Creating a multilanguage application

Every day, we meet more and more international companies, software products, and information resources that publish content on multiple languages. Yii2 provides built-in i18n support for making multilanguage applications.

In this recipe, we are translating the application interface to different languages.

Getting ready

Create a new yii2-app-basic application using the composer, as described in the official guide at <http://www.yiiframework.com/doc-2.0/guide-start-installation.html>.

How to do it...

1. Change the main menu labels in the views/layouts/main. php file to use the Yii: : t(' app/nav', '...') method:

echo Nav::widget([

'options' => ['class' => 'navbar-nav navbar-right'],

'items' => [

['label' => Yii::t('app/nav', 'Home'), 'url' => ['/site/index']],

['label' => Yii::t('app/nav', 'About'), 'url' => ['/site/about']],

['label' => Yii::t('app/nav', 'Contact'), 'url' => ['/site/contact']],

], " '

]);

1. Change all your titles and breadcrumbs to use the common Yii ::t(' app, '...') method:

$this->title = Yii::t('app', 'Contact');

$this->params['breadcrumbs'][] = $this->title;

1. Also, change all the labels of your buttons:

<div class="form-group">

<?= Html::submitButton(Yii::t('app', 'Submit'), ['class' => 'btn btn- primary'']) ?>

</div>

Change other hard-coded messages as well:

<p>

<?= Yii::t('app', 'The above error occurred while the Web server was processing your request.') ?>

</p>

1. Change the attribute labels of your ContactForm model:

class LoginForm extends Model {

public function attributeLabels() {

return [