

# Тема 2. «Базовая настройка рабочего окружения»

Грушевский Ю.В.



Что такое Yii?



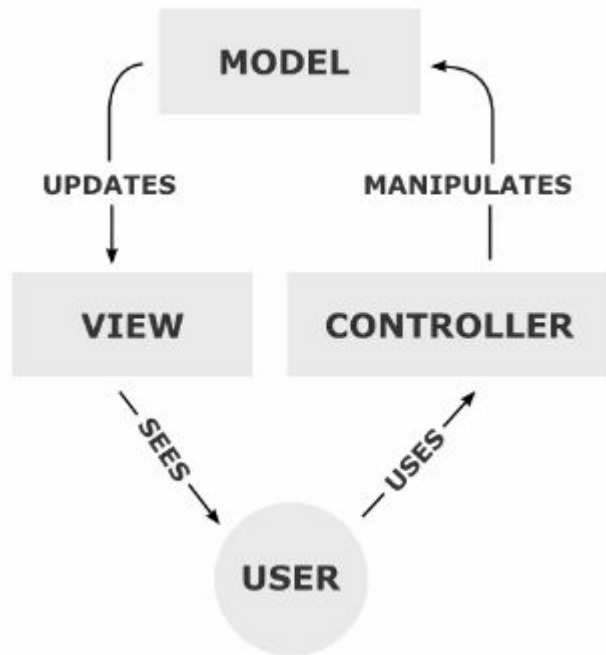
Yii – это высокопроизводительный компонентный PHP фреймворк, предназначенный для быстрой разработки современных веб-приложений.

# Особенности Yii

- Как и многие другие PHP фреймворки, для организации кода Yii использует архитектурный паттерн MVC (Model-View-Controller).
- Yii придерживается философии простого и элегантного кода, не пытаясь усложнять дизайн только ради следования каким-либо шаблонам проектирования.
- Yii является full-stack фреймворком и включает в себя проверенные и хорошо зарекомендовавшие себя возможности, такие как ActiveRecord для реляционных и NoSQL баз данных, поддержку REST API, многоуровневое кэширование и другие.
- Yii отлично расширяем. Вы можете настроить или заменить практически любую часть основного кода. Используя архитектуру расширений, легко делиться кодом или использовать код сообщества.
- Одна из главных целей Yii – производительность.

# Что такое MVC?

Model View Controller (Модель-Представление-Контроллер) — схема разделения данных приложения, и управляющей логики на три отдельных компонента: модель, представление и контроллер — таким образом, что модификация каждого компонента может осуществляться независимо



# Model

Логика манипулирования данными

Взаимодействия с БД (SELECT, INSERT, UPDATE, DELETE)

Предоставляет данные и реагирует на команды контроллера, изменяя свое состояние

# View

Отвечает за отображение данных модели пользователю, реагируя на изменения модели

Обычно содержит HTML & CSS

Общается с контроллером

Используются шаблонизаторы

# Controller

Получает данные

Обрабатывает запросы

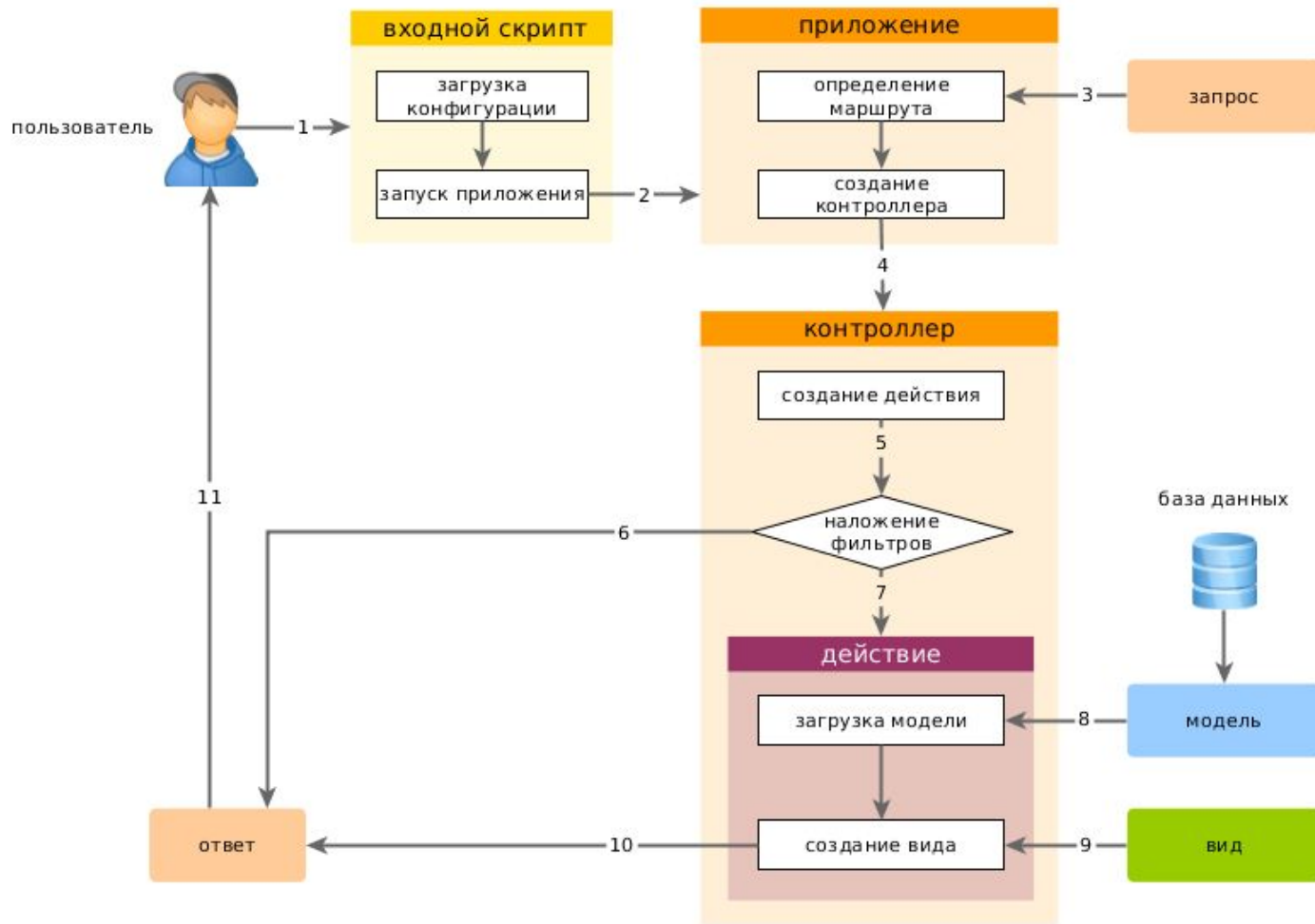
Получает данные из модели

Передаёт данные в представление



Кроме MVC, Yii приложения также имеют следующие сущности:

- входные скрипты;
- приложения;
- компоненты приложения;
- модули;
- фильтры;
- виджеты.





# Установка и настройка

# Установка

```
composer create-project --prefer-dist yiisoft/yii2-app-basic basic
```

Composer — это пакетный менеджер уровня приложений для языка программирования PHP, который предоставляет средства по управлению зависимостями в PHP-приложении.



# Установка в рамках инфраструктуры ДЭ

1. Копируем архив с фреймворком из директории public в необходимый модуль;
2. Распаковываем архив;
3. Настраиваем редиректы с помощью файла .htaccess;
4. Настраиваем параметры приложения и подключение к БД.

# .htaccess

RewriteEngine On

RewriteRule ^(.\*)\$ web/\$1

# web.php и db.php

```
web.php x
11     '@bower' => '@vendor/bower-asset',
12     '@npm'   => '@vendor/npm-asset',
13 },
14 'components' => [
15     'request' => [
16         // !!! insert a secret key in the following (if it is empty)
17         'cookieValidationKey' => 'randomstring',
18     ],
19 ],
```

```
db.php x
1 <?php
2
3 return [
4     'class' => 'yii\db\Connection',
5     'dsn'   => 'mysql:host=localhost;dbname=login_m1',
6     'username' => 'user',
7     'password' => 'pass',
8     'charset' => 'utf8',
9 ];
```

# Настройка ЧПУ

```
web.php x
14 'components' => [
15     'request' => [
16         // !!! insert a secret key in the following
17         'cookieValidationKey' => 'randomstring',
18         'baseUrl'=>'',
19     ],
```

```
web.php x
47
48 'urlManager' => [
49     'enablePrettyUrl' => true,
50     'showScriptName' => false,
51     'rules' => [
52         ],
53     ],
```



# Настройка локализации и gii

```
web.php x
1 <?php
2
3 $params = require __DIR__ . '/params.php';
4 $db = require __DIR__ . '/db.php';
5
6 $config = [
7     'id' => 'basic',
8     'language' => 'ru-RU',
```

```
web.php x
64 // uncomment the following to add your IP
65 // 'allowedIPs' => ['127.0.0.1', ':::1'],
66 ];
67 $config['bootstrap'][] = 'gii';
68 $config['modules']['gii'] = [
69     'class' => 'yii\gii\Module',
70     // uncomment the following to add your IP
71     'allowedIPs' => ['*'],
72 ];
73 }
```



Разработка БД

# Выделение базовых сущностей

- Пользователь
- Роль пользователя
- Заявка
- Статус заявки

# Пользователь

Сущность	Поле	Тип
user	id	счетчик
	username	строка, уникальный
	password	строка
	email	строка, уникальный
	first_name	строка
	last_name	строка
	middle_name	строка
	<i>role_id</i>	<i>идентификатор роли</i>

# Роль пользователя

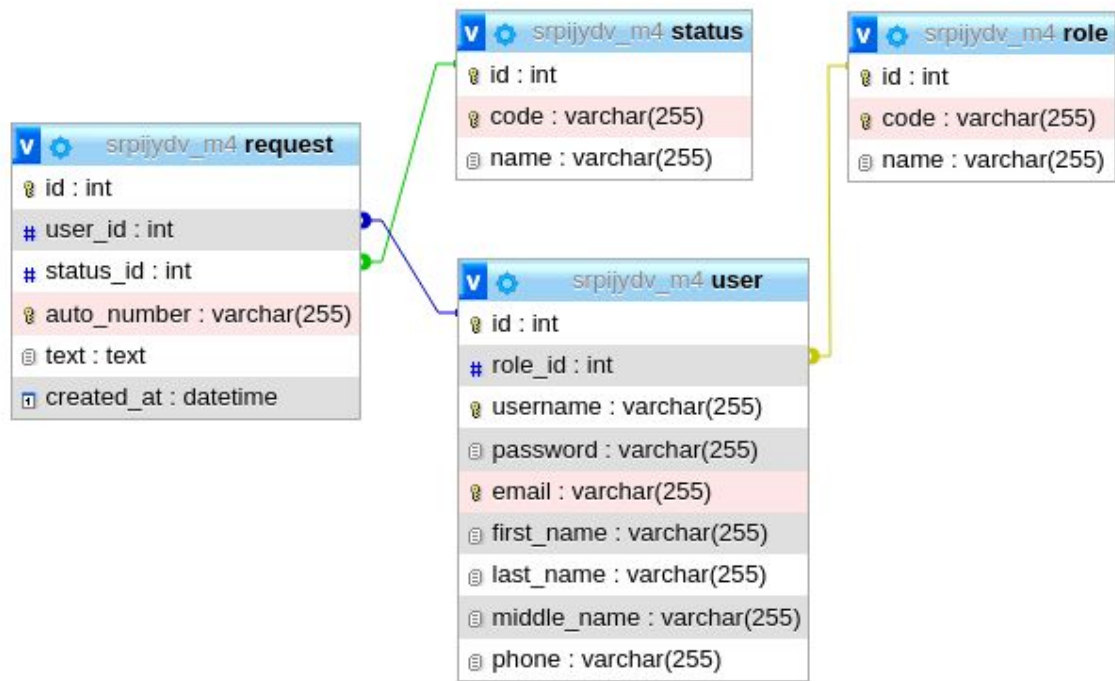
Сущность	Поле	Тип
role	id	счетчик
	code	строка, уникальный
	name	строка

# Заявка

Сущность	Поле	Тип
request	id	счетчик
	<i>user_id</i>	<i>идентификатор пользователя</i>
	auto_number	строка
	text	текст
	дата и время	дататайм, таймстамп
	<i>status_id</i>	<i>идентификатор статуса</i>

# Заявка

Сущность	Поле	Тип
status	id	счетчик
	code	строка, уникальный
	name	строка







# Работа с БД с помощью миграций

# Миграции

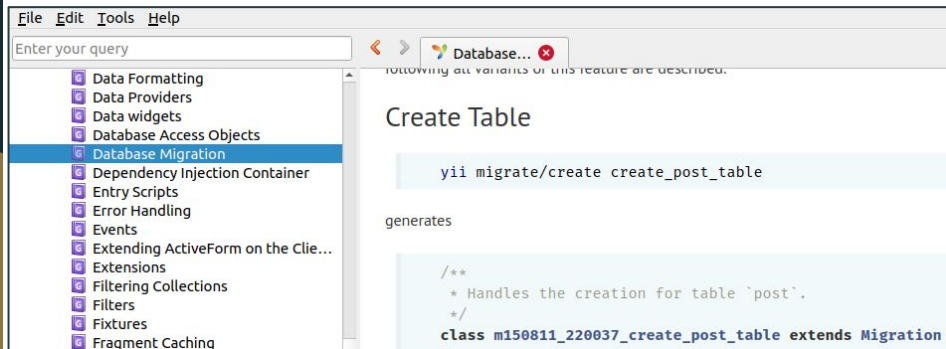
Миграции — что-то вроде системы контроля версий для вашей базы данных. Они позволяют вашей команде изменять структуру БД, в то же время оставаясь в курсе изменений других участников.

Yii предоставляет набор инструментов для миграций из командной строки, которые позволяют:

- создавать новые миграции;
- применять миграции;
- отменять миграции;
- применять миграции повторно;
- показывать историю и статус миграций.

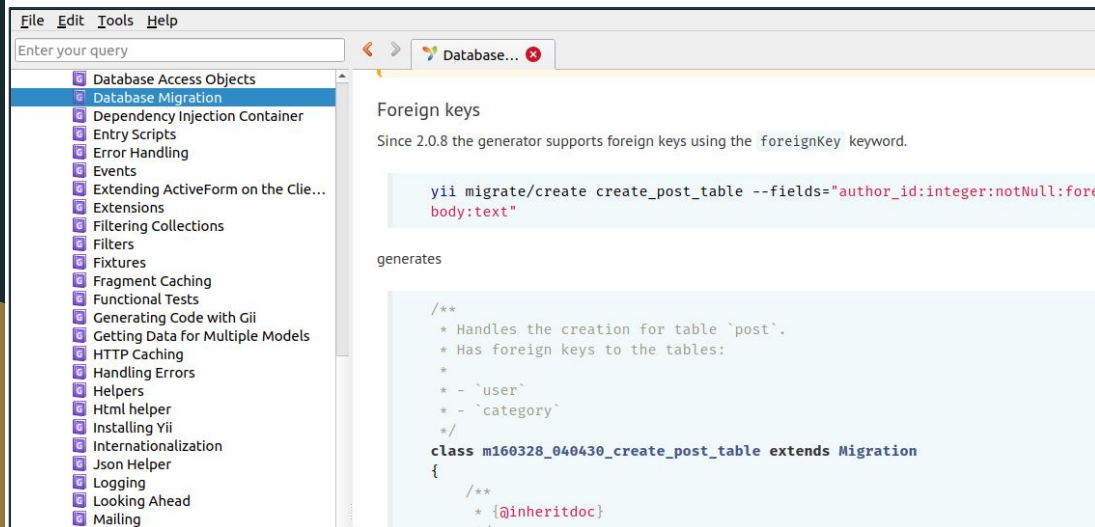
# Создание миграций

php yii migrate/create create\_role\_table



```
m230224_182558_create_role_table.php
no usages
8 class m230224_182558_create_role_table extends Migration
9 {
10     /**
11      * {@inheritdoc}
12      */
13     no usages
14     public function safeUp()
15     {
16         $this->createTable( table: '{{%role}}', [
17             'id' => $this->primaryKey(),
18             'code' => $this->string()->unique()->notNull(),
19             'name' => $this->string()->notNull(),
20         ]);
21
22         $this->insert( table: '{{%role}}', [
23             'code' => 'user', 'name' => 'Зарегистрированный пользователь',
24         ]);
25
26         $this->insert( table: '{{%role}}', [
27             'code' => 'admin', 'name' => 'Администратор'
28         ]);
29     }
30
31     /**
32      * {@inheritdoc}
33      */
34     no usages
35     public function safeDown()
36     {
37         $this->dropTable( table: '{{%role}}');
```

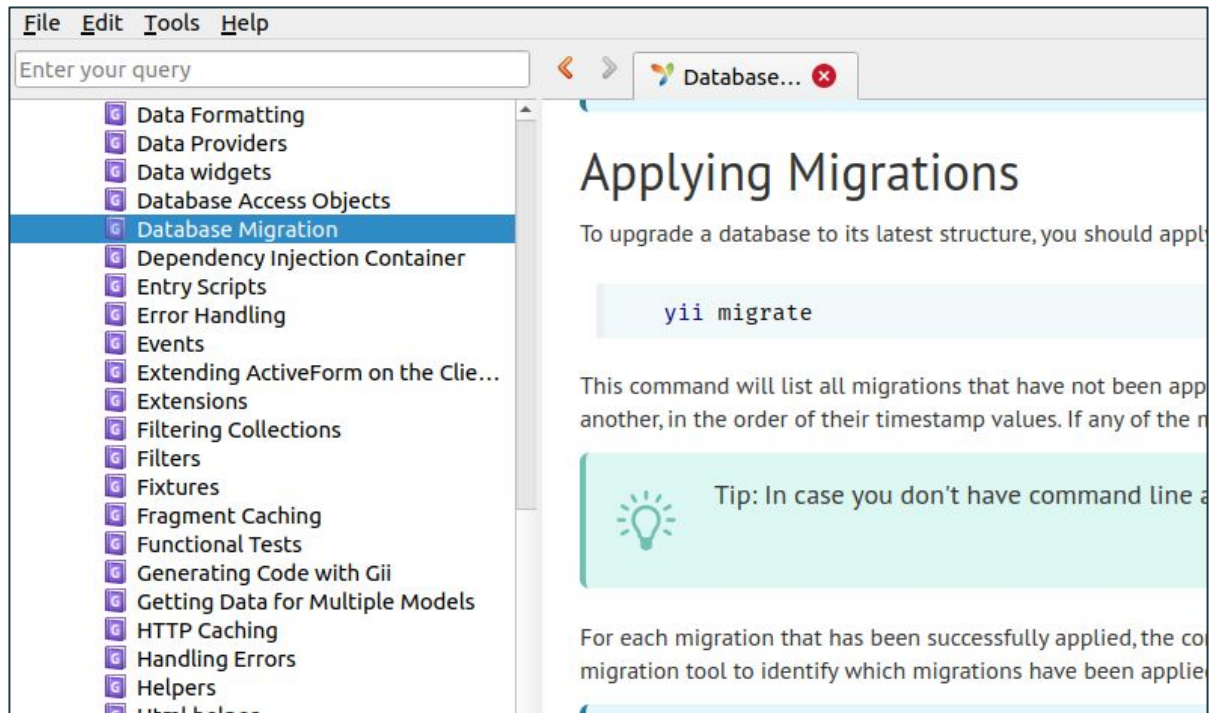
# Внешние ключи



```
m240117_115159_create_user_table.php
16 public function safeUp()
17 {
18     $this->createTable( table: '{{%user}}', [
19         'id' => $this->primaryKey(),
20         'role_id' => $this->integer()->defaultValue( default: 1),
21         'username' => $this->string()->notNull()->unique(),
22         'password' => $this->string()->notNull(),
23         'email' => $this->string()->notNull()->unique(),
24         'first_name' => $this->string()->notNull(),
25         'last_name' => $this->string()->notNull(),
26         'middle_name' => $this->string()->notNull(),
27         'phone' => $this->string()->notNull(),
28     ]);
29     // creates index for column 'role_id'
30     $this->createIndex(
31         name: '{{%idx-user-role_id}}',
32         table: '{{%user}}',
33         columns: 'role_id'
34     );
35     // add foreign key for table '{{%role}}'
36     $this->addForeignKey(
37         name: '{{%fk-user-role_id}}',
38         table: '{{%user}}',
39         columns: 'role_id',
40         refTable: '{{%role}}',
41         refColumns: 'id',
42         delete: 'CASCADE'
43     );
44     $this->insert( table: '{{%user}}', [
45         'role_id' => 2,
46         'username' => 'copp',
47         'password' => md5( string: 'password'),
48         'email' => 'admin@admin.ru',
49         'first_name' => 'Иван',
50         'last_name' => 'Иванов',
51         'middle_name' => 'Иванович',
52         'phone' => '+7(913)-999-99-99',
53     ]);
54 }
```

# Применение миграций

```
php yii migrate
```



The screenshot shows the Yii2 framework documentation page for 'Applying Migrations'. On the left is a navigation menu with various categories, and 'Database Migration' is highlighted. The main content area has the title 'Applying Migrations' and a sub-header 'To upgrade a database to its latest structure, you should apply migrations'. Below this, the command `yii migrate` is shown in a light blue box. A tip box with a lightbulb icon states: 'Tip: In case you don't have command line access, you can use the console command `yii migrate` from the command line.' The bottom part of the page mentions that for each migration that has been successfully applied, the console command `yii migrate` will list all migrations that have not been applied yet, in the order of their timestamp values.

File Edit Tools Help

Enter your query


Database...

## Applying Migrations

To upgrade a database to its latest structure, you should apply migrations.

```
yii migrate
```

This command will list all migrations that have not been applied yet, in the order of their timestamp values. If any of the migrations have been applied, they will be skipped.

 Tip: In case you don't have command line access, you can use the console command `yii migrate` from the command line.

For each migration that has been successfully applied, the console command `yii migrate` will list all migrations that have not been applied yet, in the order of their timestamp values.

# Задание (90 мин)

Разработайте миграции для следующих таблиц:

- Заявка
- Статус заявки