



ПРОЕКТ

Адаптация управления водными ресурсами трансграничных вод бассейна Амударьи к возможным изменениям климата

Отчет по позиции

2.8 Моделирование ASBmm

2.8.2. БД & интерфейс

а) Консультации, клиентская часть

Руководитель проекта, проф.

В.А.Духовный

Ответственный исполнитель
по позиции 2.8

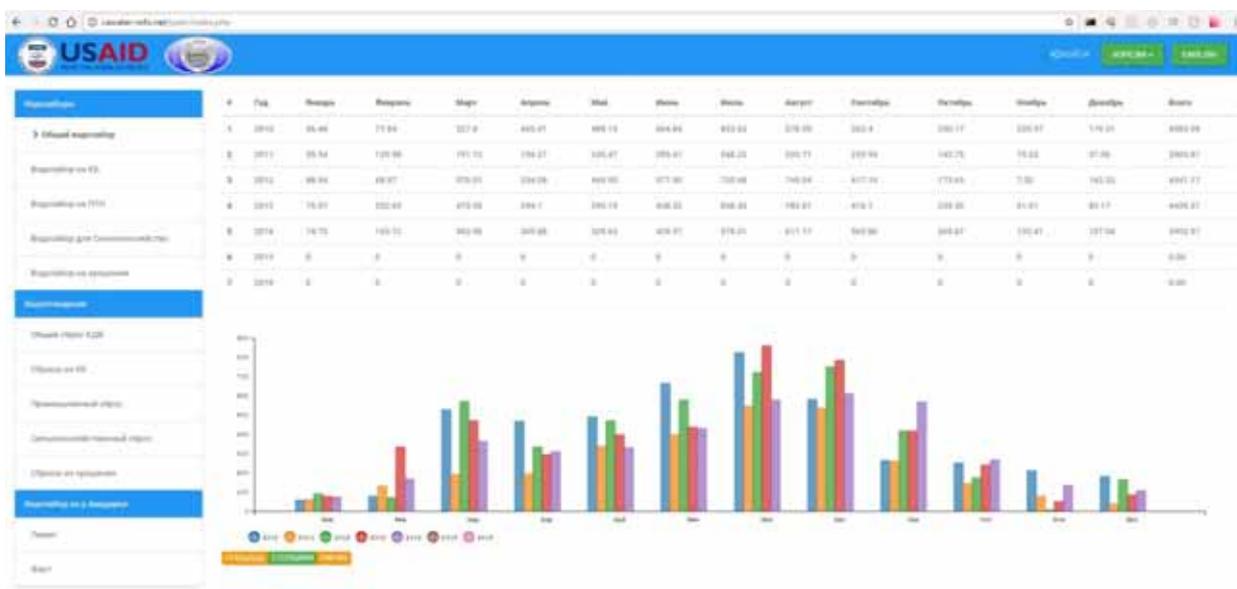
А.Г.Сорокин

Исполнитель работ 2.8.2 а)

Р. Тошпулатов

Ташкент 2016

В рамках проекта PEER создана база данных и интерфейс для водных индикаторов в разрезе зон планирования Узбекистана. На данный момент ресурс расположен на серверах НИЦ МКВК и доступен в интернете по адресу <http://cawater-info.net/peer>



Главная страница

Реализована мультиязычность, на сегодня ресурс поддерживает два языка: **Русский, Английский**. Присутствует возможность ввода данных через веб-интерфейс для авторизованных пользователей. Дизайн сайта выполнен в стиле **MaterialDesign** от компании **Google**. Веб-приложение работает по принципу **SPA** (singlepageapplication), посредством **AJAX**. Система разделена на два приложения: клиентское и серверное. Сервер выступает в качестве **API** реализованный на **Yii**, данные передаются клиентскому приложению в формате **JSON**. Данные хранятся в **СУБД MySQL**.

I. Системные требования к веб-серверу:

- Интерпретатор PHP 5.1 и выше
- Apache HTTP Server
- База данных MySQL 4.1 и выше с поддержкой расширения MySQLi или pdo_mysql.

II. Технические требования к веб-серверу:

- Место на жестком диске – 10мб
- Одноядерный процессор с тактовой частотой 2.4 ГГц

- Оперативная память объёмом 512 Мб

III. Среда разработки ПО:

1) **Yii** - фреймворк, написанный на PHP, и реализующий парадигму MVC. Архитектура и возможности:

- Высокая производительность относительно других фреймворков, написанных на PHP
- Парадигма MVC
- Интерфейсы DAO и ActiveRecord для работы с базами данных (PDO)
- Поддержка интернационализации
- Кэширование страниц и отдельных фрагментов
- Перехват и обработка ошибок
- Ввод и валидация форм
- Аутентификация и авторизация
- Использование AJAX и интеграция с jQuery
- Генерация базового PHP-кода для CRUD-операций (скаффолдинг)
- Поддержка тем оформления для их лёгкой смены
- Возможность подключения сторонних библиотек
- Миграции базы данных
- Автоматическое тестирование
- Поддержка REST

2) **AngularJS** — JavaScript-фреймворк с открытым исходным кодом. Предназначен для разработки одностраничных приложений. Его цель — расширение браузерных приложений на основе **MVC** шаблона, а также упрощение тестирования и разработки. Фреймворк работает с HTML, содержащим дополнительные пользовательские атрибуты, которые описываются директивами, и связывает ввод или вывод области страницы с моделью, представляющей собой обычные переменные **JavaScript**. Значения этих переменных задаются вручную или извлекаются из статических или динамических JSON-данных. **Angular** придерживается **MVC**-шаблона проектирования и поощряет слабую связь между представлением, данными и логикой компонентов. Используя внедрение зависимости, **Angular** переносит на клиентскую сторону такие классические серверные службы, как видовые контроллеры. Следовательно, уменьшается нагрузка на сервер и веб-приложение становится легче.

- 3) **MySQL**- свободная реляционная система управления базами данных. Разработку и поддержку MySQL осуществляет корпорация **Oracle**, получившая права на торговую марку вместе с поглощённой Sun Microsystems, которая ранее приобрела шведскую компанию MySQLAB. Продукт распространяется как под **GNU General Public License**, так и под собственной коммерческой лицензией. Помимо этого, разработчики создают функциональность по заказу лицензионных пользователей. Именно благодаря такому заказу почти в самых ранних версиях появился механизм репликации.

IV. Схема структуры связей между таблицами в СУБД

MySQL:

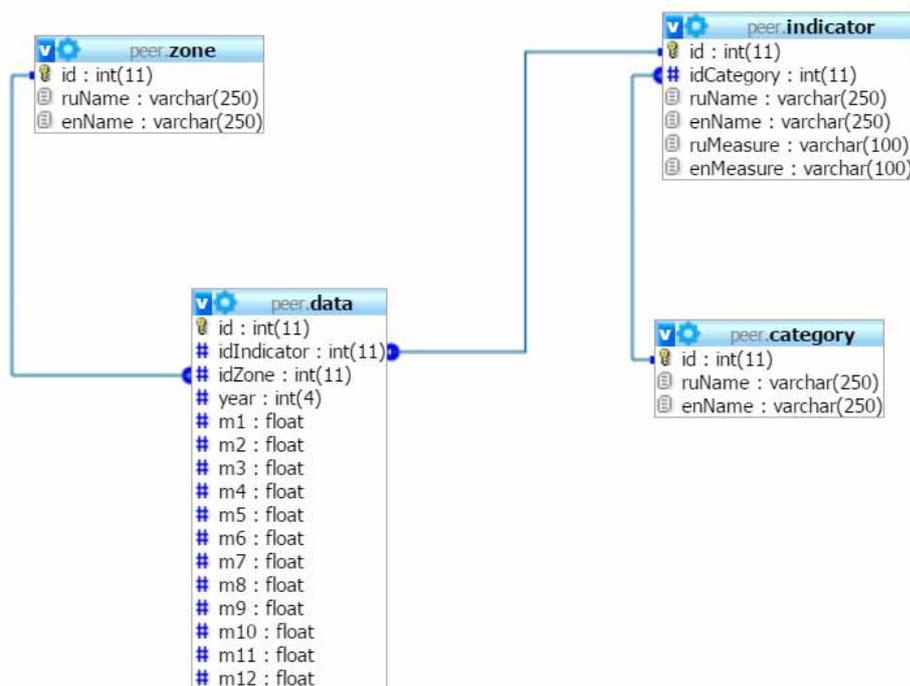


Таблица Zone[id, ruName, enName] - содержит зоны планирования используемые в системе;

Таблица Category[id, ruName, enName] - содержит категории для индикаторов используемые в системе;

Таблица Indicator[id, ruName, enName] - содержит перечень индикаторов используемые в системе;

Таблица Data[id, idIndicator, idZone, year, m1, m2, ...] - содержит данные индикаторов по годам и месяцам в разрезе зон планирований;

V. Структура файловой системы проекта:

assets/ (директория содержит закешированные файлы серверного приложения)

css/ (директория содержит все css-стили, использующие в системе)

fonts/ (директория содержит все шрифты включая иконочные, использующие в системе)

images/ (директория содержит картинки)

ng-modules/ (родительская директория пользовательского приложения)

protected/ (защищённая директория, содержащая компоненты серверного приложения)

themes/ (директория кастомизации интерфейса)

ng-modules– основная директория пользовательского интерфейса, здесь реализована SPA приложение использующая фреймворк **AngularJS**, весь код написан на языке **Javascript** в рамках парадигмы **MVC, SOLID**. Директория разделена на два раздела (app, dependencies). Директория app содержит модели, контроллеры и конфигурационные файлы пользовательского приложения. Директория dependencies содержит зависимости пользовательского приложения. Компоненты настраиваемые и хранятся в файле настроек ng-modules/app/config.js.

```

1 (function () {
2     'use strict';
3
4     /* Constants and Configs that used on the frontend */
5     angular
6         .module('app.config', [])
7         .constant('API', API())
8         .constant('MATH', MATH());
9
10    /* URL to Backend API */
11    function API () {
12
13        var prefix = '/user/index.php';
14
15        var language = "?language=" + window.language;
16
17        return [
18
19            login           : prefix + '/api/login',
20            zones           : prefix + '/api/zones'           + language,
21            zonesCreate     : prefix + '/api/zones/create'   + language,
22            zonesDelete    : prefix + '/api/zones/delete'   + language,
23
24            categories     : prefix + '/api/categories'     + language,
25            categoriesCreate : prefix + '/api/categories/create' + language,
26            categoriesDelete : prefix + '/api/categories/delete' + language,
27
28            indicators     : prefix + '/api/indicators'     + language,
29            indicatorsCreate : prefix + '/api/indicators/create' + language,
30            indicatorsDelete : prefix + '/api/indicators/delete' + language,
31
32            data          : prefix + '/api/data'           + language,
33
34            templates     : prefix + '/site/templates'     + language + '&showList=1'
35
36        ];
37    };
38
39    function MATH () {
40        return {
41            ...
42        };
43    };
44
45 }());

```

конфигурационный файл, раздел настроек запросов к API

```

1 </php>
2
3 class ApiController extends Controller
4 {
5     /* GET */
6     public function actionZones($indicator, $year)
7     {
8         echo json_encode(Zone::model()->getAll($indicator, $year));
9     }
10
11     public function actionCategories()
12     {
13         echo json_encode(Category::model()->getAll());
14     }
15
16     public function actionIndicators()
17     {
18         echo json_encode(Indicator::model()->getAll());
19     }
20
21     public function actionData()
22     {
23         echo json_encode(Data::model()->getAll());
24     }
25
26     /* POST */
27     public function actionLogin()
28     {
29         $model = new LoginForm();
30         $model->attributes = $_POST;
31         if ($model->validate() && $model->login()) echo $this->success();
32     }
33
34     public function actionZoneCreate()
35     {
36         $model = new Zone();
37         $model->attributes = $_POST;
38         if ($model->save()) echo json_encode($this->array($model, array("id", "name")));
39     }
40
41     public function actionZoneDelete($id)
42     {
43         if (Zone::model()->findById($id)->delete()) echo $this->success();
44     }
45 }

```

ApiController являющийся посредником между моделями и запросами со стороны пользователя

VI. Интерфейс

Главная страница содержит **шапку** сайта в которой расположены: меню с возможностью выбора зон планирования, смены языка, авторизации и для авторизованного пользователя кнопка добавления новых зон планирования.





+ ДОБАВИТЬ КАТЕГОРИЮ

Водозаборы + -

- Общий водозабор -
- Водозабор на КБ -

#	Год	Январь
1	2010	56.48
2	2011	59.54
3	2012	88.94
4	2013	75.97

Меню слева содержит список индикаторов. Также кнопки редактирования, добавления и удаления категорий и индикаторов для авторизованных в системе пользователей

#	Год	Январь	Февраль	Март	Апрель	Май	Июнь	Июль	Август	Сентябрь	Октябрь	Ноябрь	Декабрь	Всего
1	2010	56.48	77.84	527.6	465.41	485.35	664.84	823.63	578.36	282.4	250.17	289.97	178.31	4585.59
2	2011	59.54	129.96	191.16	194.27	335.47	395.47	548.28	595.71	339.94	143.75	75.23	37.88	2965.87
3	2012	88.94	68.87	570.91	334.08	468.98	577.91	730.98	748.04	417.78	179.85	7.92	168.32	4341.17
4	2013	75.97	332.68	479.93	294.1	395.75	638.32	898.42	793.81	475.1	229.52	81.81	83.17	4433.37
5	2014	74.78	168.72	369.96	309.84	328.62	426.37	578.91	471.77	595.96	263.87	132.41	187.04	3932.97
6	2015	0	0	0	0	0	0	0	0	0	0	0	0	0.00
7	2016	0	0	0	0	0	0	0	0	0	0	0	0	0.00

Под шапкой сайта, расположена таблица с выводом данных по выбранному индикатору, зоны планирования, годам и месяцам. Также график, с возможностью переключения вида отображения информации.

График вида "Столбики"

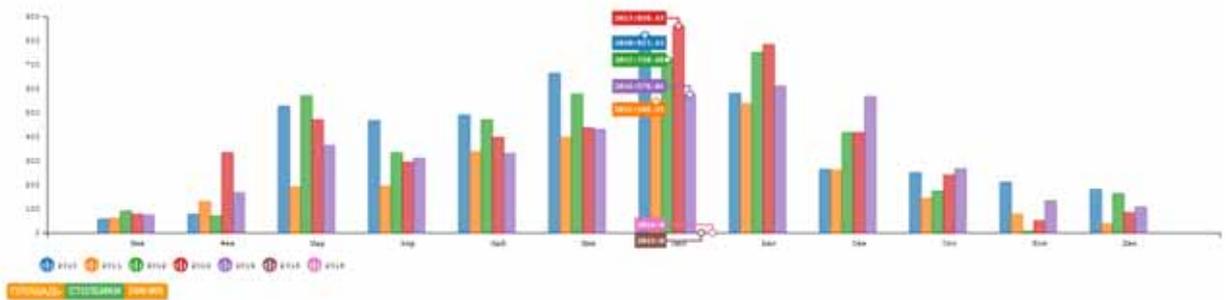


График вида "Площадь"

