

**Содержание**

[Введение 3](#_Toc483410478)

[1. Сервер 4](#_Toc483410479)

[1.1. Установка локального web-сервера 4](#_Toc483410480)

[1.2. Установка YII2 6](#_Toc483410481)

[2. База данных 7](#_Toc483410482)

[2.1 Концептуальная модель базы данных 7](#_Toc483410483)

[2.2 Физическая модель базы данных 8](#_Toc483410484)

[3. Панель администрирования 9](#_Toc483410485)

[3.1. GII 9](#_Toc483410486)

[3.2. Доработка CRUD 13](#_Toc483410487)

[4. Сайт 14](#_Toc483410488)

[4.1. Реализованные страницы 14](#_Toc483410489)

[Заключение 16](#_Toc483410490)

[Литература 17](#_Toc483410491)

# Введение

**Цели**

Получить опыт и навыки работы в рамках Группового Проектного Обучения. Освоить навыки создания и продвижения информационных веб-систем.

**Задачи**

Спроектировать и реализовать веб-ориентированную информационную систему «Облачная Платформа Ведения Реестра Инновационных Проектов», создать план продвижения информационной системы в условиях рынка.

**Актуальность**

ТУСУР является одним из ведущих вузов в области инновационных разработок и их продвижении. Отдел коммерциализации разработок ТУСУРа занимается привлечением инвесторов для реализации проектов разрабатываемых в университете, в том числе по программе ГПО.

Основной задачей нашего проекта является создание реестра проектов, который заменит текущую нуждающуюся в обновлении систему ГПО для отдела коммерциализации разработок ТУСУРа.

ТУСУР поддерживает новые технологические веяния как минимум в связи со статусом и как максимум — в связи с высокой образованностью его студентов и преподавателей, а потому его ресурсы, такие как сайт и подсайты, динамично изменяются. Мы хотим внести свой вклад посредством создания рубрикатора, содержащего научно-исследовательские проекты по классификаторам с целью привлечения инвесторов и их последующей коммерциализация.

# Организация и делегирование задач

* 1. Установка локального web-сервера

В качестве локального web-сервера был выбран OpenServer. Он является бесплатным, не требует больших ресурсов, легко конфигурируется. Была выбрана конфигурация Apache-2.4, PHP-5.6, PostgreSQL-9.5.

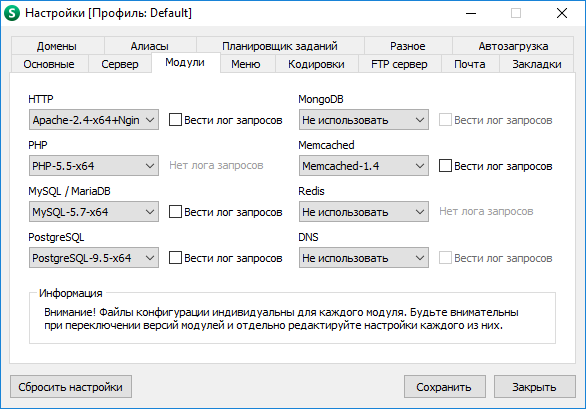


Рисунок 1 – “Конфигурация сервера”

Затем был создан каталог с системой и прописан путь к домену.

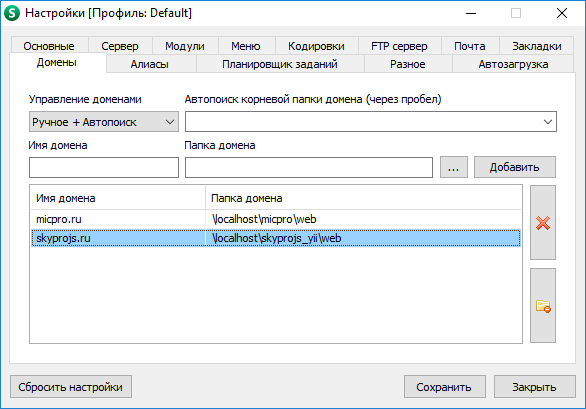


Рисунок 2 – “Путь до папки с системой”

* 1. Установка YII2

Для установки был использован Composer - менеджер зависимостей. Был загружен и установлен фреймворк yii2 последней версии.

1. База данных

2.1 Концептуальная модель базы данных

Для проектирования концептуальной модели данных было использовано средство автоматизированного проектирования Sybase Power Designer. Данное средство проектирование обладает следующим рядом преимуществ:

* большое количество вариантов представления моделей данных;
* возможность генерирования базы данных для различных систем управления базами данных (СУБД), в частности поддержка используемой в данном проекте СУБД PostgreSQL;
* интуитивно-понятный интерфейс;
* поддержка репозитория для коллективной разработки.

В ходе проектирования концептуальной модели данных ОПВРИП были выявлены следующие сущности:

* Направления
* ГРНТИ
* УДК
* Подразделения
* Проект
* Администратор
* Участник

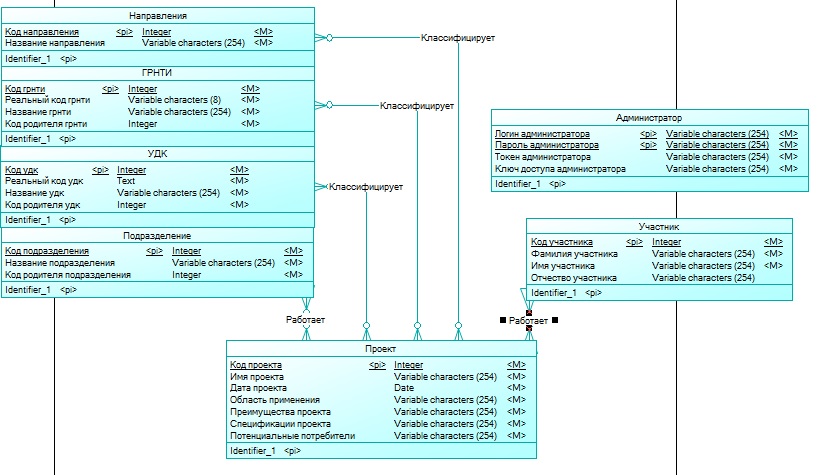
****

Рисунок 3 - "Концептуальная модель базы данных"

* 1. Физическая модель базы данных

На основе концептуальной модели базы данных была сгенерирована физическая модель данных средствами Sybase PowerDesigner.

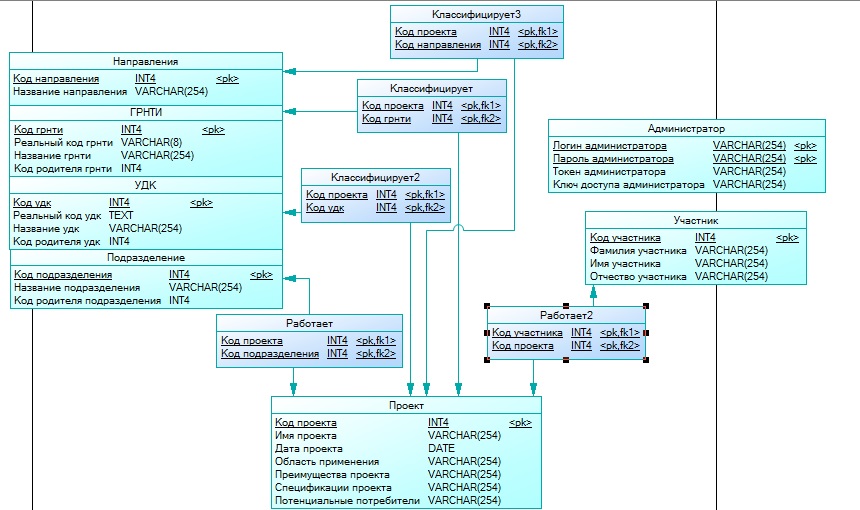


Рисунок 4 - "Физическая модель базы данных"

1. Панель администрирования
   1. GII

Для создания панели администрирования был использован GII - встроенный модуль YII.

Это средство позволяет генерировать модели, модули и CRUD (create, read, update, delete) контроллер и представление.

Сначала были сгенерированы модели для каждой сущности системы.

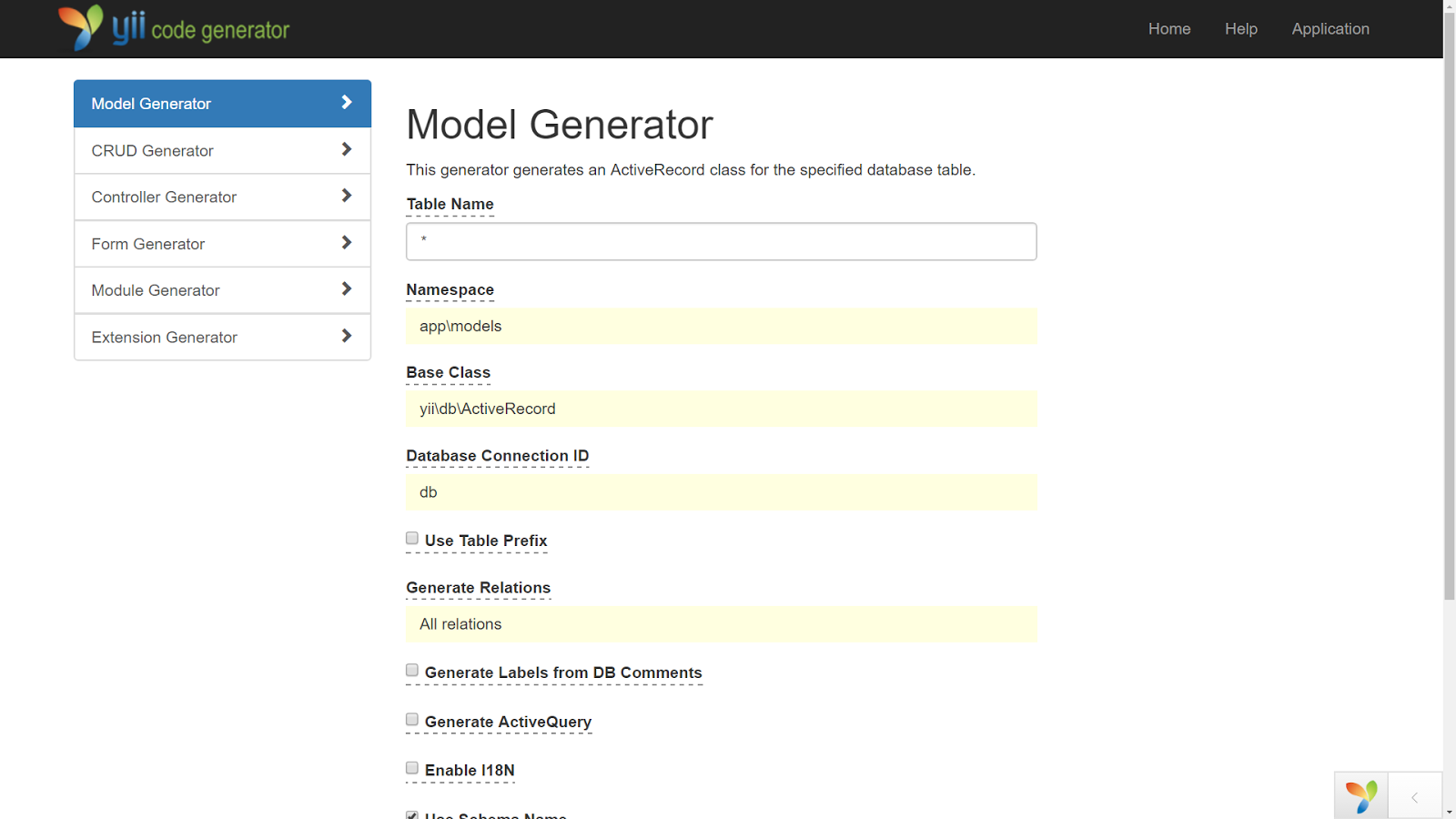


Рисунок 5 – “Генерация моделей”

Затем был сгенерирован модуль “admin” для будущей панели администрирования.

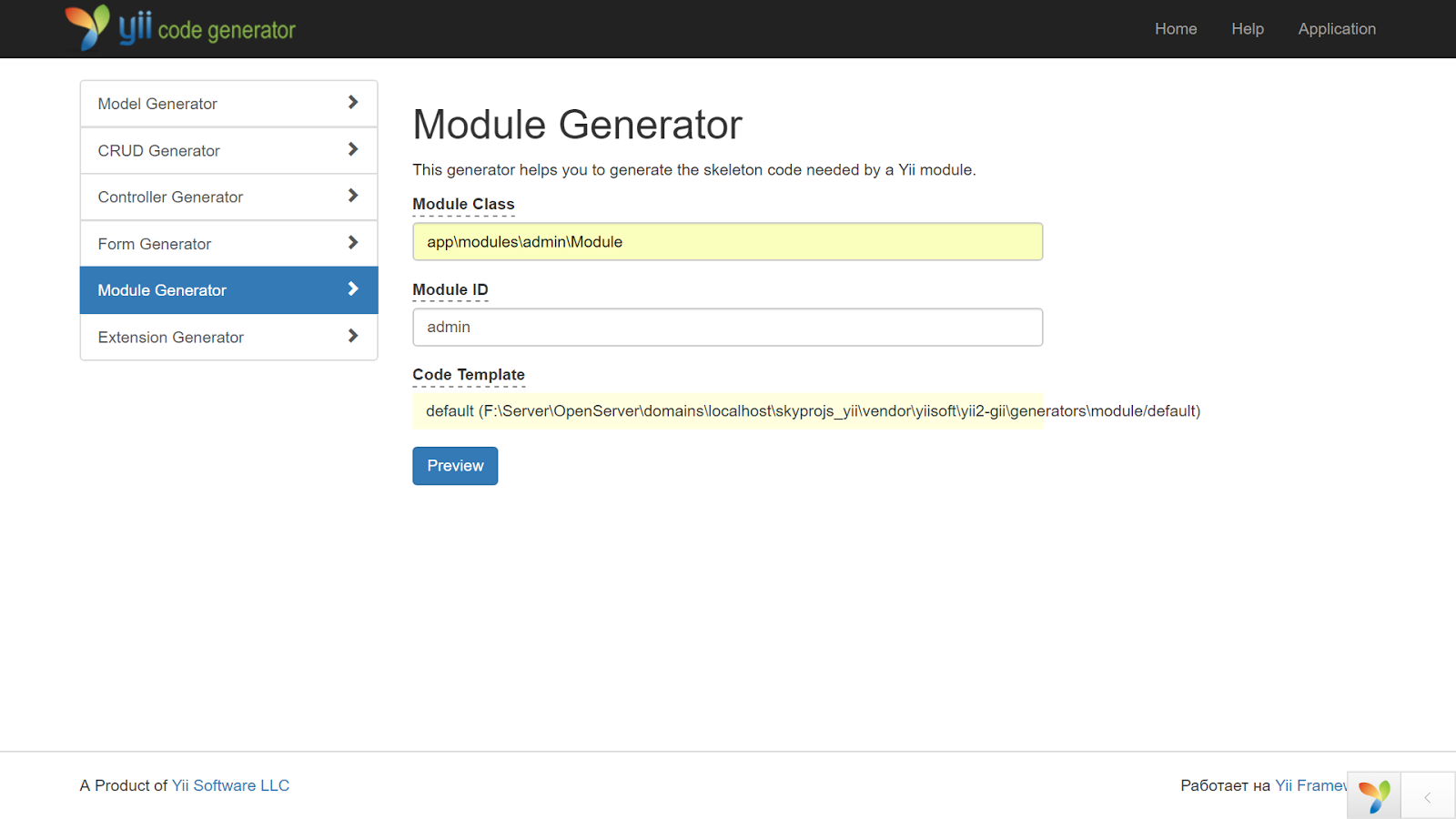


Рисунок 6 – “Генерация модулей”

Для каждой сущности в бд были сгенерированы CRUD:

1. Проекты
2. Подразделения
3. Направления
4. УДК
5. ГРНТИ

Генерация CRUD требует наличия сгенерированной модели для сущности.

CRUD представляет из себя:

* Контроллер
* Представления
* Модель поиска для сущности

В контроллере прописаны действия, каждое из которых является представлением для:

1. Вывода списков

Постранично выводит все строки из таблицы в бд.

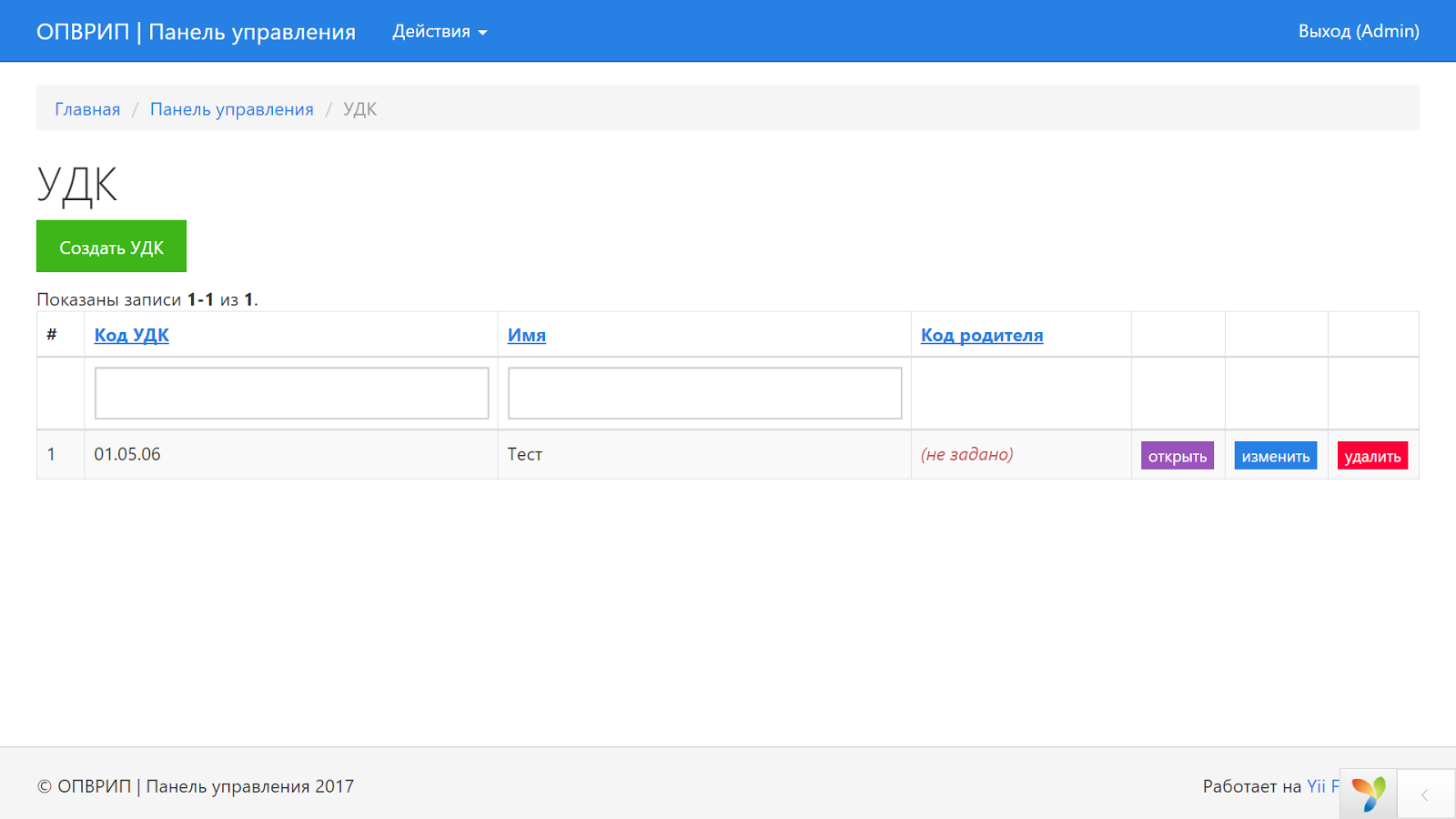


Рисунок 7 – “Пример списка записей из таблицы бд”

1. Добавления данных

Представляет из себя форму для занесения данных в таблицу.

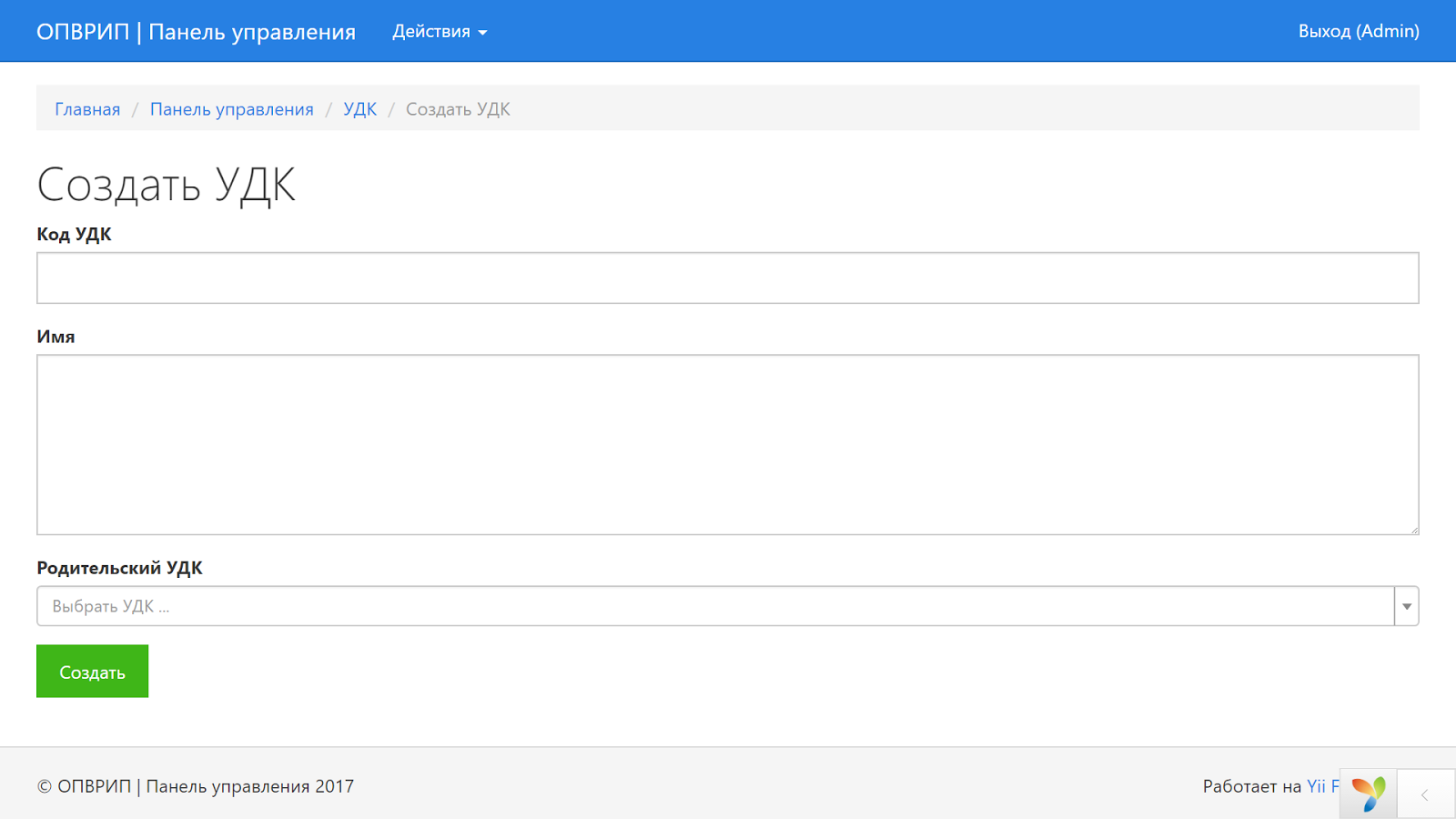


Рисунок 8 – “Пример добавления записи в бд”

1. Просмотра данных

Выводит информацию о выбранной записи из таблицы бд.

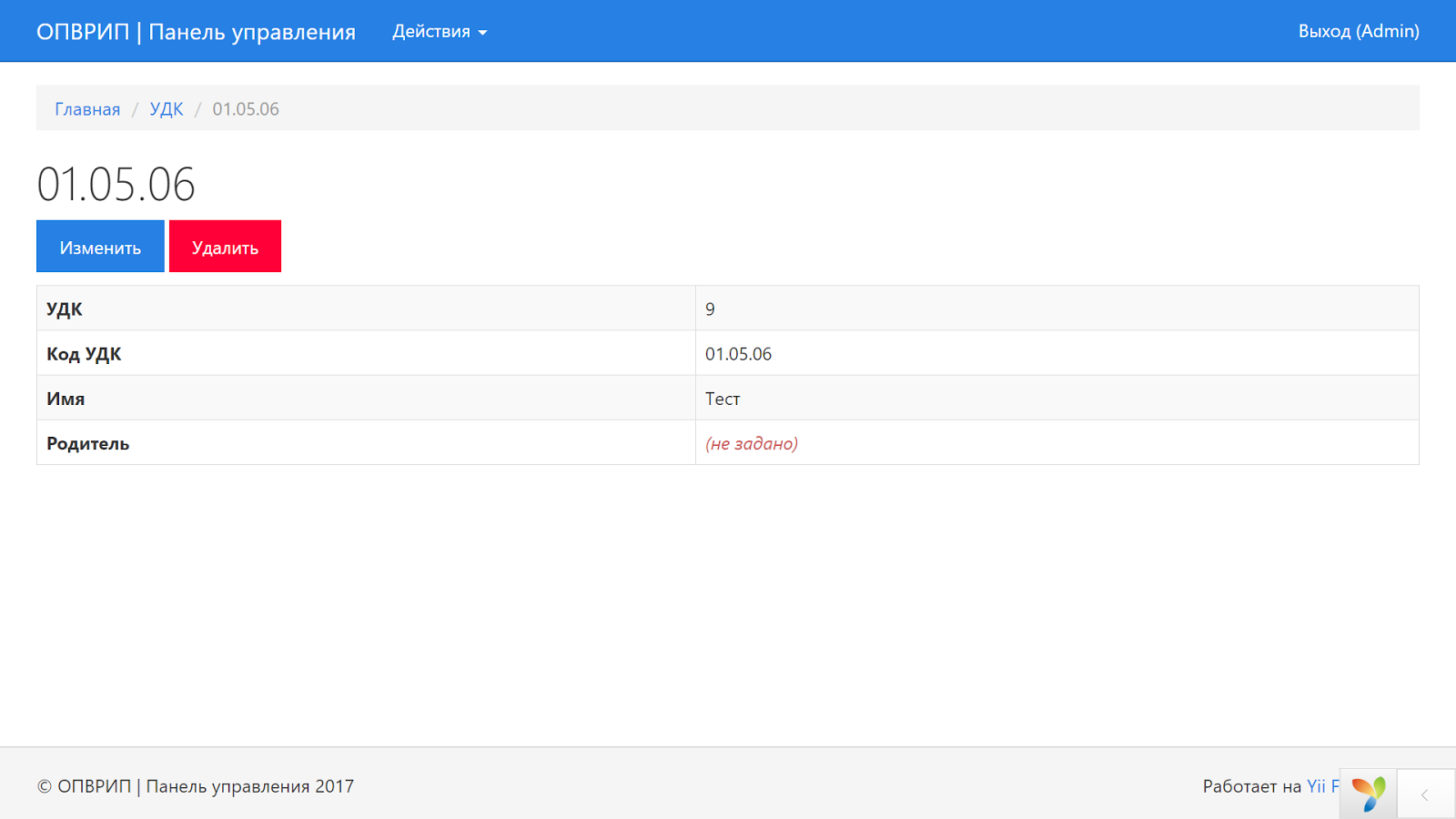


Рисунок 9 – “Пример просмотра записи из бд”

1. Редактирования данных

Представляет из себя форму для редактирования данных какой-либо записи из таблицы бд.

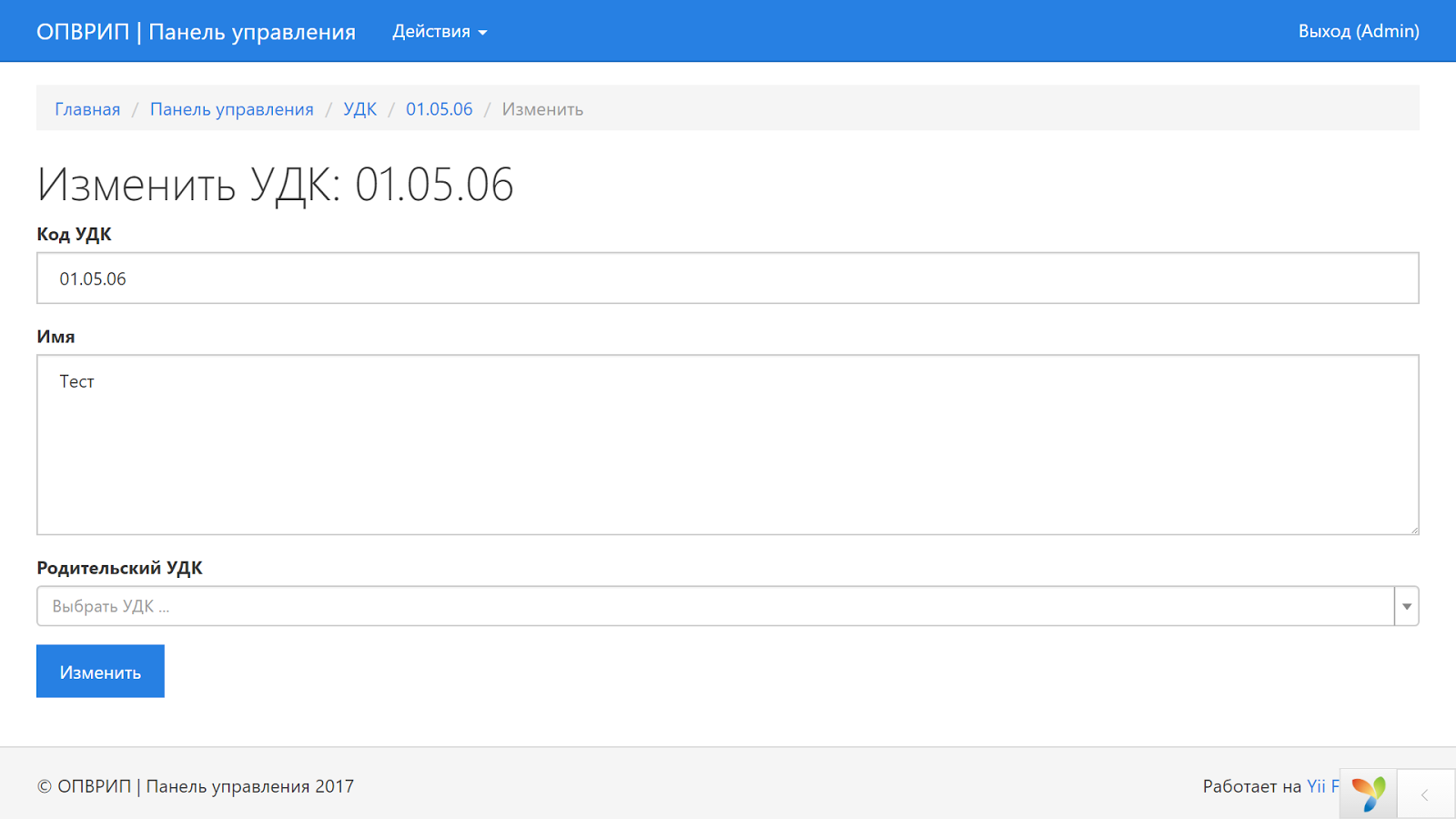


Рисунок 10 – “Пример редактирования записи из бд”

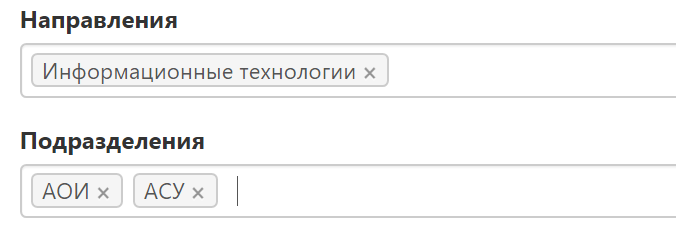
1. Удаления данных

Удаляет запись из таблицы в бд.

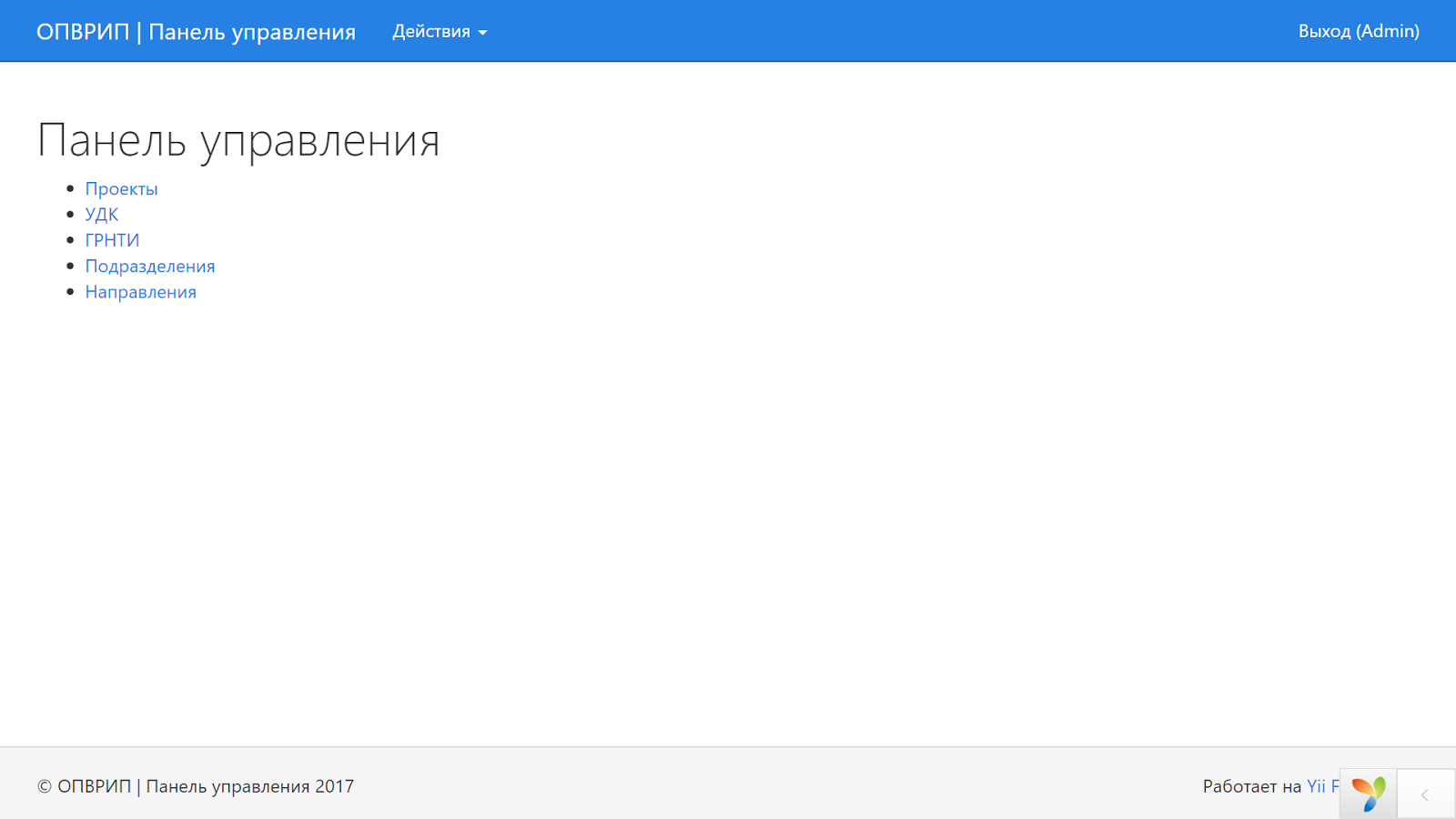
* 1. Доработка CRUD

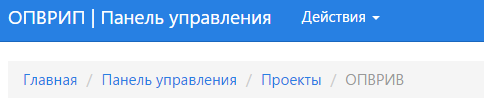
В дальнейшем сгенерированные представления CRUD были переработаны. Потребовалось их русифицировать, дополнить функционалом для управления промежуточными связями в моделях, каскадным удалением, сформировать собственный вывод список через GridView.

Через Composer была произведена установка виджета Select2 для добавления списков реализующих связь много-ко-многим через промежуточные таблицы.



На последнем этапе была создана навигация в представлении модуля admin и breadcrumbs - виджете, позволяющем перемещаться по иерархии сайта.

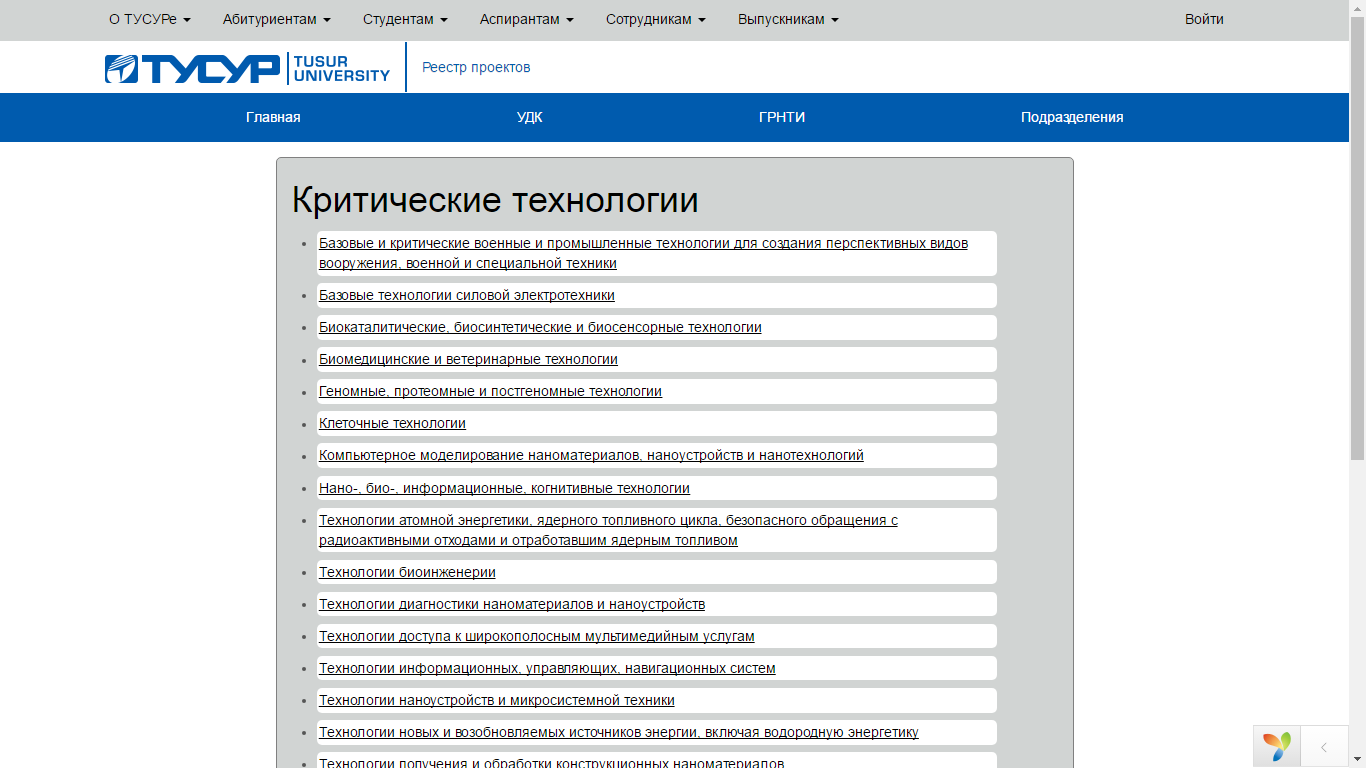




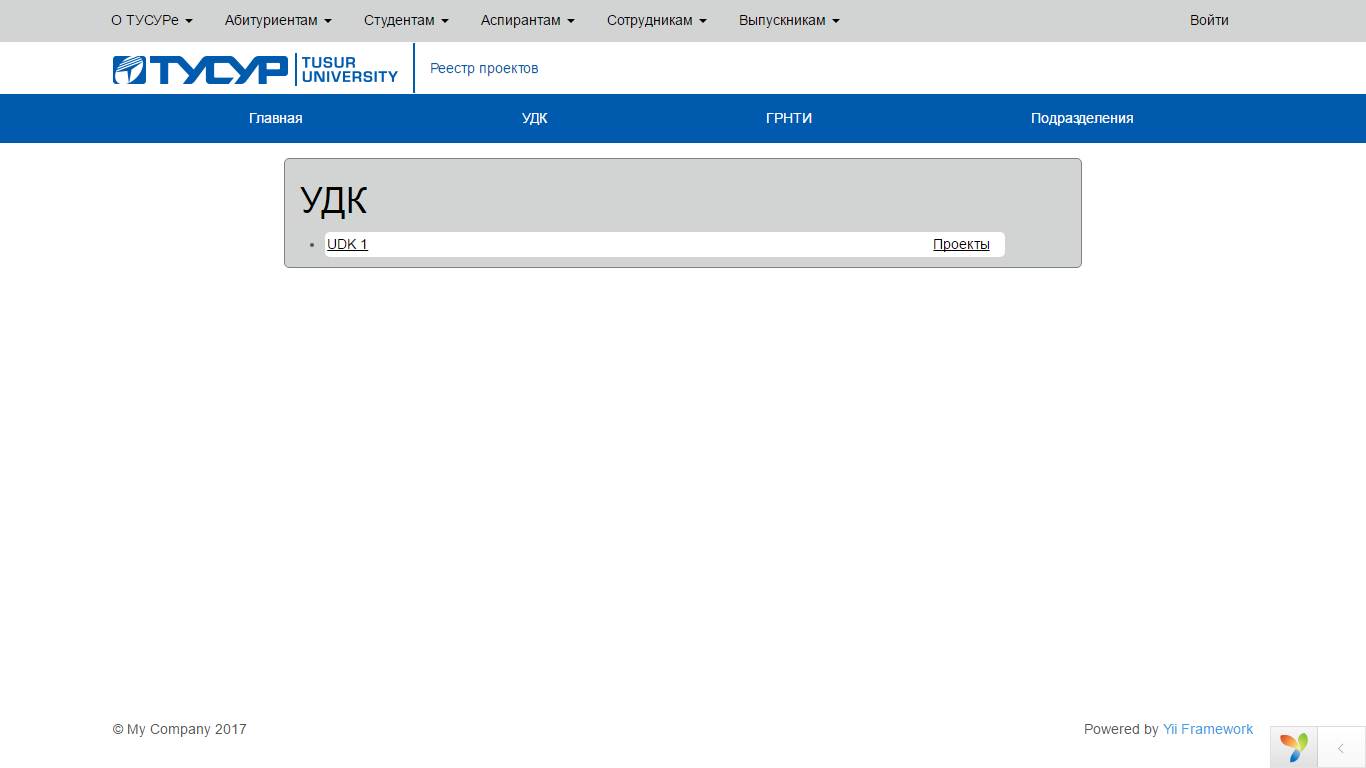
1. Сайт
   1. Реализованные страницы

При помощи сгенерированных ранее моделей были реализованы следующие страницы:

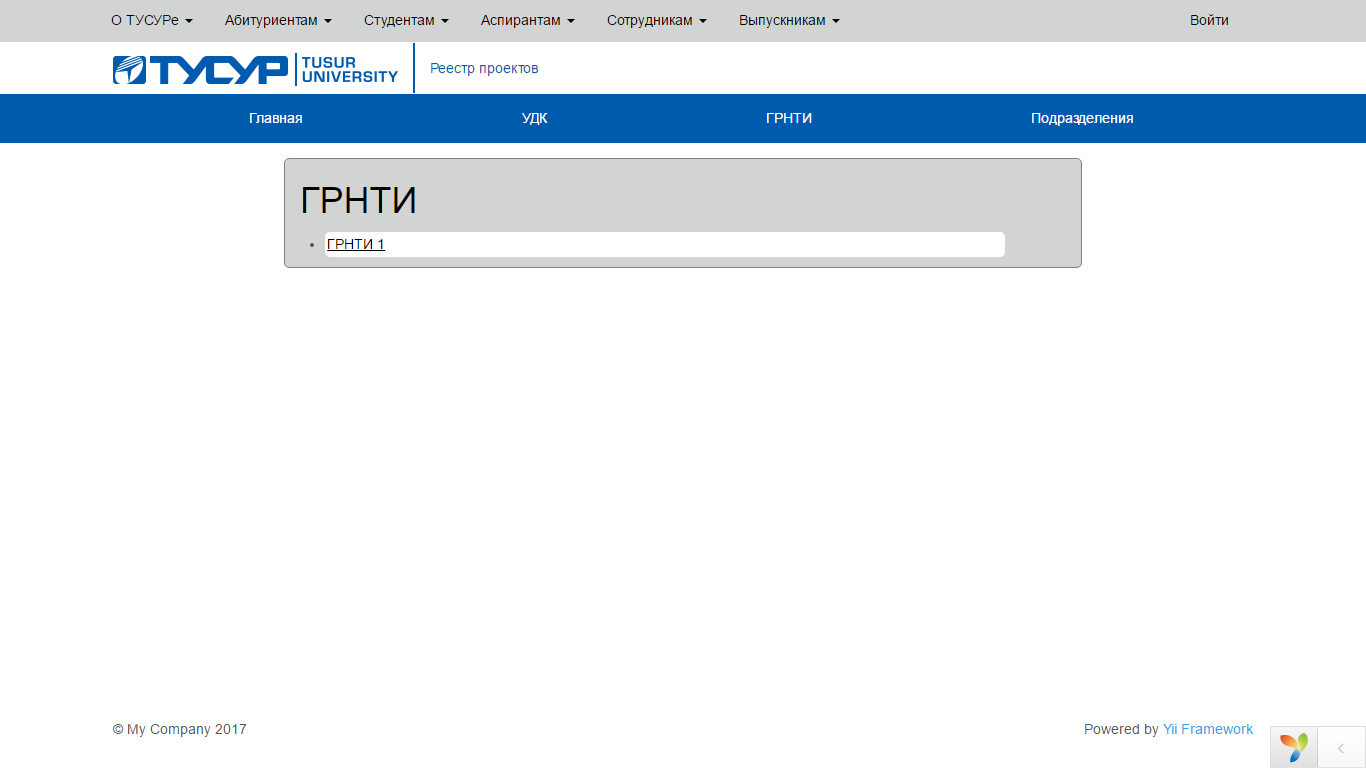
* Главная. На ней расположен перечень критических технологий со ссылками на проекты.



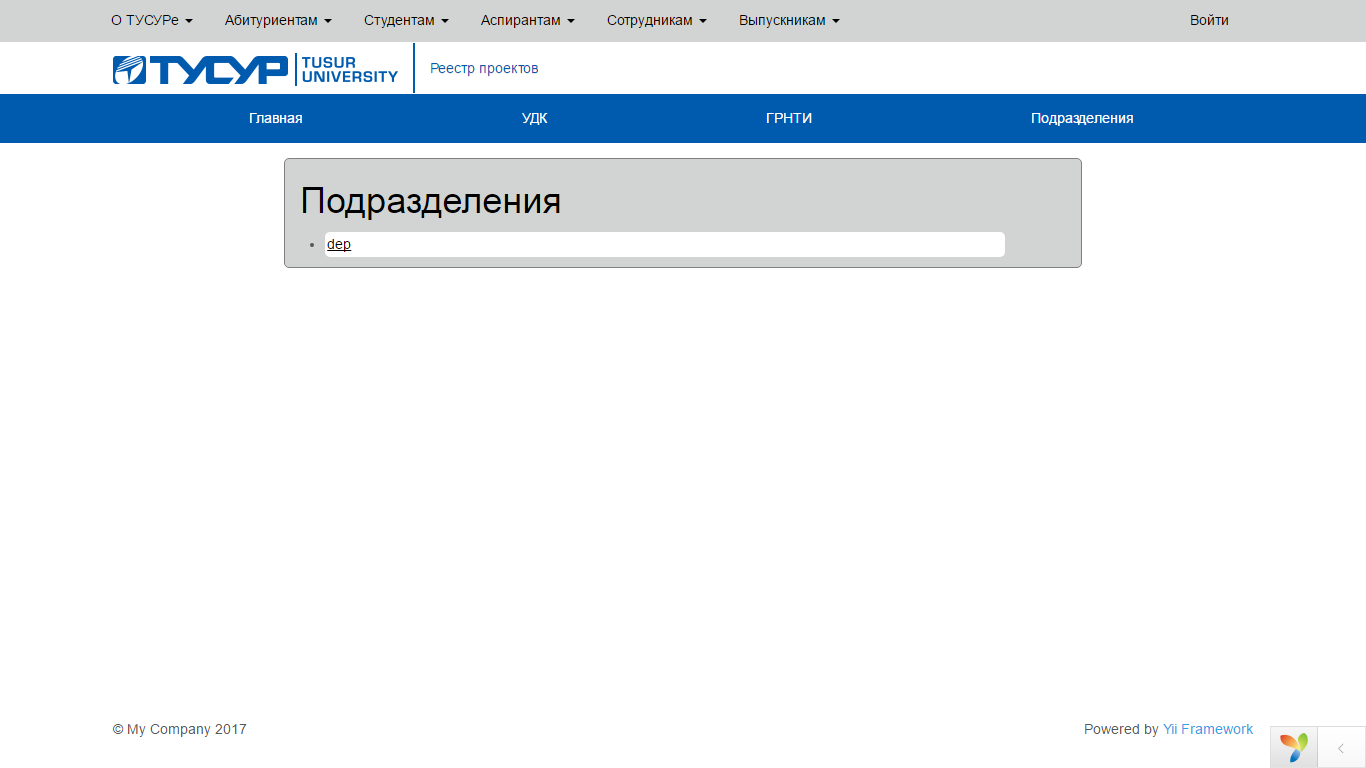
* УДК. На ней расположены классификаторы УДК со ссылками на проекты.



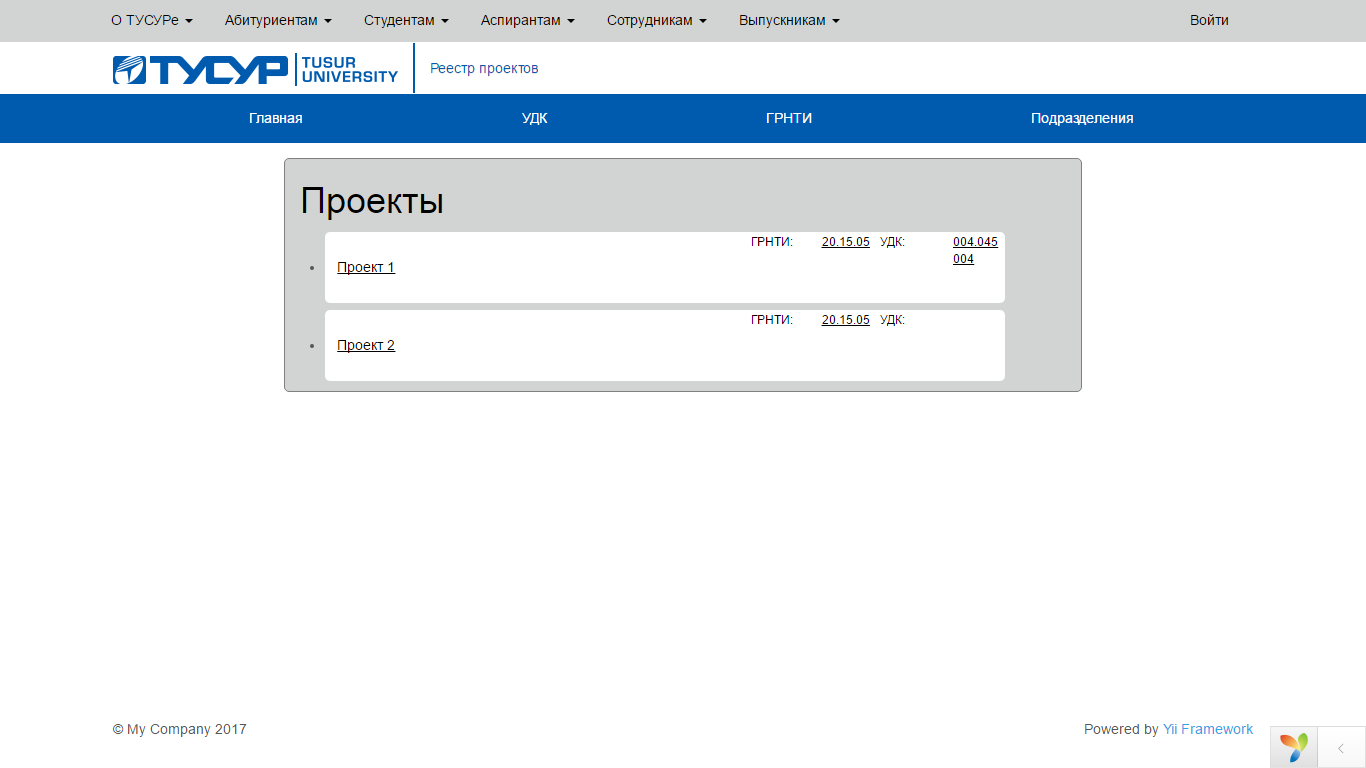
* ГРНТИ. На ней расположены классификаторы ГРНТИ со ссылками на проекты.



* Подразделения. На ней расположены подразделения со ссылками на проекты.



* Страница с проектами.



# Заключение

Проект был представлен на международной научно-практической конференции. В ходе работ был разработана панель администрирования, web-сайт системы. За время работы были достигнуты успехи в использовании фреймворков для быстрой веб-разработки, изучены принципы работы с классами, паттернами на PHP, оптимизация запросов в базу данных «PostgreSQL».

Облачная платформа ведения реестра информационных проектов, имеет упрощенный функционал, что существенно позволяет сократить временные затраты. Также система удобна в обслуживании, и легка при освоении. Гибкая система доступа, основанная на ролях. Определяет чёткие и понятные для пользователей компьютерной системы правила разграничения доступа, позволяет реализовать гибкие, изменяющиеся динамически в процессе функционирования компьютерной системы правила разграничения доступа

Основной экономический эффект достигается за счет повышения эффективности и качества управления организации, а также за счёт предупреждения (исключения) всевозможных аварийных ситуаций, связанных с некачественным обслуживанием оборудования или программного обеспечения.

# Литература

* 1. Д.Н. Колисниченко. Самоучитель PHP5, 2004 – 578 с.
  2. “Справочный ресурс веб-разработчика.” – Режим доступа: //http:// http://htmlbook.ru.
  3. “Онлайн руководство по разработке на PHP.” – Режим доступа: //http:// http://php.net.
  4. “Онлайн руководство по PostgreSQL.” – Режим доступа: //http://

1. https://www.postgresql.org/docs/9.5/static/index.html.
   1. ГОСТ Р ИСО/МЭК – 15910 Информационная технология. Процесс создания документации пользователя программного средства. – Режим доступа: //http://lib.rus.ec/
   2. Нартова А. PowerDesigner 15. Моделирование данных, 2012 – 480с.