Работа с событиями

События Yii обеспечивают простую реализацию, которая позволяет прослушивать и подписываться на различные события, происходящие в вашем веб-приложении. Например, вы можете отправить уведомление о новой статье подписчикам при каждой публикации нового материала.

Подготовка

1. Создайте новое приложение с помощью composer, как описано в официальном руководстве по <http://www.yiiframework.com/doc-2.0/guide-start-installation.html>. (от переводчика по русски <http://yiiframework.domain-na.me/doc/guide/2.0/ru/start-installation> )
2. Выполните следующий код SQL на сервере, чтобы создать таблицу статей:

CREATE TABLE 'article' (

'id' int(11) NOT NULL AUTO\_INCREMENT,

'name' varchar(255) DEFAULT NULL,

'description' text,

PRIMARY KEY ('id')

) ENGINE=InnoDB AUTO\_INCREMENT=29 DEFAULT CHARSET=utf8;

1. 3. Создайте модель Article с помощью Gii.
2. Запустите ваш webserver командой ./yii serve.

Как это работает

1. Добавить тест действия в \controllers\SiteController:

public function actionTest()

{

$article = new Article();

$article->name = ‘День Святого Валентина приближается? Вот дерьмо! Я забыл снова завести девушку!';

$article->description = ‘Бендер злится на Фрая за то, что он встречается с роботом. Держись подальше от наших женщин. У тебя металлическая лихорадка, парень. Лихорадка металла';

// $event - объект класса yii\base\Event или дочернего класса

$article->on(ActiveRecord::EVENT\_AFTER\_INSERT, function($event) {

$followers = ['john2@teleworm.us', 'shivawhite@cuvox.de', 'kate@dayrep.com'

];

foreach($followers as $follower) {

Yii::$app->mailer->compose()

->setFrom('techblog@teleworm.us')

->setTo($follower)

->setSubject($event->sender ->name)

->setTextBody($event->sender->description)

->send();

}

echo 'Emails has been sent';

});

if (!$article->save()) {

echo VarDumper::dumpAsString($article->getErrors());

};

}

1. Обновите config/web.php, компонент mailer использует следующий код.