# Процесс разработки

Как уже было сказано во введении, разработка системы ведется на языке Python. Это интерпретируемый высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. В то же время стандартная библиотека включает большой объём полезных функций.

Python поддерживает несколько парадигм программирования, в том числе структурное, объектно-ориентированное, функциональное, императивное и аспектно-ориентированное. Основные архитектурные черты — динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений и удобные высокоуровневые структуры данных. Код в Python организовывается в функции и классы, которые могут объединяться в модули

Для Python принята спецификация программного интерфейса к базам данных DB-API 2 и разработаны соответствующие этой спецификации пакеты для доступа к различным СУБД: Oracle, MySQL, **PostgreSQL**, Sybase, Firebird (Interbase), Informix, Microsoft SQL Server и SQLite.

В нашей реализации используется PostgreSQL - свободная объектно-реляционная система управления базами данных (СУБД). Сильными сторонами PostgreSQL считаются:

* высокопроизводительные и надёжные механизмы транзакций и репликации;
* наследование;
* легкая расширяемость.

Разработка веб-сайта осуществляется с помощью фреймворка Django.

Сайт на Django строится из одного или нескольких приложений, которые рекомендуется делать отчуждаемыми и подключаемыми. Это одно из существенных архитектурных отличий этого фреймворка от некоторых других (например, Ruby on Rails). Один из основных принципов фреймворка — DRY (англ. Don't repeat yourself)

Веб-фреймворк Django используется в таких крупных и известных сайтах, как Instagram, Disqus, Mozilla, The Washington Times, Pinterest, YouTube, Google и др.

Предусмотрена локализация сайта для различных языков (на данный момент разрабатывается поддержка английского языка). Однако, переводы учебных материалов должны будут предоставлять сами авторы.

## Приложение «Пользователь»

Было разработано приложение «Пользователь», которое выполняет следующие функции:

* регистрация пользователей;
* авторизация пользователей;
  + через соц. сети;
  + с использованием двухфакторной аутентификации;
* отображение страницы пользователя;
* настройки профиля пользователя;
* работа и электронной почтой и другие.

Сущность «Пользователь» базы данных содержит следующие поля (поля, помеченные звездочкой (\*) являются обязательными):

* pk\* - ID пользователя, уникальный, первичный ключ;
* username\* - имя пользователя, уникальный;
* password\* - хэш пароля;
* first\_name - имя;
* last\_name - фамилия;
* email - электронная почта;
* gender - пол;
* job - место работы;
* birth\_date - дата рождения;
* bio - биография и прочая информация;
* country - страна;
* is\_staff - возможность заходить в Django Admin;
* is\_active - используется вместо удаления аккаунта;
* date\_joined - дата регистрации;
* last\_login - дата последнего логина.

## Приложение «Коллектив»

# Заключение

Мы считаем, что нам удалось добиться значительного прогресса в разработке системы. Концепт системы, как всегда, модернизировался; начата программная реализация проекта (и некоторые части уже завершены). На следующий семестр были поставлены такие планы:

* развитие концепции «облачной информационной системы обучения студентов»;
* реализация оставшихся приложений («учебный процесс», «разработка учебного материала» и «приложение»);
* разработать базовый набор приложений;
* разработать примеры учебных материалов.