

# Module - creating a new icon

In this tutorial we will describe how to create, register and use a new icon in our module.

## What do you need before you start

- You need to install CzechIdM 9.4.0 (and higher). We have CzechIdM installed for this tutorial on server <http://localhost:8080/idm-backend>.

Source codes for this tutorial can be found in the [example module](#)

## 01 Create a icon component

We will use AbstractIcon component as superclass for creating custom icon (prepared in core advanced component library) - contains some default method implementations (common icon features as style, color ... boiler plate).

```
import React from 'react';
import { faBookmark, faKey } from '@fortawesome/free-solid-svg-icons';
import { FontAwesomeIcon } from '@fortawesome/react-fontawesome';
//
import { Advanced } from 'czechidm-core';

/**
 * Role icon - override role icon in example module.
 *
 * @author Radek Tomiška
 * @since 9.4.0
 */
export default class ExampleRoleIcon extends Advanced.AbstractIcon {

  constructor(props) {
    super(props);
  }

  renderIcon() {
    return (
      <span className="fa-layers fa-fw">
        <FontAwesomeIcon icon={ faBookmark } />
        <FontAwesomeIcon icon={ faKey } transform="shrink-7 up-2" style={{
color: '#fff' }} />
      </span>
    );
  }
}
```

We are using [font awesome](#) icon library for icon implementation, which is already configured in core.

Icon can be combined using [layers](#).

## 02 Register icon component

Component is created, now we need to register her in [component descriptor](#) for start using her in application. New icon will be available for the entity type (~name) my-roles.

Add component into module's component-descriptor.js:

```
...  
module.exports = {  
  ...  
  'components': [  
    ...  
    {  
      'id': 'custom-roles-icon',  
      'type': 'icon',  
      'entityType': [ 'my-roles'],  
      'component': require('./src/components/ExampleRoleIcon')  
    }  
    ...  
  ]  
};  
...
```

my-roles works as name for this icon.

## 03 Use icon

Icon can be used the same way as standard icon - the only difference is usage component as icon type:

```
...  
<Basic.Icon value="component:my-roles"/>  
...
```



## Advanced

We can override existing icon defined in core module.

It's just about editing module's `component.xml` descriptor:

```
...  
module.exports = {  
  ...  
  'components': [  
    ...  
    {  
      'id': 'custom-roles-icon',  
      'type': 'icon',  
      'entityType': [ 'my-roles', 'roles'],  
      'component': require('./src/components/ExampleRoleIcon'),  
      'priority': 1  
    }  
    ...  
  ]  
};  
...
```

We used previously created custom icon and define to use her for the roles entity types. We need to define higher priority (e.g. **1**) for this component, than is in core (**core** components have always **zero priority**).

From:

<https://wiki.czechidm.com/> - CzechIdM Identity Manager

Permanent link:

[https://wiki.czechidm.com/tutorial/dev/how\\_to\\_register\\_icon](https://wiki.czechidm.com/tutorial/dev/how_to_register_icon)

Last update: **2019/02/04 11:08**

