



ПРИНЦИПЫ И АЛГОРИТМЫ ПОСТРОЕНИЯ ИНТЕГРАЦИОННЫХ ПЛАТФОРМ

2020



By Subject Education



Интеграционная платформа

Интеграция – процесс объединения частей в целое

Веб-интеграция – объединение разнородных частей и систем в единую среду на базе веб.

Интеграция данных – объединение данных, находящихся в различных источниках, и предоставление данных пользователям в унифицированном виде.

Цифровая платформа – это система алгоритмизированных взаимовыгодных взаимоотношений участников сферы деятельности, осуществляемых в единой информационной среде





Концептуальные требования к разработке интеграционной платформы

Требования к разработке интеграционной платформы – это исходные данные и принципы, на основании которых проектируются и создаются информационные системы, в том числе интеграционные платформы [3]:

- 1) организовать единую платформу интеграции данных предприятий и учебных заведений, представляющую собой отдельное приложение или набор приложений и обеспечивающую взаимодействие всех задействованных в интеграции информационных систем (ИС);
- 2) использовать для хранения информации об интеграционных процессах и метамодели данных предметной области универсальную систему управления базами данных;
- 3) наличие набора инструментальных средств, обеспечивающих создание и модификацию метамодели предметной области (МПД);
- 4) для каждой ИС создать адаптер, обеспечивающий взаимодействие с ядром интеграционной платформы (прием и передачу данных);
- 5) адаптер должен обладать графическим пользовательским интерфейсом для ручного ввода данных, контроля или инициирования интеграции данных (мастер интеграции данных);
- 6) наличие универсального API для взаимодействия адаптеров ИС с ядром интеграционной платформы.



Принципы создания интеграционной платформы

Создание интеграционной платформы включает следующие **принципы** [3]:

- 1) унифицированного федеративного* доступа к данным, согласно которому приложение, являющееся потребителем или источником данных, должно взаимодействовать с единым сервером федерализации данных;
- 2) интеграции данных разнообразных информационных систем (ИС) предприятий и учебных заведений на основе информационной метамоделей предметной области (МПД). Использование МПД предметной области позволяет сделать процесс обмена данными между ИС простым и прозрачным: этот подход гарантирует, что все сообщения между ИС будут правильно поняты и интерпретированы;
- 3) эффективности масштабирования интеграционной платформы благодаря наличию развитых инструментальных средств;
- 4) разделения описания логики процессов передачи данных и описания параметров доступа к данным конкретных интегрируемых ИС. Согласно этому принципу для каждой ИС, задействованной в процессе интеграции, необходим адаптер, предоставляющий интеграционной платформе интерфейс к данным этой ИС. Адаптер должен преобразовывать информацию из схемы данных интегрируемой ИС в схему интеграционной платформы.

**Федерализация данных обеспечивает единую виртуальную картину нескольких первичных источников данных. Для получения сведений о некотором процессе, обрабатываемом в нескольких оперативных приложениях, процессор федерализации данных извлекает данные из соответствующих первичных складов данных, интегрирует их таким образом, чтобы они отвечали виртуальной картине и требованиям запроса, и отправляет результаты бизнес приложению, от которого пришел запрос.*



- 5) сервисной ориентированности архитектуры платформы: транспортный уровень движения данных интегрируемых ИС должен быть организован посредством Web сервисов, а также представление данных должно быть реализовано в формате XML. Принципы сервисноориентированной архитектуры (Service Oriented Architecture, SOA) позволяют обращаться не только к данным, хранящимся в базах данных, но и в коммерческих и заказных приложениях, Web контенте, документах, рисунках, и пр. Использование SOA в качестве основы транспортного уровня платформы интеграции поддерживает сбор и передачу данных из структурированных и неструктурированных транзакционных систем;
- 6) защищенности, согласно которому платформа интеграции должна обладать развитыми механизмами защиты передаваемых данных, а также уметь адаптироваться к различным архитектурам безопасности интегрируемых ИС.

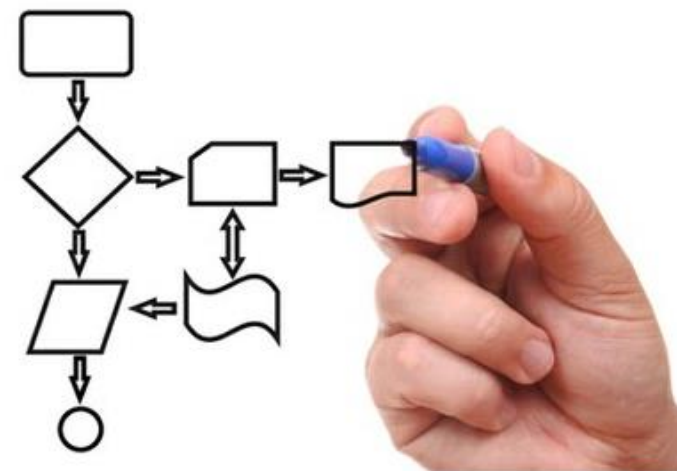


Этапы разработки интеграционной платформы

На практике интеграцией платформ занимаются разработчики программного обеспечения. Выделяют следующие стадии разработки:

- **Предпроектная:** выявление требований и определение концепции
- **Проектирование:** разработка архитектуры платформы и ее отдельных представлений, программная реализация
- **Последующие стадии** жизненного цикла: тестирование, внедрение, сопровождение платформы

Для заказчиков самой важной является предпроектная стадия





Рекомендации по разработке технического задания по построению интеграционной платформы

Техническое задание – основной документ, содержащий требования заказчика к системе, в соответствии с которыми осуществляется создание и разработка конечного продукта.

Техническое задание необходимо

Заказчику, чтобы

- понять, что ему необходимо;
- принять конечный продукт в соответствии с требованиями технического задания.

Исполнителю, чтобы

- понять и усвоить поставленную задачу;
- грамотно спланировать ресурсы;
- избежать излишней работы над проектом.

Конечному потребителю, чтобы

- получить удовольствие от пользования качественным продуктом.



Примерная структура технического задания

Общие сведения о проекте и требования

Обоснование работ
Общее описание проекта
Требования к формату предоставляемых результатов
Технические требования
Верстка
Программное обеспечение
SEO-оптимизация и ЧПУ
Языковые версии
Отображение для разных регионов
Требования к системе администрирования

Общие блоки и сервисы

Header (шапка сайта)
Footer (подвал сайта)
Всплывающие формы
Заказать исследование
Откликнуться на вакансию
Хлебные крошки

Главная страница

Каталог услуг

Шаблон страницы “Услуга”

О компании

Клиенты и благодарности

Контакты

Научная деятельность и новости

Листинг статей
Страница статьи

Блог

Листинг статей
Страница статьи

ГОСТы

О нас пишут

Вакансии

Кейсы

Листинг кейсов
Страница кейса

Вспомогательные и сервисные страницы

Текстовая страница
Простой шаблон
Шаблон с целевым действием
Страница ошибки 404



Примерная структура технического задания

2. Общие блоки и сервисы

2.1. Header (шапка сайта)

- **Логотип организации**
[Кликабелен](#), ведет на главную страницу сайта.
- **Основное навигационное меню**...
Собирается вручную. Может иметь вложенность ([max](#))
- **Дополнительное навигационное меню**...
Собирается вручную в CMS сайта. Может иметь вложенности).
- **Контакты**
Почта (ссылка), соц. сети ([facebook](#), ссылка), номер телефона, номер телефона из модуля Контакты, который редактируется вручную. Телефон [кликабелен](#) на мобильных устройствах.
- **Поиск**
[Поиск](#) производится по следующим разделам сайта:
 - Каталог услуг
Поля для поиска: наименование услуги, аннотация,
 - Блог
Поля для поиска: наименование статьи, аннотация,
 - Научная деятельность
Поля для поиска: наименование статьи, аннотация,
 - Новости
Поля для поиска: наименование новости, аннотация,
 - Кейсы
Поля для поиска: наименование кейса, текст
 Результаты поиска отображаются на [специальной странице](#).

2.2. Footer (подвал сайта)

- **Логотип организации**
[Кликабелен](#), ведет на главную страницу сайта.

4.1. Шаблон страницы “Услуга”

Страница состоит из блоков, некоторые из которых являются обязательными, остальные скрываются, если не заполнены данными в CMS.

* обязательные блоки и элементы

- **Первый экран***
 - Название услуги*
 - Аннотация*
 - **Лидовая кнопка***
Текст кнопки задается вручную в CMS.
 - Преимущества компании*...
Такие же, как на [главной странице](#).
 - Тематическое изображение*
- **Ссылки на внутренние страницы**
Дочерние категории услуг
- **Текст-описание услуги***
- **Фото**...
Из библиотеки медиафайлов. Увеличиваются для просмотра.
- **Случай из практики**
Карусель. Добавляются кейсы из модуля [Кейсы](#).
- **Отзывы**...
В таком же формате, как на [главной странице](#).
- **Инфографика “О компании”***
Такая же, как на [главной странице](#).
- **Лидовый блок***
 - **Заголовок блока***...
Формируется вручную в CMS для текущей страницы.
 - **Форма***...
Состоит из четырех полей:
 - Имя* (текстовое)

- Телефон*
- E-mail*
- Комментарии к заявке...

Если заявка отправляется со страницы услуги 1-го уровня или с текстовой страницы, то поле остается пустым, пользователь заполняет его по своему желанию. Если со страницы 2-го и последующих уровней, то в поле автоматически прописывается название услуги (страницы), которое пользователь может откорректировать.

Отправка формы происходит по нажатию кнопки [Отправить запрос](#). При условии успешного прохождения [валидации](#), заявка отправляется в CMS и на почту администратору сайта и сохраняется в CMS с указанием персональных данных клиента и информации по заказанной услуге (заголовок страницы, с которой отправлена заявка). Для пользователя выводится текст подтверждения отправки заявки. Под кнопкой выводится текст согласия с пользовательским соглашением со ссылкой на него (открывается в новом окне).

- **Менеджер**
[Менеджер](#) прикрепляется к странице вручную в CMS (выбирается сотрудник из тех, которые добавлены вручную в CMS, и для которого стоит пометка “Использовать как контактное лицо”, т.е. для него указываются контактные данные). Если менеджер не прикреплен, то блок скрывается.
У каждого менеджера должны быть выведены следующие данные:
 - фото*
 - имя*
 - должность*
 - телефон* ([кликабелен](#) на мобильных устройствах)
 - e-mail (ссылка)

5. О компании

- **Первый экран**
 - Название страницы*
 - Аннотация*
 - Преимущества компании*...
Такие же, как на [главной странице](#).



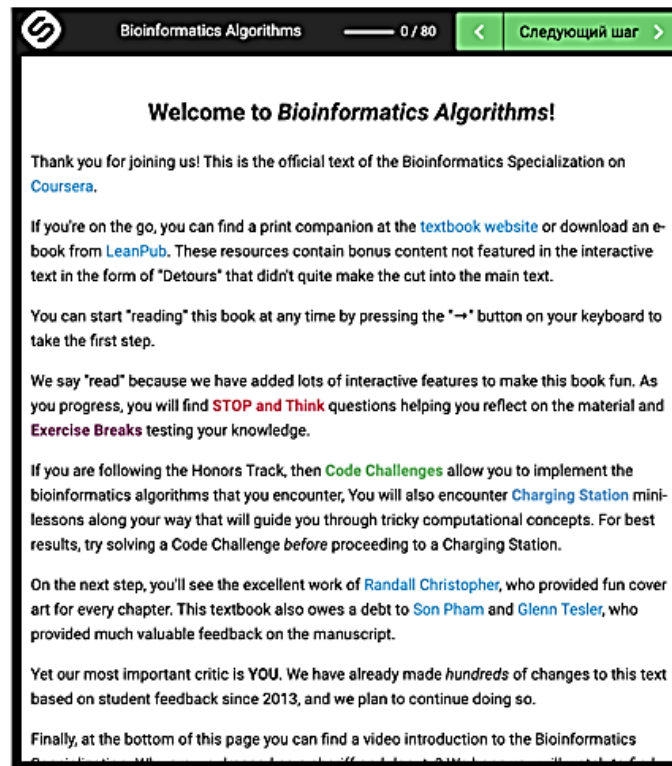
Приемы интеграции платформ на примере Stepik

После составления технического задания программисты приступают к созданию платформы. На сегодняшний день разработаны различные подходы к интеграции сайтов, информационных систем. Например, образовательная платформа и конструктор онлайн-курсов Stepik предлагает следующие формы интеграции любому пользователю.

Встраивание Stepik через `iframe`

`iframe` — это стандартный способ отображать один сайт внутри другого.

С его помощью можно встроить Stepik в другую платформу и наоборот.





Приемы интеграции с другими платформами на примере Stepik

Интеграция по протоколу LTI

LTI (Learning Tools Interoperability (IMS LTI 1.1)) — стандартный протокол, который описывает взаимодействие учебных платформ между собой. Он позволяет автоматически произвести аутентификацию пользователя одной платформы на другой платформе и передавать данные об успеваемости со второй на первую. Stepik поддерживает протокол в обе стороны.

Мой курс

Настройки интеграции по LTI

LTI Key (Consumer)

LTI Secret

Создавать приватные профили для учащихся

[Сохранить настройки LTI](#)

- Описание
- Содержание
- Права доступа
- Интеграция по LTI**
- Сертификаты

[Редактировать](#)



Приемы интеграции с другими платформами на примере Stepik

Авторизация по протоколу OAuth 2

OAuth 2 — протокол, с помощью которого можно настроить авторизацию на другой платформе через аккаунт на Stepik. Протокол дает безопасный способ обмена персональной информацией, избавляет от необходимости доверять приложению логин и пароль, а также позволяет выдавать ограниченный набор прав, а не все сразу.

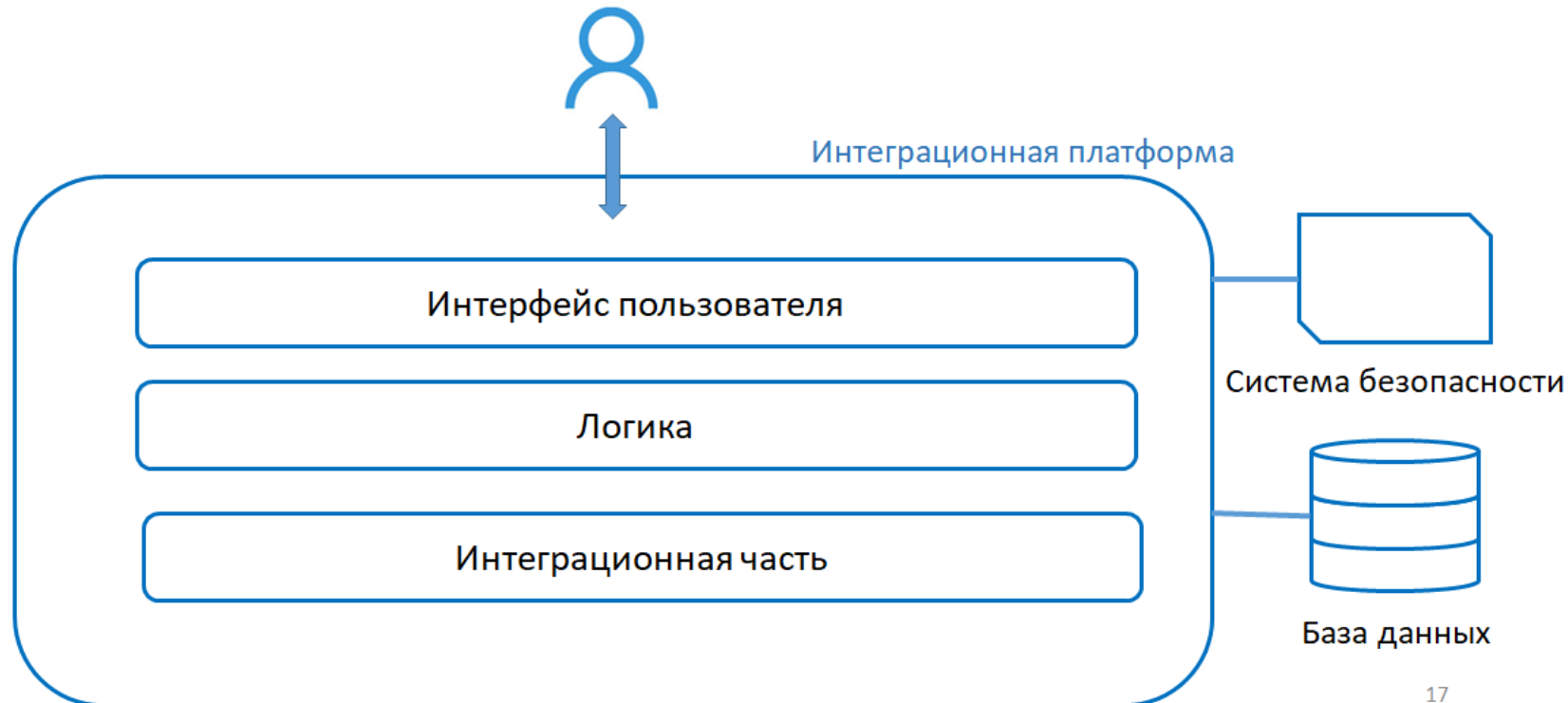
Сам Stepik поддерживает OAuth 2 авторизацию через аккаунты VK, Facebook, Google, GitHub и Twitter.

The screenshot shows the Stepik login interface. At the top, there are two tabs: "Войти" (Login) and "Регистрация" (Registration), with "Войти" being the active tab. Below the tabs is a close button (X). The main form contains two input fields: "E-mail" and "Пароль" (Password). Below these fields is a green "Войти" button. Underneath the button is a blue link "Напомнить пароль" (Remember password). At the bottom of the form, there is a section titled "Или войдите через социальные сети" (Or log in through social networks), which is circled in red. This section contains five social media icons: VK, Facebook, Google, GitHub, and Twitter.



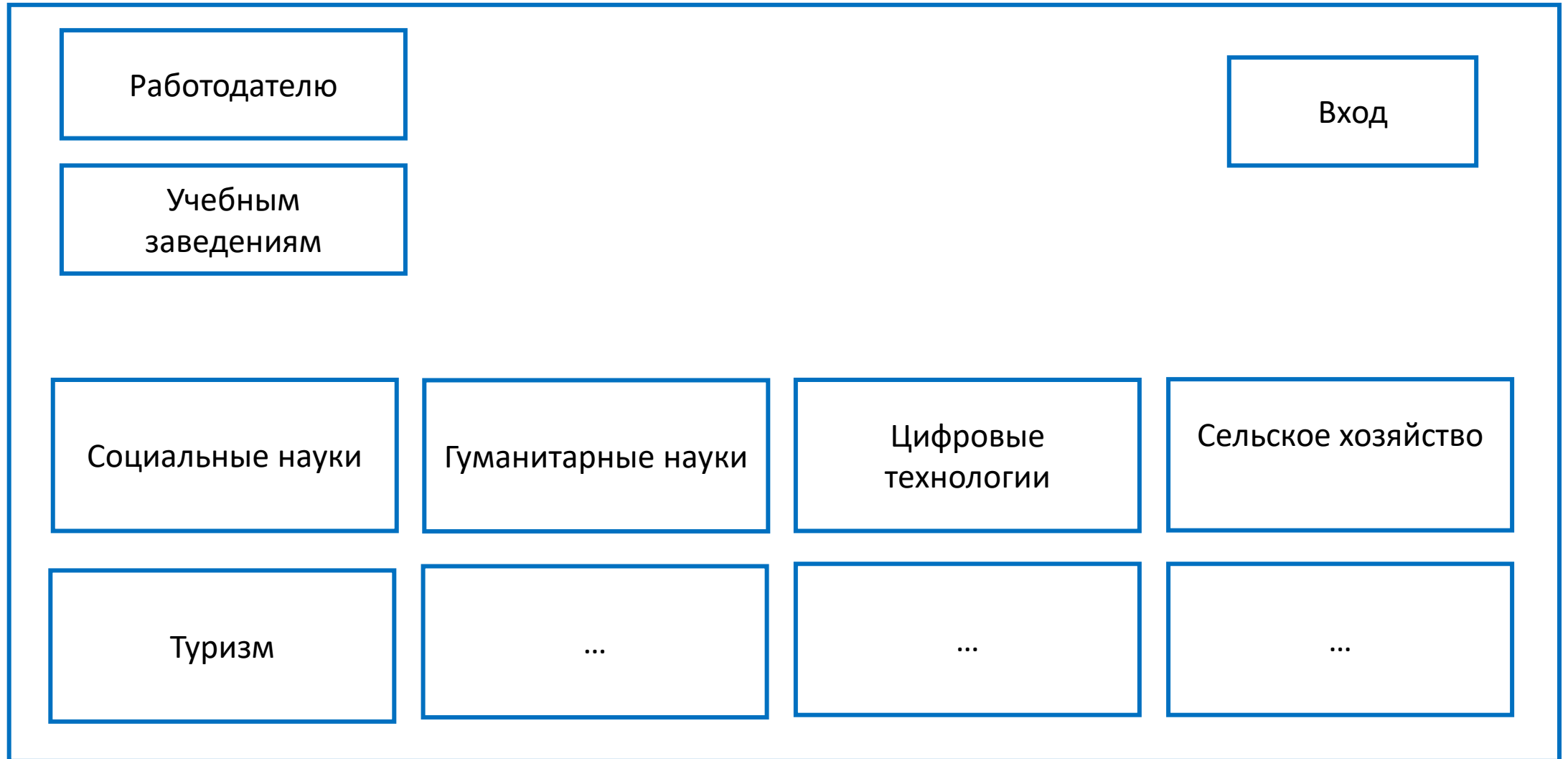
Интерфейс пользователя на интеграционной платформе

При работе над интеграционной платформой программисты разрабатывают интерфейс пользователя, продумывают логику, хранение данных, вопросы авторизации, безопасности. В зависимости от требований работа над платформой может продолжаться от месяца до нескольких лет.





Макет интерфейса интеграционной платформы





Источники:

- 1) Анализ требований к автоматизированным информационным системам. Лекция 2: Понятие требования. Классификации требований <https://intuit.ru/studies/courses/2188/174/lecture/4714>
- 2) Справочный центр Stepik. Интеграция с другими платформами <https://support.stepik.org>
- 3) В.В. Вейбер, С.А. Богдан, А.В. Кудинов, Н.Г. Марков. Концепция построения платформы для интеграции производственных данных нефтегазодобывающей компании. Томский политехнический университет /ОАО «ВостокГазпром», г. Томск