

10. Жүйеге кіру беттерін өзгерту

Жоспар:

10.1 Жүйеде тіркелген қолданушылар бойынша жүйеге кіру функционалдықтарын оңтайландыру

10.2 Қолданушы паролі атты өрісін орнату және оның сақталуын қамтамасыз ету

10.3 «Жаңа қолданушыны тіркеу» функциясын қосу

10.4 «Парольді қалпына келтіру» функциясын қосу

Мақсаты: Қолданушыны жүйеде тіркеу, оның паролін орнату және қайта қалпына келтіру функцияларын жобада қосуды үйрену.

Кілтті сөздер: жүйеге кіру, қолданушы паролі, жаңа қолданушыны тіркеу, парольді қалпына келтіру.

10.1 Жүйеде тіркелген қолданушылар бойынша жүйеге кіру функционалдықтарын оңтайландыру

Алдыңғы тарауды қолданушылар туралы деректері бар негізгі беттеріне қажетті қосымша функционалдық мүмкіндіктерді қосқан болатынбыз. Енді жүйеде тіркелген қолданушылардың жүйеге кіру бойынша қажетті функцияларын жобада қосуымыз қажет.

Алғашқы тарауларда Gii модулімен жаңадан User.php файлын қалыптастырар алдында жобаның C:\OpenServer\domains\academ-cloud.kz\models каталогында жатқан User.php файлының көшірмесін жасап, оның атауын User1.php деп өзгерткенбіз. Аталған файлдан төменде көрсетілген функцияларды өзіміздің User.php файлына көшірейік және кейбір функцияларды қажетінше толықтырайық (аталған файлдар C:\OpenServer\domains\academ-cloud.kz\models каталогында жатыр):

```
public static function findIdentity($id)
{
    return static::findOne(['id' => $id, 'status' =>
self::STATUS_ACTIVE]);
}

public static function findIdentityByAccessToken($token,
$type = null)
{
```

```

        throw new
NotSupportedException('"findIdentityByAccessToken" is not
implemented.');
```

```

    }

    public static function findByUsername($username)
    {
        return static::findOne(['username' => $username, 'status'
=> self::STATUS_ACTIVE]);
    }

    public function getId()
    {
        return $this->getPrimaryKey();
    }

    public function getAuthKey()
    {
        return $this->auth_key;
    }

    public function validateAuthKey($authKey)
    {
        return $this->getAuthKey() === $authKey;
    }

    public function validatePassword($password)
    {
        return Yii::$app->security->validatePassword($password,
$this->password_hash);
    }

```

АҒЫМДАҒЫ файлға кейін жобаға бірқатар функционалдық мүмкіндіктерді қосу қажет болатындықтан, келесідей функцияларды да қосқанымыз жөн:

```

public static function findByPasswordResetToken($token)
{
    if (!static::isPasswordResetTokenValid($token)) {
        return null;
    }
    return static::findOne([
        'password_reset_token' => $token,
        'status' => self::STATUS_ACTIVE,
    ]);
}

public static function isPasswordResetTokenValid($token)
{
    if (empty($token)) {
        return false;
    }

```

```

    }
    $timestamp = (int)substr($token, strrpos($token, '_') +
1);
    $expire = Yii::$app->
>params['user.passwordResetTokenExpire'];
    return $timestamp + $expire >= time();
}

public function generatePasswordResetToken()
{
    $this->password_reset_token = Yii::$app->security->
>generateRandomString() . '_' . time();
}

public function removePasswordResetToken()
{
    $this->password_reset_token = null;
}

```

Жоғарыдағы функцияларды қосқан соң төмендегі сурет 10.1-ге сәйкес \$password айнымалысын UserController.php файлындағы барлық функциялардың (аталған файл C:\OpenServer\domains\academ-cloud.kz\controllers каталогында орналасқан) алдында жариялауымыз керек.

```

C:\OpenServer\domains\academ-cloud.kz\controllers\UserController.php - Sublime Text 2 (UNREGISTERED)
Файл Правка Выделение Поиск Вид Перейти Инструменты Проект Опции Справка

UserController.php
1 <?php
2
3 namespace app\controllers;
4
5 use Yii;
6 use app\models\forms\UserForm;
7 use app\models\search\UserSearch;
8 use yii\web\Controller;
9 use yii\web\NotFoundHttpException;
10 use yii\filters\VerbFilter;
11
12
13 class UserController extends Controller
14 {
15
16     ... public $password;
17
18     public function behaviors()

```

Сурет 10.1. \$password айнымалысын жариялау

Айнымалыны жариялаған соң жүйеде тіркелген қолданушы арқылы жүйеге кіру процедурасын тексеруге болады, бірақ алдымен жүйеде тіркелген екі қолданушының біріне кәштелген парольді қолмен орнатуымыз керек (логині мен паролі сәйкесінше admin және admin болсын), «admin» паролінің кәштелген нұсқасын алу үшін

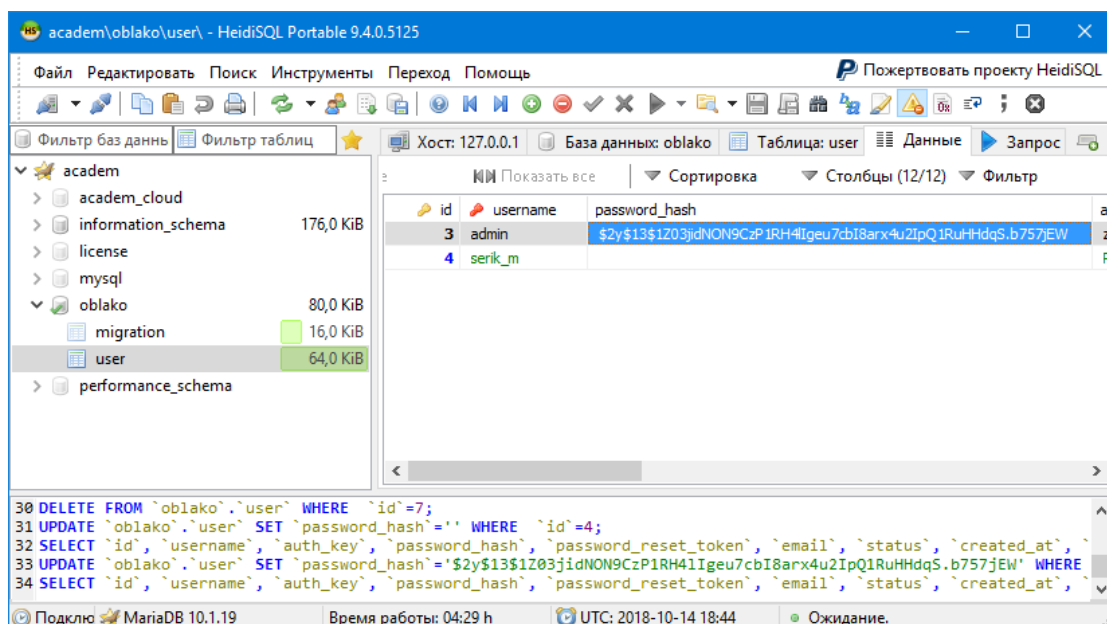
C:\OpenServer\domains\academ-cloud.kz\controllers каталогында жатқан SiteController.php файлыны ашып, төмендегідей жаңа action функциясын қосамыз:

```
public function actionAaa ()
{
    $password = Yii::$app->security->generatePasswordHash('admin');
    return $password;
}
```

Яғни «admin» паролінің кэштегелген нұсқасын алу үшін браузерде <http://academ-cloud.kz/site/aaa> сілтемесі арқылы алуға болады, біздің мысалымызда ол мынадай түрге ие болады:

\$2y\$13\$1z03jidNON9CzP1RH4lIgeu7cbI8arx4u2IpQ1RuHHdqS.b757jEW

Аталған кэштегелген пароль нұсқасын көшіріп алып, төмендегі сурет 10.2-де көрсетілгендей, MySQL менеджерін іске қосып, ондағы «oblako» деректер базасының «user» кестесіндегі «Данные» вкладкасына көшіп, admin қолданушысының «password_hash» бағанасына орнатып сақтаймыз.



Сурет 10.2. Паролінің кэштегелген нұсқасын база кестесінде орнату

Жоғарыдағы қадамдарды орындаған соң браузерде <http://academ-cloud.kz/site/login> сілтемесін немесе жобаның мәзір жолындағы «Войти в систему» пунктін таңдау арқылы қолданушының авторизациялық деректерін енгізетін «Жүйеге кіру» бетін ашатын боламыз. Аталған бетте сәйкесінше логин мен парольге admin және admin деп енгізіп, «Войти» батырмасын шертеміз. Егер жоғарыда сипатталған барлық қадамдар дұрыс орындалған жағдайда аталған қолданушының атынан біз жүйеге кіретін боламыз.

Жүйеге кіргенімізді басты беттегі «Құрметті admin! Бұлтқа қош келдіңіз!» деген сөздерінен, сонымен қатар жобаның мәзір жолындағы «Жүйеге кіру» пунктін таңдаған кезде «Жүйеге кіру» бетіне өтпейтінінен білуге болады.

10.2 Қолданушы паролі атты өрісін орнату және оның сақталуын қамтамасыз ету

Қолданушы деректері бар беттермен жұмыс істеу барысында оның құрамында паролі бар өріс болмаған еді, сол себепті авторизациялық деректері ретінде қолданушының паролін орнату қажеттілігі туындайды. Ол үшін ең алдымен C:\OpenServer\domains\academ-cloud.kz\models каталогына өтіп, жаңадан «forms» деген каталогына UserForm.php атты жаңа файлын құрып, ондағы келесі кодтарды тереміз:

```
<?php

namespace app\models\forms;
use app\models\User;

class UserForm extends User
{
    public $password;

    public function rules()
    {
        $rules = parent::rules();
        $rules[] = ['password', 'string'];
        return $rules;
    }

    public function beforeSave($insert)
    {
        if ($this->password) {
```

```

        $this->setPassword($this->password);
        $this->generateAuthKey();
    }
    return parent::beforeSave($insert);
}
}

```

Содан кейін C:\OpenServer\domains\academ-cloud.kz\controllers каталогында жатқан UserController.php файлы ашып, жоғары бөлігінде орналасқан төмендегі файлдар сілтемесі бар бірінші код жолын екінші код жолымен алмастырамыз:

```
use app\models\User; → use app\models\forms\UserForm;
```

Модель файлы көрсететін сілтеме ауысқандықтан, ағымдағы файлда actionCreate функциясында кездесетін келесі бірінші код жолын екінші код жолымен алмастырамыз:

```
$model = new User(); → $model = new UserForm();
```

Ағымдағы файлдың төменгі бөлігінде findModel(\$id) функциясына көшіп, онда кездесетін келесі бірінші код жолын екінші код жолымен алмастырамыз:

```
if (($model = User::findOne($id)) !== null) → if (($model = UserForm::findOne($id)) !== null)
```

Парольді орнату және авторизациялық кілтті генерациялау бойынша қосымша келесідей функцияларды User.php модель файлына (бұл файл C:\OpenServer\domains\academ-cloud.kz\models каталогында орналасқан) қосамыз:

```

public function setPassword($password)
{
    $this->password_hash = Yii::$app->security->generatePasswordHash($password);
}

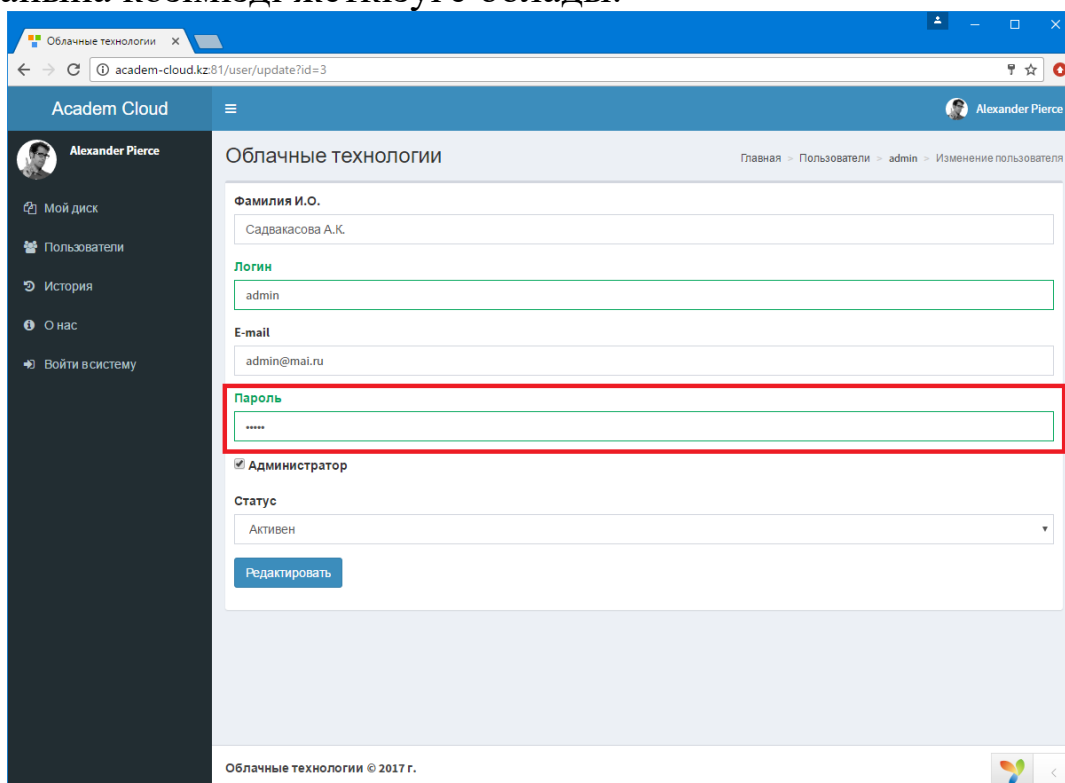
public function generateAuthKey()
{
    $this->auth_key = Yii::$app->security->generateRandomString();
}

```

Ең соңында C:\OpenServer\domains\academ-cloud.kz\views\user каталогына көшіп, ондағы _form.php файлын ашып, келесі түрге ие қосымша паролі бар өрісті қосамыз:

```
<?=$form->field($model, 'password')->passwordInput([
    'autocomplete' => 'new-password',
]) ?>
```

Нәтижесінде төмендегі сурет 10.3-те көрсетілгендей, жаңа қолданушыны қосқан кезде немесе жүйеде тіркелген қолданушының деректерін өзгерткен кезде қосымша «Пароль» атты өрістің пайда болғанына көзімізді жеткізуге болады.



Сурет 10.3. Қолданушы паролі атты өрісін орнату кескіні

Қолданушының сәйкес деректерін енгізіп болған соң, деректерді сақтаған кезде паролдің кәштелген нұсқасының базада жазылғанын көруге болады және қолданушының сақтаған авторизациялық деректері (логин мен пароль) арқылы жүйеге кіруге болады.

«Жүйеде кіру» бетінде авторизациялық деректерді қате енгізген жағдайда көрсетілетін «Incorrect username or password» жазбасын өзгерту қажеттілігі туындайды, ол үшін UserController.php файлын ашып, validatePassword функциясының ішіндегі жоғарыдағы

жазбаны «Некорректно были введены авторизационные данные пользователя» деп алмастыру қажет.

10.3 «Жаңа қолданушыны тіркеу» функциясын қосу

Жүйеде тіркелмеген қолданушы өзін-өзі жүйеде тіркеу қажеттілігі туындайды. Ол үшін «Жүйеге кіру» бетіндегі алдын-ала дайындалған «Зарегистрироваться» сілтемесі бар жазбасын қолдану керек. Арнайы «Жаңа қолданушыны тіркеу» функциясын қосу үшін төмендегі қадамдарды орындау керек.

Алдымен жаңа қолданушыны тіркеу моделін құруымыз керек, ол үшін C:\OpenServer\domains\academ-cloud.kz\models каталогына өтіп, онда жаңа SignupForm.php атты файлың құрып, келесі кодтарды тереміз:

```
<?php

namespace app\models;

use Yii;
use yii\base\Model;

class SignupForm extends Model
{
    public $username;
    public $email;
    public $password;

    public function rules()
    {
        return [
            ['username', 'trim'],
            ['username', 'required'],
            ['username', 'unique', 'targetClass' =>
'\app\models\User', 'message' => 'В системе зарегистрирован
такой логин!'],
            ['username', 'string', 'min' => 2, 'max' => 255],
            ['email', 'trim'],
            ['email', 'required'],
            ['email', 'email'],
            ['email', 'string', 'max' => 255],
            ['email', 'unique', 'targetClass' =>
'\app\models\User', 'message' => 'В системе зарегистрирован
пользователь с такой электронной почтой'],
            ['password', 'required'],
            ['password', 'string', 'min' => 6],
        ];
    }
}
```



```

    }

    public function signup()
    {
        if (!$this->validate()) {
            return null;
        }
        $user = new User();
        $user->username = $this->username;
        $user->email = $this->email;
        $user->setPassword($this->password);
        $user->generateAuthKey();
        if ($user->save()) {
            return $user;
        }
        return null;
    }

    public function attributeLabels()
    {
        return [
            'fio' => 'Фамилия И.О.',
            'username' => 'Логин',
            'email' => 'E-mail',
            'password' => 'Пароль',
        ];
    }
}

```

Жаңа қолданушыны тіркеу моделін құрып болған соң C:\OpenServer\domains\academ-cloud.kz\controllers каталогында жатқан SiteController.php файлы ашып, жаңа actionSignup экшн функциясын қосамыз, аталған функция келесі кодқа сәйкес болуы тиіс:

```

public function actionSignup()
{
    $this->layout = 'main-login';
    $model = new SignupForm();
    if ($model->load(Yii::$app->request->post())) {
        if ($user = $model->signup()) {
            if (Yii::$app->getUser()->login($user)) {
                return $this->goHome();
            }
        }
    }
    return $this->render('signup', [
        'model' => $model,
    ]);
}

```

Аталған функцияда жоғарыда құрған модельге сілтемесі бар болғандықтан, ағымдағы файлдың жоғарғы бөлігінде сол модельдің сілтемесін тағайындау керек:

```
use app\models\SignupForm;
```

Алдыңғы тарауда көрсетілгендей, C:\OpenServer\domains\academ-cloud.kz\themes\site каталогында жатқан login.php файлын өзгерту кезінде «Тіркелу» жазбасы /site/signup жолына сілтемеленіп тұр, сол себепті жобаның C:\OpenServer\domains\academ-cloud.kz\views\site каталогына көшіп, жаңадан signup.php атты файлын құрамыз да, келесі кодтарды тереміз:

```
<?php

use yii\helpers\Html;
use yii\bootstrap\ActiveForm;
use yii\helpers\Url;

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

$fieldOptions1 = [
    'options' => ['class' => 'form-group has-feedback'],
    'inputTemplate' => "{input}<span class='glyphicon glyphicon-user form-control-feedback'></span>"
];
$fieldOptions2 = [
    'options' => ['class' => 'form-group has-feedback'],
    'inputTemplate' => "{input}<span class='glyphicon glyphicon-envelope form-control-feedback'></span>"
];
$fieldOptions3 = [
    'options' => ['class' => 'form-group has-feedback'],
    'inputTemplate' => "{input}<span class='glyphicon glyphicon-lock form-control-feedback'></span>"
];
?>
<div class="login-box">
    <div class="login-logo">
        <a href="#"><b>Облачные технологий</b></a>
    </div>
    <div class="login-box-body">
        <p class="login-box-msg">Регистрация пользователя</p>
        <?php $form = ActiveForm::begin(['id' => 'login-form', 'enableClientValidation' => false]); ?>
```

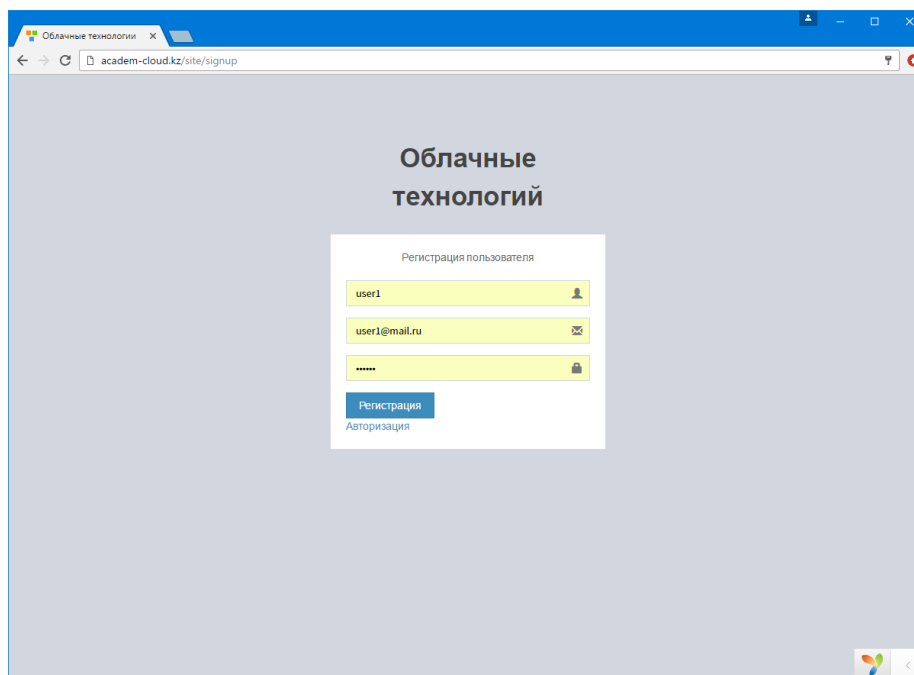
```

        <?= $form
        ->field($model, 'username', $fieldOptions1)
        ->label(false)
        ->textInput(['placeholder' => $model-
>getAttributeLabel('Логин')]) ?>
        <?= $form
        ->field($model, 'email', $fieldOptions2)
        ->label(false)
        ->textInput(['placeholder' => $model-
>getAttributeLabel('E-mail')]) ?>
        <?= $form
        ->field($model, 'password', $fieldOptions3)
        ->label(false)
        ->passwordInput(['placeholder' => $model-
>getAttributeLabel('Пароль')]) ?>
        <div class="row">
            <div class="col-xs-5">
                <?= Html::submitButton('Регистрация',
[ 'class' => 'btn btn-primary btn-block btn-flat', 'name' =>
'login-button']) ?>
            </div>
        </div>
        <div>
            <a href="<?=
Url::to(['/site/login']) ?>">Авторизация</a>
        </div>
        <?php ActiveForm::end(); ?>
    </div>
</div>

```

Нәтижесінде «Жүйеге кіру» бетіндегі «Тіркелу» атты сілтемесі бар жазбасын шерткен кезде төмендегі сурет 10.4-те көрсетілгендей, «Қолданушыны тіркеу» атты беті жүктелетін болады. Онда жүйеде тіркелмеген қолданушы өзінің авторизациялық деректерін енгізе отырып, төменде орналасқан «Тіркелу» атты батырмасын шерткен кезде өзін-өзі жүйеде тіркей алады.

Ағымдағы беттің төменгі жағында қайта «Жүйеге кіру» бетіне алмастыратын арнайы «Авторизация» атты сілтемесі бар жазбаны көруге болады.



Сурет 10.4. «Қолданушыны тіркеу» атты беттің көрінісі

10.4 «Парольді қалпына келтіру» функциясын қосу

Жүйеде тіркелген қолданушы өзінің паролін ұмытқан жағдайда оның паролін қалпына келтіру қажеттілігі туындайды. Ол үшін «Жүйеге кіру» бетіндегі алдын-ала дайындалған «Восстановить пароль» сілтемесі бар жазбасын қолдану керек. Арнайы «Парольді қалпына келтіру» функциясын қосу үшін төмендегі қадамдарды орындау керек.

Алдымен парольді қалпына келтіру моделдерін құруымыз керек, ол үшін `C:\OpenServer\domains\academ-cloud.kz\models` каталогына өтіп, онда жаңа `PasswordResetRequestForm.php` атты файлын құрып, келесі кодтарды тереміз:

```
<?php

namespace app\models;

use Yii;
use yii\base\Model;

class PasswordResetRequestForm extends Model
{
```

```

public $email;

public function rules()
{
    return [
        ['email', 'trim'],

        ['email', 'required'],
        ['email', 'email'],
        ['email', 'exist',
            'targetClass' => '\app\models\User',
            'filter' => ['status' =>
User::STATUS_ACTIVE],
            'message' => 'Не найдено в базе пользователь
с таким e-mail'
        ],
    ];
}

public function sendEmail()
{
    $user = User::findOne([
        'status' => User::STATUS_ACTIVE,
        'email' => $this->email,
    ]);
    if (!$user) {
        return false;
    }
    if (!User::isPasswordResetTokenValid($user->password_reset_token)) {
        $user->generatePasswordResetToken();
        if (!$user->save()) {
            return false;
        }
    }
    return Yii::$app
        ->mailer
        ->compose(
            ['html' => 'passwordResetToken-html', 'text'
=> 'passwordResetToken-text'],
            ['user' => $user]
        )
        ->setFrom([Yii::$app->params['supportEmail'] =>
Yii::$app->name . ' robot'])
        ->setTo($this->email)
        ->setSubject('Сброс пароля для пользователя - ' .
Yii::$app->name)
        ->send();
}
}

```

Сонымен қатар ағымдағы каталогта жаңа ResetPasswordForm.php файлын құрып, төмендегі кодтарды жазамыз:

```
<?php

namespace app\models;

use yii\base\Model;
use yii\base\InvalidArgumentException;

class ResetPasswordForm extends Model
{
    public $password;
    private $_user;

    public function __construct($token, $config = [])
    {
        if (empty($token) || !is_string($token)) {
            throw new InvalidArgumentException('Password reset
token cannot be blank.');
```

```
        }
        $this->_user =
User::findByPasswordResetToken($token);
        if (!$this->_user) {
            throw new InvalidArgumentException('Wrong password
reset token.');
```

```
        }
        parent::__construct($config);
    }

    public function rules()
    {
        return [
            ['password', 'required'],
            ['password', 'string', 'min' => 6],
        ];
    }

    public function resetPassword()
    {
        $user = $this->_user;
        $user->setPassword($this->password);
        $user->removePasswordResetToken();
        return $user->save(false);
    }
}
```

Парольді қалпына келтіру модельдерін құрып болған соң C:\OpenServer\domains\academ-cloud.kz\controllers каталогында

жатқан SiteController.php файлын ашып, жаңа actionRequestPasswordReset және actionResetPassword атты экшн функцияларын қосамыз, аталған функциялар келесі кодқа сәйкес болуы тиіс:

```
public function actionRequestPasswordReset()
{
    $this->layout = 'main-login';
    $model = new PasswordResetRequestForm();
    if ($model->load(Yii::$app->request->post()) && $model->validate()) {
        if ($model->sendEmail()) {
            Yii::$app->session->setFlash('success',
            'Проверите свой email для дальнейших инструкций.');
```

```
            return $this->goHome();
        } else {
            Yii::$app->session->setFlash('error', 'Извините,
            we are unable to reset password for email provided.');
```

```
        }
    }
    return $this->render('passwordResetRequestForm', [
        'model' => $model,
    ]);
}

public function actionResetPassword($token)
{
    try {
        $model = new ResetPasswordForm($token);
    } catch (InvalidParamException $e) {
        throw new BadRequestHttpException($e->getMessage());
    }
    if ($model->load(Yii::$app->request->post()) && $model->validate() && $model->resetPassword()) {
        Yii::$app->session->setFlash('success', 'Новый пароль
        успешно сохранен');
```

```
        return $this->goHome();
    }
    return $this->render('resetPasswordForm', [
        'model' => $model,
    ]);
}
```

Аталған функцияларда жоғарыда құрған модельдерге және қосымша кітапханаларға сілтемесі бар болғандықтан, ағымдағы файлдың жоғарғы бөлігінде сол модельдердің және кітапханалардың сілтемелерін тағайындау керек:

```
use app\models\PasswordResetRequestForm;
use app\models\ResetPasswordForm;
```

```
use yii\web\BadRequestHttpException;
use yii\base\InvalidParamException;
```

Жоғарыда айтылғандай, login.php файлын өзгерту кезінде «Восстановить пароль» жазбасы /site/request-password-reset жолына сілтемеленіп тұр, сол себепті жобаның C:\OpenServer\domains\academ-cloud.kz\views\site каталогына көшіп, passwordResetRequestForm.php атты файлын құрамыз да, келесі кодтарды тереміз:

```
<?php

use yii\bootstrap\ActiveForm;
use yii\helpers\Html;

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

$fieldOptions1 = [
    'options' => ['class' => 'form-group has-feedback'],
    'inputTemplate' => "{input}<span class='glyphicon glyphicon-envelope form-control-feedback'></span>"
];
?>
<div class="login-box">
    <div class="login-logo">
        <a href="#"><b>Облачные технологии</b></a>
    </div>
    <div class="login-box-body">
        <p class="login-box-msg">Восстановление пароля</p>
        <?php $form = ActiveForm::begin(['id' => 'login-form', 'enableClientValidation' => false]); ?>
        <?= $form
            ->field($model, 'email', $fieldOptions1)
            ->label(false)
            ->textInput(['placeholder' => $model->getAttributeLabel('Email')]) ?>
        <div class="row">
            <div class="col-xs-5">
                <?= Html::submitButton('Восстановить',
                    ['class' => 'btn btn-primary btn-block btn-flat', 'name' => 'login-button']) ?>
            </div>
        </div>
        <?php ActiveForm::end(); ?>
    </div>
</div>
```


Сонымен қатар ағымдағы каталогта resetPasswordForm.php атты файлын құрып, келесі кодтарды жазамыз:

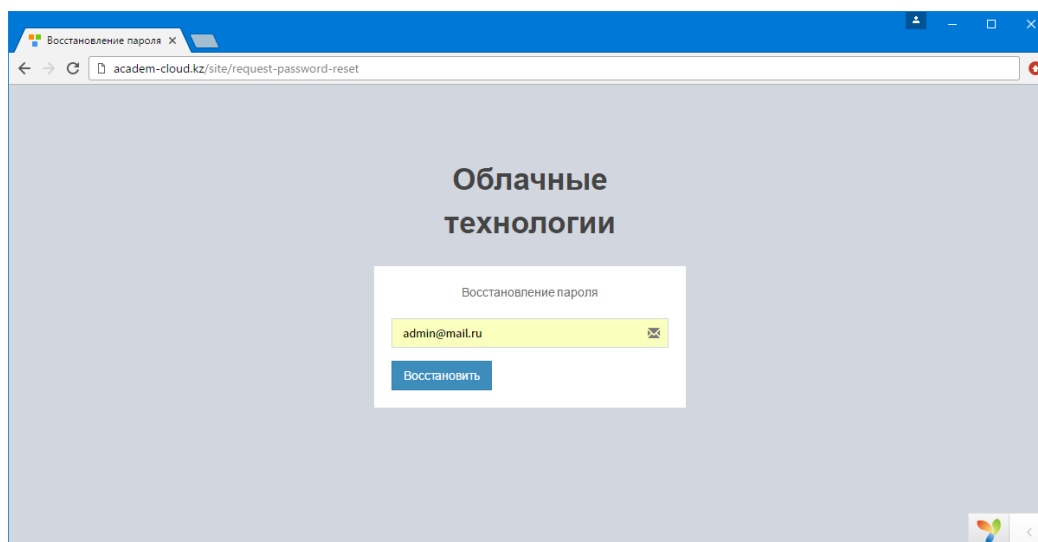
```
<?php

use yii\helpers\Html;
use yii\bootstrap\ActiveForm;

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

<div class="site-reset-password">
    <h1><?= Html::encode($this->title) ?></h1>
    <p>Please choose your new password:</p>
    <div class="row">
        <div class="col-lg-5">
            <?php $form = ActiveForm::begin(['id' => 'reset-
password-form']); ?>
            <?= $form->field($model, 'password')-
>passwordInput(['autofocus' => true]) ?>
            <div class="form-group">
                <?= Html::submitButton('Save', ['class' =>
'btn btn-primary']) ?>
            </div>
            <?php ActiveForm::end(); ?>
        </div>
    </div>
</div>
```

Нәтижесінде «Жүйеге кіру» бетіндегі «Парольді қалпына келтіру» атты сілтемесі бар жазбасын шерткен кезде төмендегі сурет 10.5-те көрсетілгендей, «Парольді қалпына келтіру» атты беті жүктелетін болады.



Сурет 10.5. «Парольді қалпына келтіру» атты беттің көрінісі

Онда жүйеде тіркелген қолданушы өзінің паролін ұмытқан жағдайда өзінің e-mail электрондық поштасын енгізе отырып, төменде орналасқан «Восстановить» атты батырмасын шерткен кезде көрсетілген электрондық поштасына парольді қалпына келтіру сілтемесі бар хат келетін болады.

Бақылау сұрақтары:

1. Қолданушылардың жүйеге кіруі қалай ұйымдастырылады?
2. Қолданушы паролі атты өрісі қалай орнатылады?
3. Жаңа қолданушыны тіркеу қалай орындалады?
4. Парольді қалпына келтіру қызметі қалай қосылады?