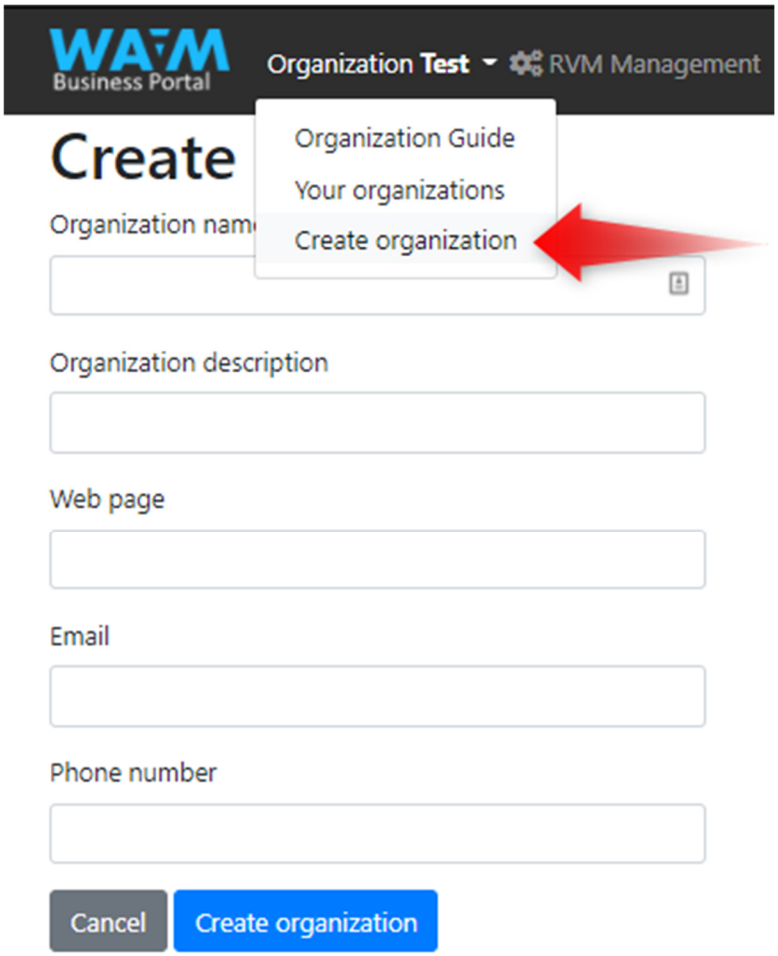
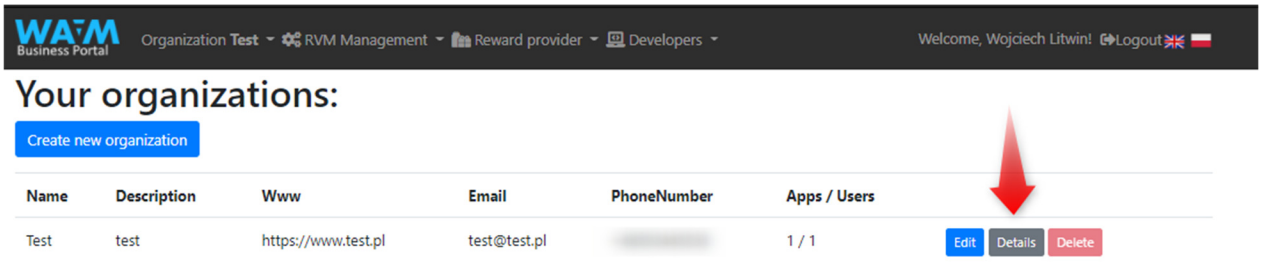
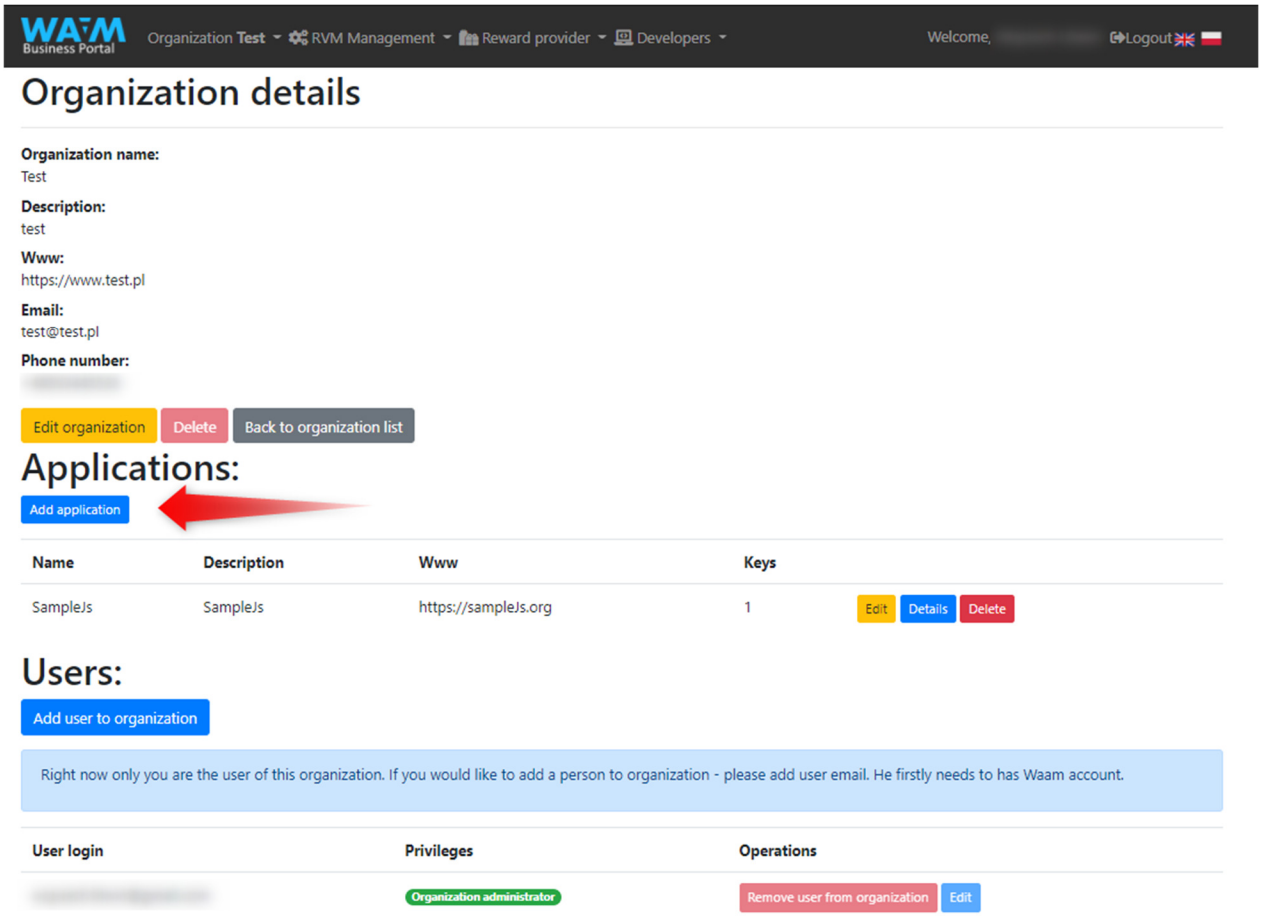
1) Create account at https://account.waam-machines.com
2) Confirm your email
3) Go to https://business.waam-machines.com and log in
4) Create new organization



5) Go to organization details

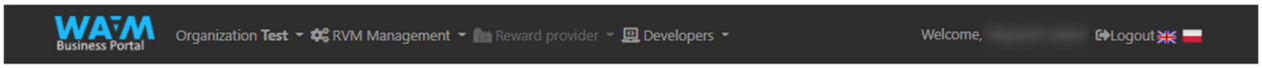


6) Add new application



6) Fill the data, but remember:
    a. Application name must be unique
    b. Application logo URL is optional
    c. Redirect URI is the uri that you will be redirected to after successful logon
    d. Post logout redirect uri will be the address you will be redirected after logout
    e. You need to specify the correct OAuth grant type. If you want more information there are some links on the form.

For the sample application in javascript – select Code OAuth grant type.



Create application

Application name

SampleJs

Application description

SampleJs

Application WWW

https://SampleJs.org

Application logo url

https://SampleJs.org

☐ Does your application requires PKCE?

Redirect Uris

https://localhost:5006/callback.html

If more then one, separate with semicolon ;

Post logout redirect Uris

https://localhost:5006/index.html

If more then one, separate with semicolon

Open Id Front Channel logout Uri

Scopes (more info):

- ☑ Open Id (scope: openid)
- ☑ Basic Profile information (scope: profile)
- ☑ Read account ballance (scope: developers.read_ballance)
- ☑ Enable in-app purchase (scope: developers.purchase)
- ☑ Read account history (scope: developers.read_history)

Select OAuth grant type (ℹ more info : Grant types, OAuth 2 Simplified, A Guide To OAuth 2.0 Grants)

- ○ Client credentials
- ◉ Code
- ○ DeviceFlow
- ○ Code and client credentials
- ○ Hybrid
- ○ Hybrid and client credentials
- ○ Implicit
- ○ Implicit and client credentials
- ○ Resource owner password
- ○ Resource owner password and client credentials

Cancel | Create application

Sample Javascript application.

It contains 3 files:

index.html - main file

callback.html – page called after logging in

app.js – javascript code

index.html:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <title></title>
</head>
<body>
    <button id="login">Login</button>
    <button id="getClaims">get my claims</button>
    <button id="getWallet">get my wallet</button>
    <button id="logout">Logout</button>

    <pre id="results"></pre>

    <script src="oidc-client.js"></script>
    <script src="app.js"></script>
</body>
</html>
```

Callback.html:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <title></title>
</head>
<body>
    <script src="oidc-client.js"></script>
    <script>
        new Oidc.UserManager({response_mode:"query"}).signinRedirectCallback().then(f
unction() {
            window.location = "index.html";
        }).catch(function(e) {
            console.error(e);
        });
    </script>
</body>
</html>
```

App.js

Remember – that your addresses needs to be the same as in the App Registration

```
function log() {
    document.getElementById('results').innerText = '';

    Array.prototype.forEach.call(arguments, function (msg) {
        if (msg instanceof Error) {
            msg = "Error: " + msg.message;
        }
        else if (typeof msg !== 'string') {
            msg = JSON.stringify(msg, null, 2);
        }
        document.getElementById('results').innerHTML += msg + '\r\n';
    });
}

document.getElementById("login").addEventListener("click", login, false);
document.getElementById("getClaims").addEventListener("click", getClaims, false);
document.getElementById("getWallet").addEventListener("click", getWallet, false);
document.getElementById("logout").addEventListener("click", logout, false);

var config = {
    authority: "https://account.waam-machines.com",
    client_id: "SampleJs",
    redirect_uri: "https://localhost:5011/callback.html",
    response_type: "code",
    scope: "openid profile developers.read_balance_current_user",
    post_logout_redirect_uri: "https://localhost:5011/index.html",
};
var mgr = new Oidc.UserManager(config);
mgr.getUser().then(function (user) {
    if (user) {
        log("User logged in", user.profile);
    }
    else {
        ``
        log("User not logged in");
    }
});

function login() {
    mgr.signinRedirect();
}

function getClaims() {
    mgr.getUser().then(function (user) {
        var url = "https://api.waam-machines.com/identity";

        var xhr = new XMLHttpRequest();
        xhr.open("GET", url);
        xhr.onload = function () {
            log(xhr.status, JSON.parse(xhr.responseText));
        }
```

```
        xhr.setRequestHeader("Authorization", "Bearer " + user.access_token);
        xhr.send();
    });
}

function getWallet() {
    mgr.getUser().then(function (user) {
        var url = "https://api.waam-machines.com/api/v1/users/me/wallet";

        var xhr = new XMLHttpRequest();
        xhr.open("GET", url);
        xhr.onload = function () {
            log(xhr.status, JSON.parse(xhr.responseText));
        }
        xhr.setRequestHeader("Authorization", "Bearer " + user.access_token);
        xhr.send();
    });
}

function logout() {
    mgr.signoutRedirect();
}
```