
CaseManagement Documentation

CaseManagement

Sep 01, 2021

1	Glossary	3
2	Architecture overview	5
2.1	Domain model	6
2.2	Clustering model	6
3	How to launch the solution ?	7
4	Publish case plan	9
4.1	Add a case file	10
4.2	Assign permissions	12
4.3	Save and publish case file	13
5	Launch a case clan instance	15
6	View case plan instance	17
7	CMMN 1.1	19
7.1	Task	19
7.1.1	Human task	19
7.1.2	Process task	19
7.2	SEntries	20
7.3	Milestone	21
7.4	Decorators	21
7.4.1	Manual activation rule	21
7.4.2	Repetition rule	21

CaseManagement is an open source framework enabling any DOTNETCORE solutions to host a CMMN engine. It supports most of the concepts introduced by the Case Management Model And Notation (CMMN) standard version 1.1 for examples :

- 1) Human task : Task that is performed by a Case worker.
- 2) Automatic task : Can be used in the Case to call a Business process.
- 3) Sentry : combination of an “event and/or condition”.
- 4) Milestone : represents an achievable target.
- 5) Case file : represent a piece of information of any nature, ranging from unstructured to structured and from simple to complex, which information can be defined based on any information modeling language.

Github: <https://github.com/simpleidserver/SimpleIdServer>

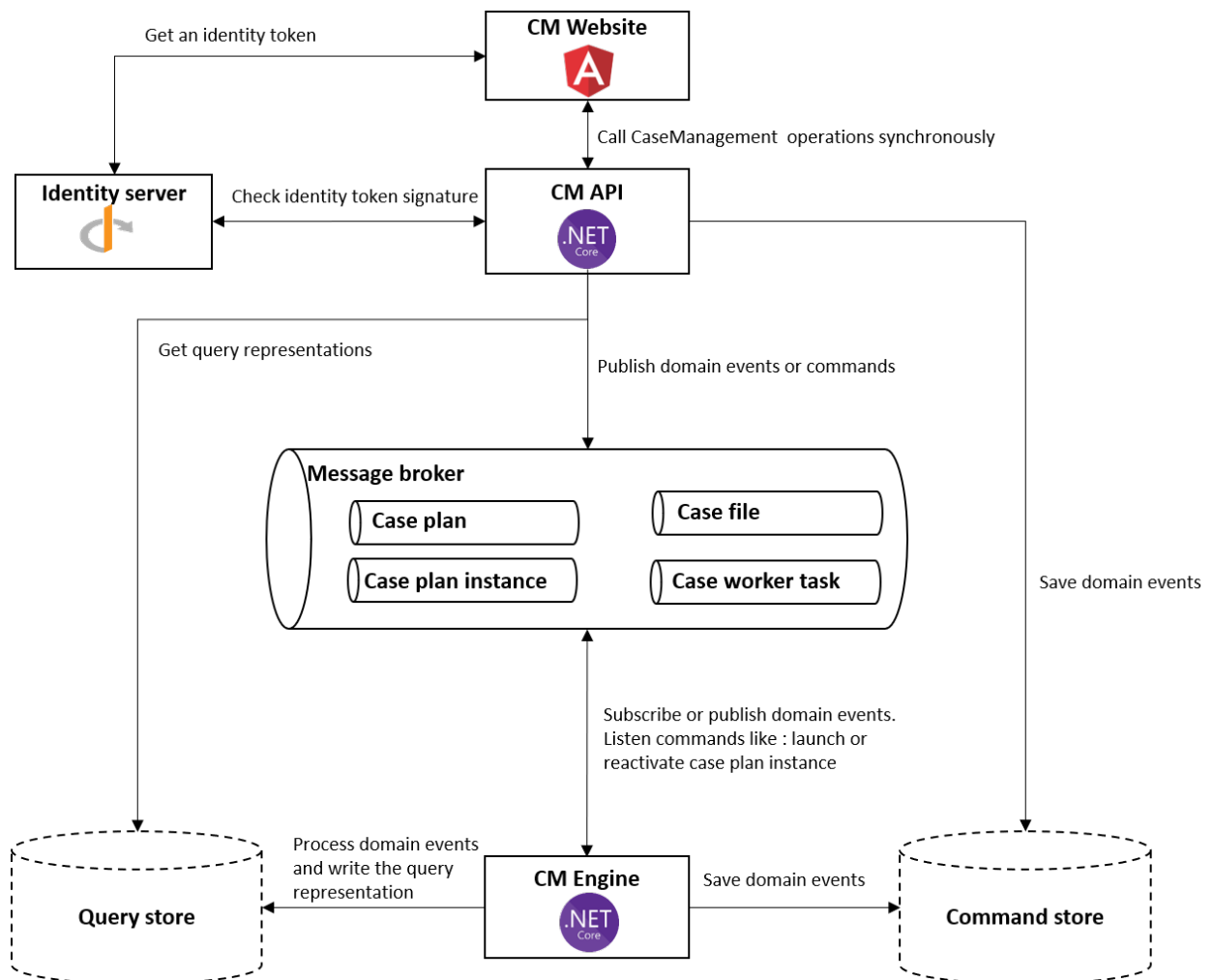
Nuget feed: <https://www.myget.org/F/advance-ict/api/v3/index.json>

CHAPTER 1

Glossary

Term	Definition
Case plan	Involves actions taken regarding a subject in a particular situation to achieve a desired outcome
Case plan instance	Instance of a case plan launched by a user or by an external event
Milestone	Represents an achievable target
Task	Atomic unit of work. An action is deciding what must be done, a task is what must be done
SEntries	A SEntry is a combination of an event and / or condition. When the event is received, a condition might be applied to evaluated whether the event has effect or not.

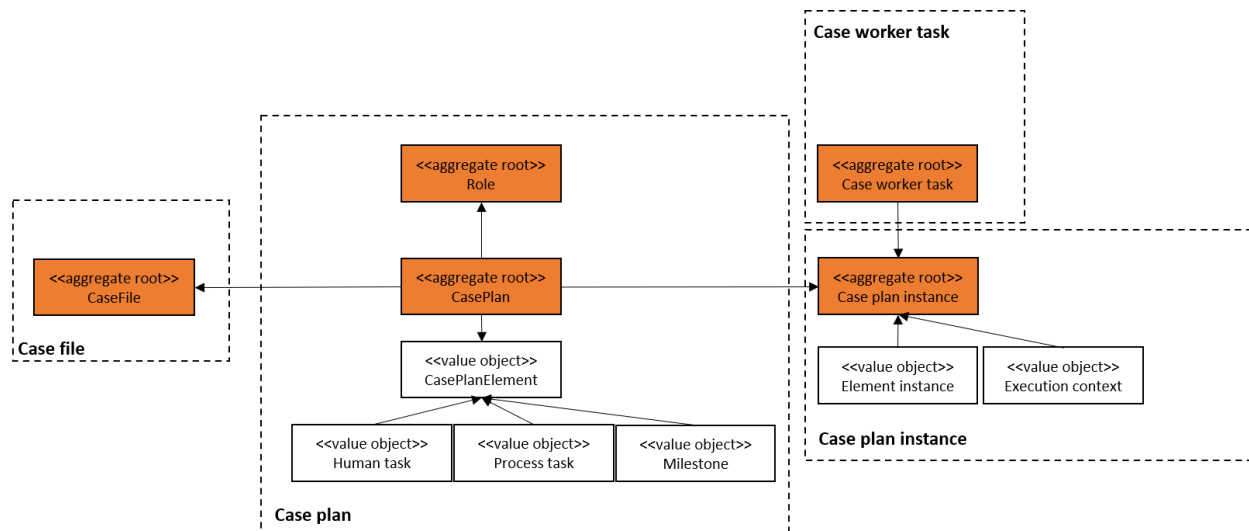
Architecture overview



- **CaseManagement (CM) Website** : the website is developed with angular. It is used by **case workers** or **business analysts** to manage the lifecycle of cases for example : manually activate a task or edit the definition of case plans.
- **Identity server** : OPENID server used by the solution to authenticate the end-user and limit the access to certain API operations based on the claims of the end user, for example : only a user with the role **businessanalyst** can update a case plan.
- **CaseManagement (CM) API** : REST.API service which exposes operations to interact with the engine for example : update the case plan or launch a case plan instance.
- **CaseManagement Engine** : Execute a case plan instance.

2.1 Domain model

The DDD aggregate pattern has been used to identify the different domain models present in the CaseManagement API.



2.2 Clustering model

The case engine can be distributed to different nodes in a cluster. Each case engine must then connect to a shared database.

How to launch the solution ?

To launch the entire solution follow the steps below :

1. Fetch the sources from [Github](#).
2. Open a command prompt and navigate to the directory.
3. Launch the following command:

```
launch.cmd
```

One console application is launched for each project.

The case management website is available at this URL (<http://localhost:8080>). Only users with **businessanalyst** or **caseworker** role can access to the website. By default there is one businessanalyst and one caseworker account configured :

Property	Value
Login	businessanalyst
Password	password

Property	Value
Login	caseworker
Password	password

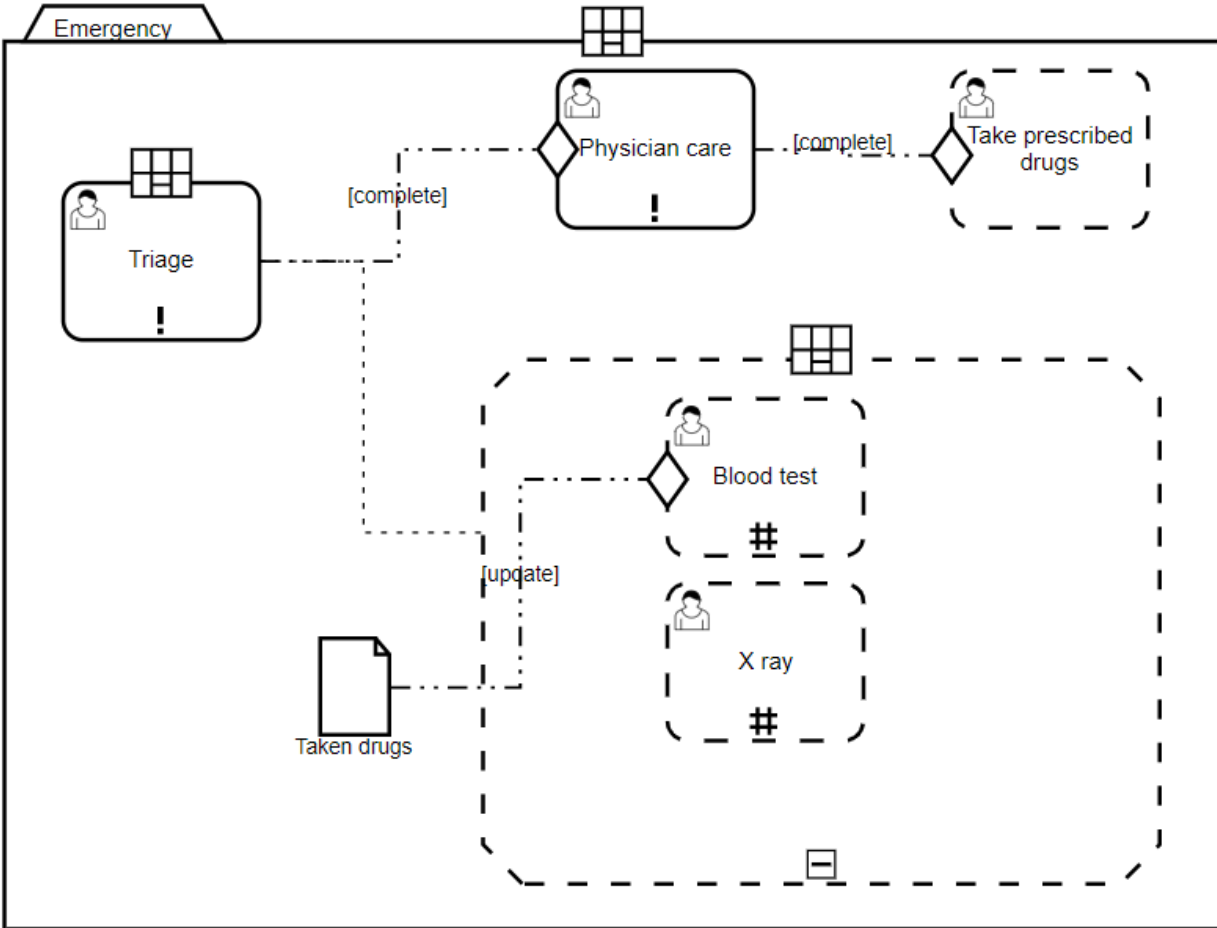
Publish case plan

As a **business analyst** you can add and edit **case files**. Each case file contains one or more **case plan**, it can be expressed and modeled with Case Management Model And Notation (CMMN) version 1.1.

A **case plan** describes the process that involves actions taken regarding a **subject** in a particular **situation** to achieve a desired outcome :

- The **subject** of a case may be a person, a legal action or a business transaction.
- The **situation** commonly includes data that inform and drive the actions taken in a case.

For example in an emergency, patients must be sorted, those who can benefit from an immediate response and treatment at the highest level of care are identified, and those who can wait can receive advice or be directed to an appropriate service. In this case the subject is the **patients**, the situation is the **degree of emergency** and the desired outcome is the treatment of the patients sorted by the degree of emergency. The case is modeled below :

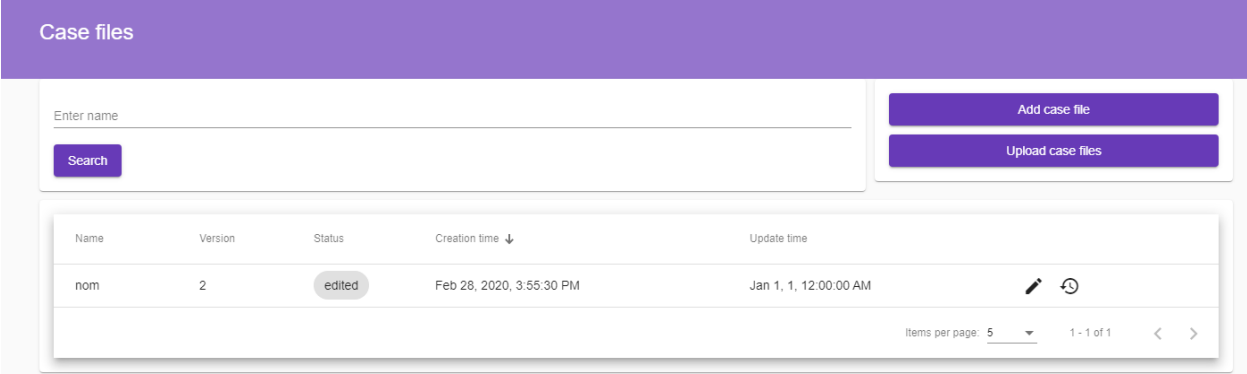


4.1 Add a case file

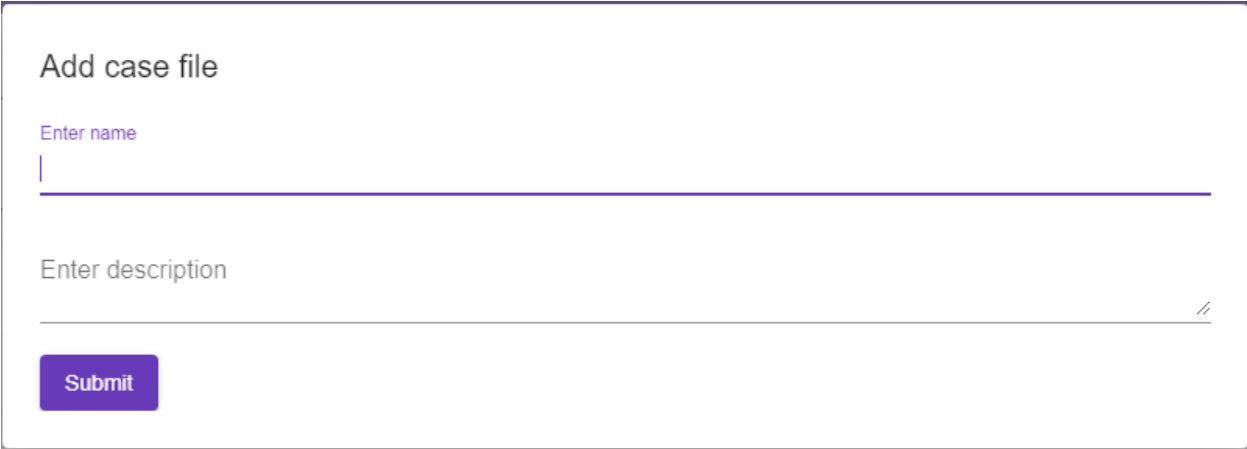
Browse the case management website, authenticate as a business analyst with the following credentials :

Property	Value
Login	businessanalyst
Password	password

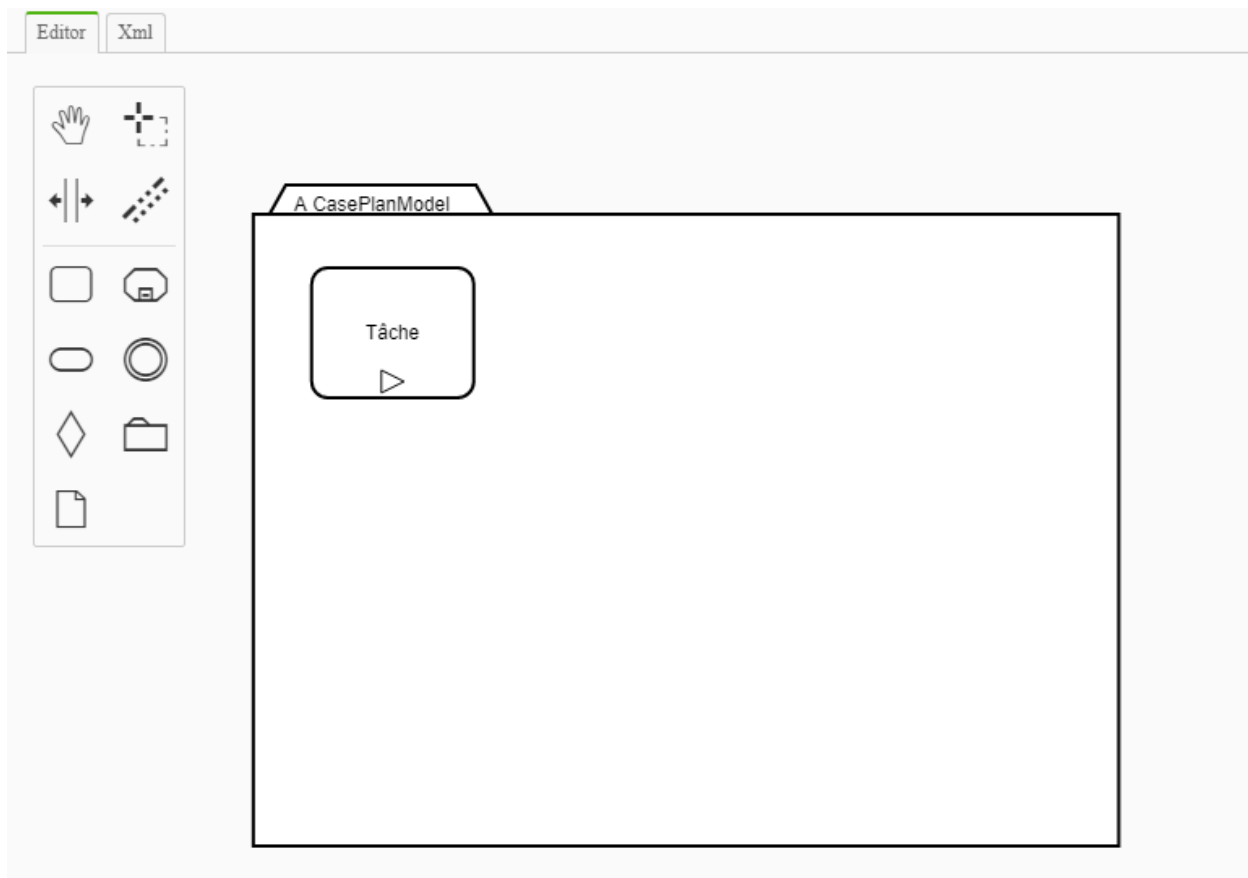
Under the **Cases** menu, click on **Case files**. The window displays the list of case files created by the authenticated business analyst.



Click on the button **Add case file**, fill in the form with a name and a description and submit it.



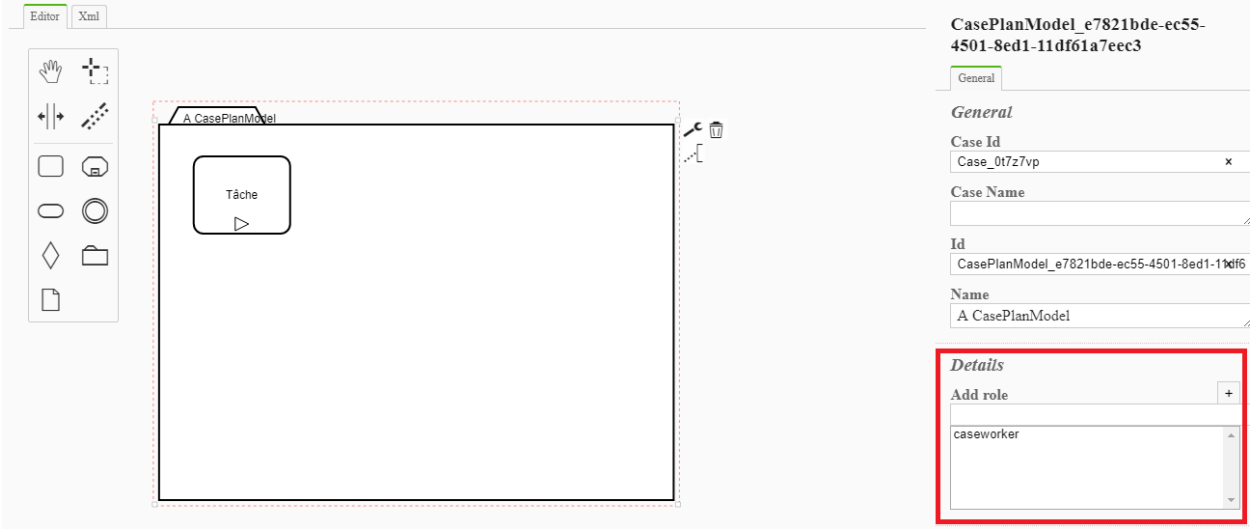
The user agent is redirected to a new window where the case file can be edited. By default the case file contains one case plan, its content can be edited by using the toolbar. For more information about the different elements, please refer to reference.



4.2 Assign permissions

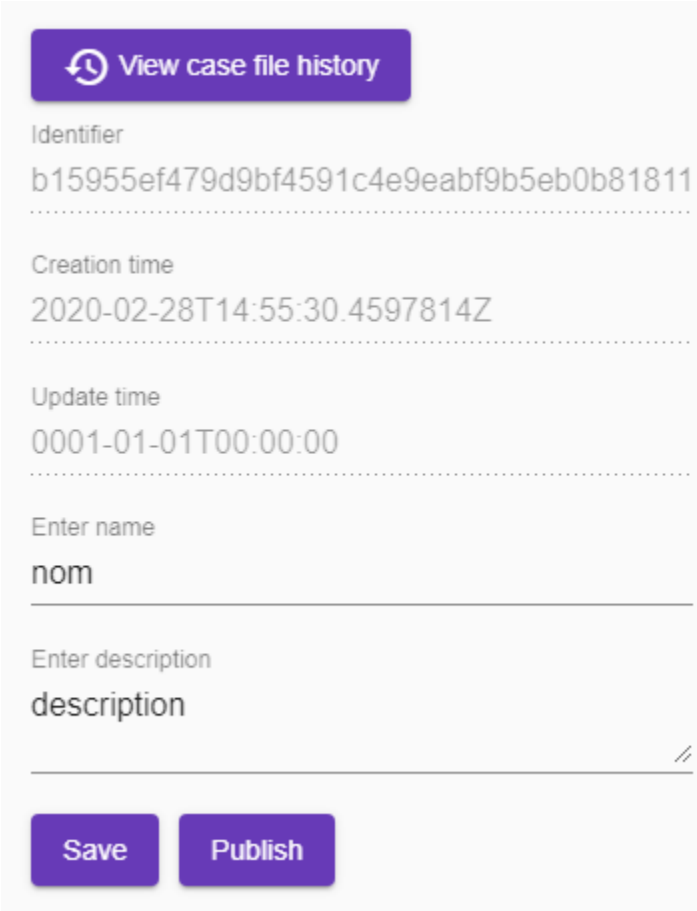
As a business analyst you can assign a list of roles to a case plan. Each role contains a list of case workers. At the moment the roles are not editable, they can be configured in the **Startup.cs** class of the **CaseManagement.CMMN.Host** project.

By default only one **caseworker** role with one **caseworker** user exists. This role can be assigned to the case plan by clicking on it and update the roles in the properties panel.



4.3 Save and publish case file




When the case file is complete, click on the button **Save** to save its content and click on **Publish** to publish the content. When the case file is published, its version is incremented and its case plans are published.



Launch a case plan instance

Always as a **business analyst**, click on the **Case plans** button under the **Cases** menu. The window displays the list of case plans created by the authenticated business analyst.

The screenshot shows the 'Case plans' interface. At the top, there is a search bar with the placeholder text 'Enter name'. Below the search bar is a 'Filters' dropdown menu and a 'Search' button. The main content area displays a table with the following data:

Name	Version	Creation time ↓	
A CasePlanModel	1	Feb 28, 2020, 3:55:30 PM	  

At the bottom right of the table, there is a pagination control showing 'Items per page: 5', '1 - 1 of 1', and navigation arrows.



Select a case plan to see its details, the window displays the following information :

- List of case worker tasks or forms which have been created during the execution of the case plan.
- List of case plan instances.

The screenshot shows the 'Case plan' details interface. The breadcrumb is 'Case plans \ Case plan'. On the left, there is a sidebar with the following information:

- Name: A CasePlanModel
- Creation time: Feb 28, 2020, 3:55:30 PM
- Case file: [View case file](#)

The main area contains two buttons: 'Launch instance' and 'View case plan instances'. Below these buttons is a tabbed interface with 'Case instances', 'Form instances', and 'Tasks'. The 'Case instances' tab is active, showing a table with the following data:

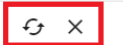
Identifier	State	Creation time ↓	
f6412ecc-7f33-4c8f-800c-5fd8b4a07c9	Completed	Feb 28, 2020, 3:55:36 PM	 

At the bottom right of the table, there is a pagination control showing 'Items per page: 5', '1 - 1 of 1', and navigation arrows.

CaseManagement Documentation

Click on the **Launch instance** button to create and launch a new case plan instance. After 4 seconds, the instance will be displayed in the corresponding table. Different actions are displayed next to the instance, they can be performed by the **business analyst**. A case plan instance has 6 different states and depending on them one or more actions can be performed.

State	Actions
Active	Suspend, Terminate
Completed	Re-activate, close
Terminated	Re-activate, close
Failed	Re-activate, close
Suspended	Re-activate, close

Identifier	State	Creation time ↓	
f6412ecc-7f33-4c8f-800c-5fd8b4a07cf9	Completed	Feb 28, 2020, 3:55:36 PM	

View case plan instance

Authenticate as a **case worker** with the following credentials :

Property	Value
Login	caseworker
Password	password

Under the **Cases** menu, click on the **Case plan instances** button. The window displays the list of case plan instances assigned to the authenticated case worker.

Identifier	State	Creation time ↓	
f6412ecc-7f33-4c8f-800c-5fd8b4a07cf9	Completed	Feb 28, 2020, 3:55:36 PM	👁
bd332a81-4880-4653-a270-6a081c7d5988	Completed	Feb 28, 2020, 3:55:04 PM	👁

Items per page: 5 1 - 2 of 2 < >

Select a case plan instance to see its details, the window displays the following information :

- List of active activities.
- List of enable activities.
- List of completed activities.
- The milestones.
- State of the case plan instance.

[Case plan instances](#) \ Case plan instance

A CasePlanModel (Feb 28, 2020, 3:55:36 PM) [View the case plan](#)

Completed

What happened so far?

Active activities

Enable activities

Completed activities

 Tâche

7.1 Task

7.1.1 Human task

A HumanTask is a Task that is performed by a Case worker.



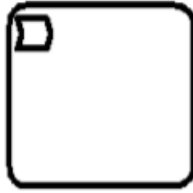
Parameters :

- **performerRef** : The performer of the HumanTask, a role must be specified for example : **caseworker**.
- **case:formKey** : Identifier of the form for example : **form**.

```
<cmmn:humanTask id="HumanTask_1c1771g" name="HumanTask" performerRef="caseworker" ↵  
↵case:formKey="form" />
```

7.1.2 Process task

A ProcessTask can be used in the Case to call a Business Process.



Parameters :

- **processRef** : A reference to a Process for example : **incrementtask**.
- **mappings** : Zero or more ParameterMapping objects. A ParameterMapping of a ProcessTask specifies how an input of the ProcessTask is mapped to an input of the called Process and how an output of the called Process is mapped to an output of the ProcessTask.
- **processRefExpression** : If processRefExpression is specified, it is assumed that the expression evaluates to a QName which is a valid QName of an existing Process.

```
<cmmn:processTask id="ProcessTask_1ci0zd0" name="Repetition Task" processRef=
↪"incrementtask">
  <cmmn:parameterMapping sourceRef="increment" targetRef="increment" />
</cmmn:processTask>
```

7.2 Sentries

Can be used by : Stage, Tasks, Milestone.

A Sentry “watches out” for important situations to occur (or “events”), which influence the further proceedings of a Case (and hence their name). A Sentry is a combination of an “event and/or condition.” When the event is received, a condition might be applied to evaluate whether the event has effect or not.



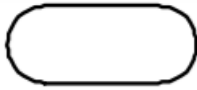
Parameters :

- **OnParts** : Defines the OnPart of the Sentry
 - **standardEvent** : Reference to a state transition in the lifecycle of a Stage, Task, EventListener or Milestone.
- **IfParts** : Defines the IfPart of the Sentry.
 - **condition** : A condition that is defined a Expression. The Expression MUST evaluate to boolean.

```
<cmmn:sentry id="Sentry_0f4my3g">
  <cmmn:planItemOnPart id="PlanItemOnPart_0tuf377" sourceRef="PlanItem_18sdiyp">
    <cmmn:standardEvent>complete</cmmn:standardEvent>
  </cmmn:planItemOnPart>
  <cmmn:ifPart>
    <cmmn:condition>context.GetNumberVariable("increment") == 2</cmmn:condition>
  </cmmn:ifPart>
</cmmn:sentry>
```


7.3 Milestone

A Milestone represents an achievable target, defined to enable evaluation of progress of the case. No work is directly associated with a Milestone, but completion of set of tasks or the availability of key derivables typically leads to achieving a Milestone.



```
<cmmn:milestone id="Milestone_0xqhnpg" name="milestone" />
```

7.4 Decorators

7.4.1 Manual activation rule

Can be used by : **Tasks** and **Stages**.



Specifies under which conditions **Tasks** and **Stages**, once enabled, start manually or automatically.

Parameters :

- **Condition** : A condition that is defined as an Expression. An expression that **MUST** evaluate to boolean. If the expression evaluates to **FALSE**, then the instance of the Task or Stage **MUST** be activated automatically when it is in stage Available, otherwise it **MUST** wait for manual activation.

```
<cmmn:manualActivationRule id="ManualActivationRule_0wcl9eu">
  <condition>true</condition>
</cmmn:manualActivationRule>
```

7.4.2 Repetition rule

Can be used by : **Tasks**, **Stages** and **Milestones**.



A RepetitionRule specifies under which conditions **Tasks**, **Stages** and **Milestones** will have repetitions. Each repetition is a new instance of it. The trigger of the repetition is SEntry, that is referenced as entry criterion, being satisfied, whereby an OnPart of that SEntry occurs.

For example : A Task might be repeated each time a certain document is created.

Parameters :

- **Condition** : A condition that is defined as an Expression. An expression that **MUST** evaluate to boolean. If the expression evaluated to **TRUE**, then the instance of the Task, Stage or Milestone may be repeated, otherwise it **MUST NOT** be repeated.

```
<cmmn:repetitionRule id="RepetitionRule_1a5vcca">
  <cmmn:condition>context.GetNumberVariable("increment") &lt; 2</cmmn:condition>
</cmmn:repetitionRule>
```