# Overview

## Problem description

During the corona virus pandemic, we have faced many difficulties and one of them is the handling of administrative affairs. For safety people tend not to leave the house, tackling every chore at home. With that trend, I decided to create a web application that helps people handle administrative tasks at home through the internet.

The procedure for handling an administrative task can be simplified as follows:

The administrators of the organization create a form template and public to the organization’s clients. Clients will fill out some sections in this form, sign it, go to the organization, and submits the form to the authorized people in the organization, who has responsibility to confirm the information of the client in this form, fill some section in the form and then store the form in the organization's data (in paper or in electronic form). The administrative task is completed.

The web application will simulate the procedure for handling the administrative task as follow:

Administrators of the organization structures a form and digitally sign it. Clients search for the form template, that they need. Clients and staffs will fill out their respective information in the form, digitally sign it and send this information to the server. The information from creating the form structure by administrator, creating an instance of the form by a client, filling, or editing information in the form by the client and the staff, ... will be saved by blockchain. Edits made by unauthorized users will not be added to the blockchain. Blockchain will send to everyone in the organization, and they will decide to save it to their local storage or not. The administrative task is completed.

# Use case overview (see use case diagrams in file Use\_Case\_Diagram.mdj or file html-docs/index.html)

User can login and perform tasks according to his role. There are 3 roles: admin, staff, and client.

Only admins can manage form structure, manage other users, and manage roles.

Client or staff can initiate a form instance. Then participant can submit the form instance to other participant, that responsible to one or more sections in the form.

Staffs and clients will participate in a form instance and they must digitally sign their own section in that form instance.

Staffs and clients will manage their own form instance.

# User’s use cases:

## Login to account

1. User requests to the “login page”.
2. User fills the user name (or email) and password input fields and submits to the server. [if login fail, see exception a]

Exception a: User fill wrong user name and/or password

1. Notify user about the error(s).
2. User fills the user name and password again and submits to the server

Result: Login success, navigate user to the “home page”.

## Register an account

1. User requests to the “register an account page”.
2. User fills the personal information (user name, password, first name, last name, email, phone,…) and submits to the server[Exception a]

Exception a: User fill wrong information.

1. Notify user about the error(s)
2. User fills the personal again and submits to the server.

Result: registration success, navigate user to the “home page”.

## Reset password

1. User requests to the “reset password page”.
2. User fills the old password, new password, confirm new password input and submits to the server. [exception a]

Exception a: User fill wrong old password or/and new password or/and confirm new password input.

1. Notify user about the error(s).
2. User fills the old password, new password, confirm new password input again and submits to the server.

Result: Reset password success, navigate user to the “home page”.

## Search form model

1. At the home page, user click the button “view all form templates” to view all the form templates that user has permission to edit an instance of them.
2. Users can search for a form template by name or by id.

Result: User can choose the form template to edit.

## Logout account

1. User click the logout button [exception a]

Exception a: User is editing a form instance but didn’t submit to the server.

1. Notify user to complete the form and submit to the server or discard the edit.

Result: Logout success, navigate user to the “login page”.

# Admin’s use cases

## Manage form structure

### Create a form structure

1. At the home page, admin clicks “create form structure” button.
2. Admin creates a form with an UI, digitally signs the form structure, then submits the form structure in JSON format to sever. [Exception a, b]

Exception a: Form structure is not in the right format.

1. Notify user about the error(s) and return to the “form structure editor”.

Exception b: User did not digitally sign the form structure.

1. Notify user to sign digitally the form structure and return to the “form structure editor”.

Result: Create form structure success, navigate user to the “view all format templates” page

### Update a form structure

1. At the “view all form templates” page, admin choose a template to edit the form structure.
2. Admin edits the form structure, digitally sign the form structure again and submit to the server. [exception a, b]
3. Notify admin, that all the form instances of this template will effect the client.

Exception a: Form structure is not in the right format.

1. Notify user about the error(s) and return to the “form structure editor”.

Exception b: User did not digitally sign the form structure.

1. Notify user to sign digitally the form structure and return to the “form structure editor”.

Result: update form structure success, navigate user to the “view all format templates” page

### Delete a form template

1. At the “view all form templates” page, admin chooses a form template to delete.
2. Notify admin, that delete form template will delete all its instances.

Result: Delete form template success.

## Manage users

### Update role of a user

1. At the “home page”, admin click button “view all users”.
2. Admin can choose a user and updates role of this user [Exception a]

Exception a: user already has this role.

1. Notify admin that user already has this role.
2. Return admin to the “view all users” page

Result: Update role of a user success. Navigate admin to the “view all users” page

## Manage role

### Create new role

1. At the “home page” admin clicks the button “view all roles” to view all roles in the organization.
2. At the “view all roles” admin clicks the button “add new role”.
3. Admin edits the role in the “role editor”.
4. Admin click button “save” to save the role. [exception a]

Exception a: Role name already existed.

1. All role name must be unique. Admin change the role name and save again.

Result: Create new role success, navigate admin to the “view all roles” page.

### Update role

At the “view all roles” page, admin chooses a role to update by clicking the button “update this role”.

Admin edits the role in the “role editor”.

Admin click button “save” to save the role or click “cancel” to cancel the update. [exception a]

Exception a: Role name already existed.

All role name must be unique. Admin change the role name and save again.

Result: Update role success, navigate admin to the “view all roles” page.

# Staff’s use cases

## Manage form instance

### Update a form instance

1. At the “home page”, staff clicks button “view all form instances” to view all his own form instances.
2. At the “view all form instances” page, staff choose a form instance to update his form instance by clicking “update this form instance” button.
3. In the form instance editor, staff update his form instance, digitally signs the form again and submit to the server.

Result: Update form instance success, navigate staff to the “view all form instances” page.

### Delete a form instance

1. At the “view all form instances” page, staff choose a form instance to delete his form instance by clicking “delete this form instance” button.
2. Notify staff, that the form will be deleted from the server.

Result: Delete a form instance success, navigate staff to the “view all form instances” page.

# Client’s use cases

## Manage form instance

### Update a form instance

1. At the “home page”, client clicks button “view all form instances” to view all his own form instances.
2. At the “view all form instances” page, client choose a form instance to update his form instance by clicking “update this form instance” button.
3. In the form instance editor, client update his form instance, digitally signs the form again and submit to the server.

Result: Update form instance success, navigate client to the “view all form instances” page.

### Delete a form instance

1. At the “view all form instances” page, client choose a form instance to delete his form instance by clicking “delete this form instance” button.
2. Notify client, that the form will be deleted from the server.

Result: Delete a form instance success, navigate client to the “view all form instances” page.