Permissions coding

Since we have introduced permissions, all coding that has something to do with access control and authorization will need to refer to permissions, and not to user roles, because we don't want to hard-code roles into the code. Permissions are going to be hard coded instead, which makes the system more flexible to future changes of tasks associated with the jobs in the museum.

Firts of all you need to execute a special migration in your system to enable the permissions database:

```
./yii migrate --migrationPath=@yii/rbac/migrations/
```

Let's start with the basic question: **how do I create a permission?**
There is no graphical interface for this designed into the system, because creation/editing/removal of permissions is an operation that is done only by the developers, and never by the admins and users of the system.
Since the standard implementation of the permissions system in Yii stores permissions in the database, we need to insert a record in the database for each permission that we want to implement into the system.
In my opinion the best way to do this is creating migrations that insert permissions into the system. I this way when the system gets installed on the future machine of the museum they will get all the necessary permissions by just executing the standard migrations.

In terms of coding. To create a permissions you need to call:

```
$permission=Yii::$app->authManager->createPermission("PermissionName");
$permission->description="Put here a meaningful description, it will be shown to the admins";
Yii::$app->authManager->add($permission);
```

In this way you have added the permission into the system and you can start using it in your controllers.

NOTE: I have already created a set of permissions that I think we need, so before your create new permissions check if the one you want already exists in /permissions/index

How do I use permissions?
In the controllers you usually need to check if a user has a permissions to do a specific action. You can do this using the AccessControl filter as described here:
http://www.yiiframework.com/doc-2.0/guide-security-authorization.html#access-control-filter

In the above documentation it is not mentioned that in the ‘role’ array you can also specify names of the permissions and not only names of roles (which we should not use). So an example of AccessControl usage can be:

```
public function behaviors()
    {
        return [
            'access' => [
                'class' => AccessControl::className(),
                'only' => ['create', 'update', 'delete'],
                'rules' => [
                    [
                        'allow' => true,
                        'actions' => ['create','update',
'delete'],
                        'roles' => ['manageTimeslots'],
                    ],
                ],
            ],
        ];
    }
```

Usually you wouldn’t need more than this in a controller. But if you do, you can manually check permissions of the current user in the body for the action by calling:

```
Yii::$app->user->can(“PermissionName”);
```

You can do the same in views, when (for example) you need to hide buttons or links to certain users

How do I set my system to test my permissions work?
I have created a setup controller to create one initial user that will be given the admin role with full access to the system. (it will also create the Admin, Coordinator and Instructor roles)
Just visit: /setup
Then you can login with it and test all you need. You can also create new roles and adjust permissions for all of them while you are logged in with it from the /permission controller