TU Delft Delft University of Technology

FACULTY OF TECHNOLOGY, POLICY AND MANAGEMENT
ENGINEERING SYSTEM AND SERVICES SECTION
ICT TRACK

TB141IC – ICT System Engineering and Rapid Prototyping MVC in Mendix - Exercises Handout

Jacopo De Stefani - Marijn Janssen

February 12, 2024

Introduction

This document contains the proposed exercises for the practical sessions. Each subsection covers a different practical session.

Exercises

Practical 1 - MVC in Mendix - View: Pages

The first session focuses on exploring Mendix and implementing the View part of the Model-View-Controller software architecture in Mendix.

Exercise 1

Register/Login to Mendix and explore the platform. Create a new app from a blank template and explore the Mendix platform.

Can you describe what the role of the following elements are?

- Project Buzz
- Stories
- Team
- Feedback
- Documents
- Team Server
- General Settings

Exercise 2

In the app created in **Exercise 1**, create a form to support the loading of the data from **User Story 1**:

`Businesses` can register on the platform using a form asking at least for the name of their business, their address, their VAT number and an official telephone number.

In order to do so, you need to create a new page, and create the corresponding domain model, through the dedicated Mendix features. Finally, you can proceed to link the form with the domain model.

Exercise 3

Adapt the application so that the form created in **Exercise 2**, can be accessible from the homepage.

In order to do so, you need to add a button or a card widget and link it to the previously created page. Add the check mark on Create New Object to solve the consistency error.

Exercise 4

With the newly found knowledge of **Exercise 2** and **Exercise 3**, create a new page showing an overview of Business Data.

In order to do so, you need to create a new page, select a list template, and adapt it to display the relevant information.

Exercise 5

Finally, add a direct link to the page created in **Exercise 4** on the home page and adapt the navigation bar to have a quick access to all the created pages.

Practical 2 - MVC in Mendix - Model: Domain Model

The second session reviewes the concepts related to the View and introduces the Model part of the Model-View-Controller software architecture in Mendix. The second session focuses on how to link View and Model, and how to create a Domain model involving different relationships between the entities.

Exercise 6

Create a new Mendix app and define the domain model for an application managing training events. A training event needs to have a connection to a course, location, and trainer. This way, you can store in the system what course will be given, where it will be hosted, and who will teach it. A training event will always be for only one course, at one location, and hosted by one teacher. However, over time, a course will be scheduled more than once, locations will be used more than once, and teachers will host several training events. A course is characterized by a title, a duration a price and a description, a location by its name and address and a trainer by its name and e-mail address.

Practical 3 - MVC in Mendix - Controller: Microflows and Data Validation

The third session focuses on discovering the notation of Microflow and understanding how to perform data validation in Mendix at different levels.

Exercise 7

In the app defined in **Exercise 1**, create two new entities: `Product`, having as attributes: a name, a description and a price and `ProductStock` having as attributes: a `Business`, a `Product` and the quantity of products left in stock.

After that, introduce the following constraints on the **Domain Model** of the application.

Entity	Attribute	Validation Rule
Business	Name	Required, Unique
	Address	Required
	VatNumber	Required, <= 50 characters
	PhoneNumber	Required, <= 50 characters
	EmailAddress	Required, Email format
Product	Name	Required, Unique
	Price	Required, >= 0
	Description	Required
ProductStock	StartDate	Required
	Business selection	Required
	Product selection	Required
	Quantity	Required,>= 0

The constraints need to be introduced via **Mendix Studio Pro**.

Exercise 8

Based on the Domain Model defined in **Exercise 7**, employ a regular expression to enforce a validation rule on the EmailAddress fields, via the **Domain Model** of the application.

The constraints need to be introduced via **Mendix Studio Pro**.

Tip: *You don't have to know how to write Regular Expressions. Just use your favorite search engine and look for Regex and then the thing you want to validate.*

Exercise 9

Based on the app defined in **Exercise 7**, create a form to be able to edit a `ProductStock` object.

Then, create a microflow connected to the Save button named **ACT_ProductStock_Save**. The microflow will need to:

1. Commit the object to the database
2. Close the page

Exercise 10

Update on the microflow defined in **Exercise 9**, to perform a validation on a `ProductStock` object. The microflow will need to:

1. Check that the `Quantity` is not `empty`

Exercise 11

Update on the microflow defined in **Exercise 10**, to perform additional validations on a `ProductStock` object. The microflow will need to:

1. Check that the `Business` attribute is not `empty`
2. Check that the `Product` attribute is not `empty`

Exercise 12

Create pages to add new `Products`, `Businesses` and `ProductStocks` in order to test the functioning of the validation processes.

Practical 4 - MVC in Mendix: Controller - Microflows and Computed Attributes

The fourth session focuses on implementing Microflow to perform different actions and automatic computations.

Exercise 13

In the app defined in **Exercise 7**, add two attributes: `SaleDuration` (in days) and `EndDate` to the `ProductStock` object.

Then, create a microflow to perform an automatic computation of the `EndDate` based on `StartDate` and `SaleDuration`.

Exercise 14

In the app defined in **Exercise 13**, add an attribute: `Subtotal` (in Euro) to the `ProductStock` object.

Then, create a microflow to perform an automatic computation of the `SubTotal` based on `Quantity` and `Price`.

A retrieve operation might be required to solve the exercise.

Exercise 15

In the app defined in **Exercise 14**, add an attribute: `ExpectedSales` (in days) to the `Business` object.

Then, create a microflow to perform an automatic update of the `ExpectedSales` every time a new `ProductStock` object is created.

Practical 5 - Mendix Advanced Aspects - Security, Mapping and Dashboarding

The fourth session focuses on implementing security aspects in Mendix (namely access control) as well as implementing more advanced functionalities using Microflows and Mendix built-in capabilities.

Exercise 16

Create a new application to manage Students and implement security at the Domain Model level.

In order to do so, you need to:

1. Create a new app

2. Add a new entity `Student` to the domain model
3. Create the corresponding form and overview pages.
4. Enable security.
5. Add a new **Domain Model Rule** to only allow **Read** access to the `User` Module role.

Test the application to see what the implication of your access control rule are. The constraints need to be introduced via **Mendix Studio Pro**.

Exercise 17

Based on the application developed in **Exercise 16**, introduce a conditional visualization rule on the pages that are made inaccessible based on the previously introduced rules.

The constraints need to be introduced via **Mendix Studio Pro**.

Exercise 18

Based on the application defined in **Exercise 13**, add the visualization the localization of the business on a map, reachable from the shop overview page.

In order to do so, you need to:

1. Create a new page
2. Add a map widget to the page
3. Update the Domain Model in order to include the required attributes in the `Business` entity
4. Map the newly created attributes onto the map widget.

Test the application to verify the correct visualization on the map.

Exercise 19

Based on the application defined in **Exercise 13**, add the visualization of the sales of the application over time, reachable from the shop overview page.

In order to do so, you need to:

1. Create a new page
2. Add a graph widget to the page
3. Update the Domain Model in order to add a new `Sale` entity including a quantity and a sale date, and relationship to `Business` and `Product` entities
4. Create a microflow to retrieve all the sales related to the considered shop
5. Configure the graph widget in order to include a **Series**, based on the aforementioned microflow, and to use sales date as **X-Axis attribute** and quantity as a **Y-Axis attribute**.

Test the application to verify the correct visualization of the graph.

Licensing