. So, all the services should be organized under the root – API Host, which has the same URL prefix.

In the example provided in Demo model, we have created a host called “COMP9322_Services”, with type “API_Host”. Under that there are four services: DownloadDataset, Execute Model, Import Dataset and VisualizeDataset. The type of all services should be “Method”.

Use the “Edit API” tab to enter URL, version information etc. Each service can have service type defined (GET/PUT/POST/DELETE)
You can define parameters for each service in the “API Parameters” tab.

For example, “Import Dataset” Service accepts two input parameters – Data Source and Dataset Type defined as information objects and returns Output Dataset.



API Fields

□	Label	Name	Description	Call	Field Level	Datatype	Range	Default Value	MultiValued	Required
□	Data Source	DataSource	The data source we need data from	Request	In Header	anyURI	DataSource			
□	Output Dataset	OutputDataset	Output returned from import service	Response	In Body	anyURI	Dataset Dataset			
□	OutputDatasetType	OutputDatasetType	What should be the structure (columns) of the returned data set	Request	In Body	anyURI	DatasetType Dataset Format			

Note how you can differentiate input and output parameters for the service vial “Call” column and how “Range” column can be used to link information objects into parameters.
In Capsifi framework “anyURI” data type represents an information object defined under the Concept model.

Explore other services and their parameters.

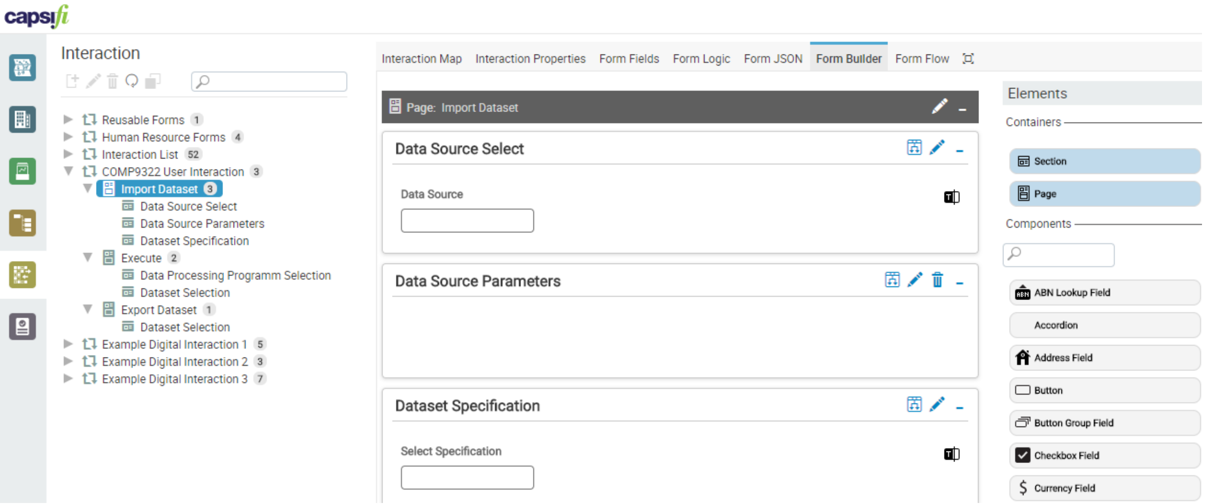
If you go to the “API Documentation” tab you can see the documentation of your service on Swagger (Open API) JSON format. You can copy this JSON and use it within any system compatible with swagger for visualization, document generation and code generation. For example, you can visualize your service documentation at https://editor.swagger.io/ by pasting generated documentation there.

3. Modelling Interfaces

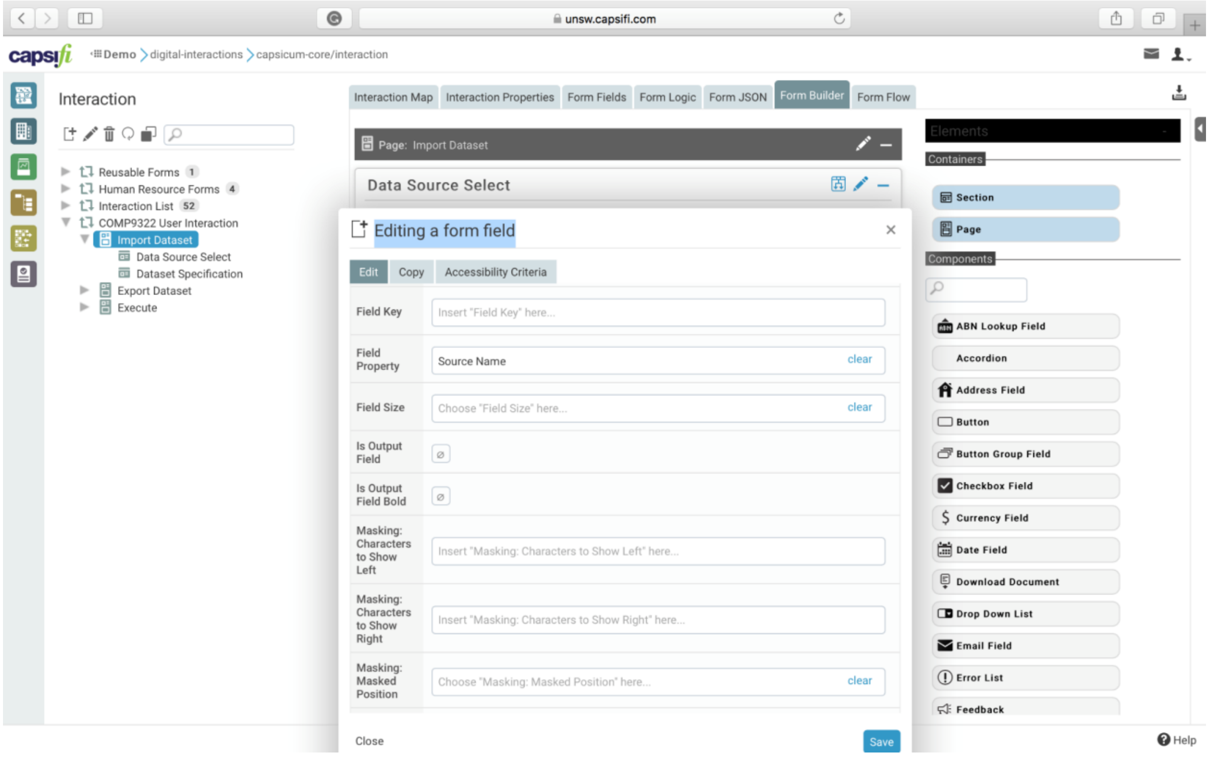
Once we define the information models and service models, we need to create user interfaces that enable users to communicate with services via information objects. For example, an interface should be defined to input desired data source and a dataset type, which can be linked to Import Dataset service to retrieve respective dataset.
Capsifi framework supports this via Interaction modelling interface. From the home page, navigate to “Digital Interactions -> Getting started ->Digital Wireframe Modelling”.

Example set of interfaces can be found at the root “COMP9322 User Interaction”. Capsifi interfaces are organized as set of Forms which contain pages. Each page can have different section where input options can be entered as components.

Observe three pages we have designed: “Import Dataset”, “Export Dataset” and “Execute” including their sections and components. You can access a visual of a selected form on “Form Builder” tab. Given below is the definition for Import Dataset form.



In your own model you can add new components by drag-and-drop from the right-hand pane (please note that this feature is available on Google Chrome browser only).
Click the “Edit” button (pencil) to add label, description etc. to the selected component.
As these forms are designed to manipulate information objects we modelled before, you can link each component into a property of an information object. When you click edit (e.g. to edit Data Source) and scroll down, the “Field Property” drop down list can be used to select appropriate property as shown below. Here the form field “Data Source” is linked to the Field Property “Source Name” we defined in “DataSource” concept.



Editing a form field

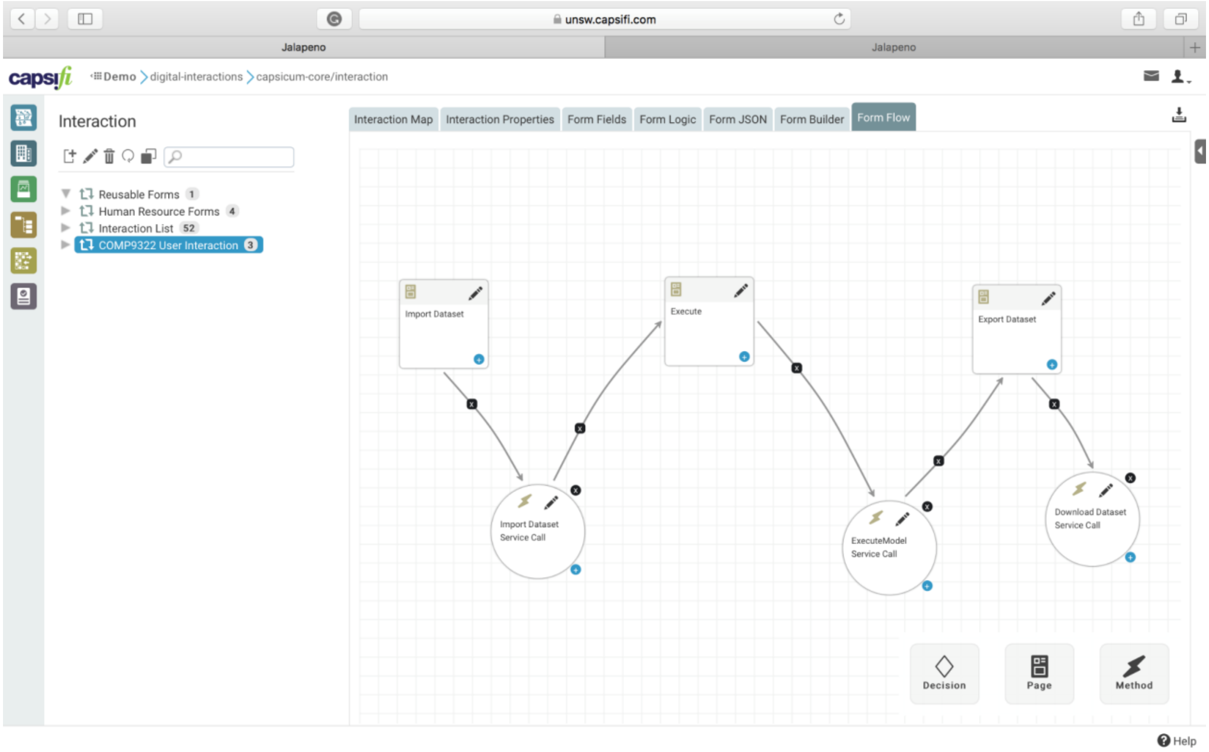
Field Key	Insert "Field Key" here...
Field Property	Source Name clear
Field Size	Choose "Field Size" here... clear
Is Output Field	⌀
Is Output Field Bold	⌀
Masking: Characters to Show Left	Insert "Masking: Characters to Show Left" here...
Masking: Characters to Show Right	Insert "Masking: Characters to Show Right" here...
Masking: Masked Position	Choose "Masking: Masked Position" here... clear

Observe and identify how other forms are linked to information objects via Field Property.

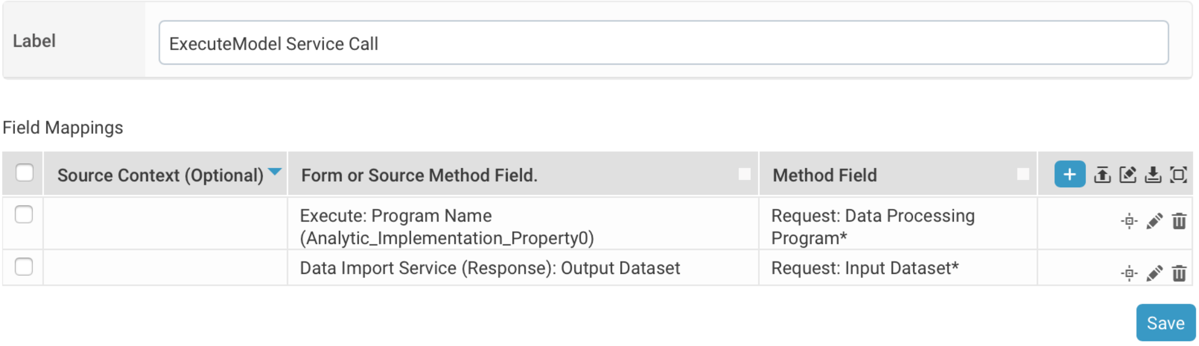
4. Modelling the Process Instances

Once we defined necessary services and interfaces, Capsifi Interaction Modelling enable us to link them together, defining decision logics and information flows among them.
From the home page, navigate to “Digital Interactions -> Getting started -> Digital Wireframe Modelling” and select “Form Flow” tab.

In the Demo model, if you select “COMP3922 User Interaction” you can observe how flow is defined between the GUIs we defined and the services.



If you click Edit icon on each service call, you can see how information coming from Forms and previous service calls are chained into the inputs of respective service.
For example, the method fields of ExecuteModel Service Call are matched to the Program Name field in “Data Processing Program” form and Output Dataset of Import Dataset Service Call.



Label	ExecuteModel Service Call

Field Mappings

□	Source Context (Optional)	Form or Source Method Field.	Method Field	+
□		Execute: Program Name (Analytic_Implementation_Property0)	Request: Data Processing Program*	
□		Data Import Service (Response): Output Dataset	Request: Input Dataset*	

Save

Note: In your GUI model you can add new pages, methods (services) and decision logic to this form flow by drag-and-drop the three icons shown in bottom right corner.