

1. Глушаков С.В. Delphi2007. Самоучитель / С.В. Глушаков, А.Л. Клевцов. — изд. 4-е, доп. и перераб. — М.: АСТ: АСТ МОСКВА:ХРАНИТЕЛЬ, 2008. —448 с.
2. Кормен, Т., Лейзерсон, Ч., Ривест, Р., Штайн, К. Алгоритмы: построение и анализ = Introduction to Algorithms / Под ред. И. В. Красикова. — 2-е изд. — М.: Вильямс, 2005. — 1296 с. — ISBN 5-8459-0857-4.. Глава 26. Максимальный поток.
3. Хромоненко А.Д. Delphi7 / А.Д. Хромоненко, В.Э., Гофман Е.В. Мещеряков — 2-е издание, перераб. и доп. — СПб.:БХВ-Петербург, 2010.—1136 с.:ил.+CD-ROM-(В подлиннике)

ПОДГОТОВКА ДАННЫХ НА САЙТЕ ДЛЯ ПРИЛОЖЕНИЯ С ЭЛЕМЕНТАМИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Автор: Карпов Максим Александрович, студент 2 курса государственного университета «Дубна», филиал «Протвино»

Научный руководитель: к.т.н., доцент кафедры филиала «Протвино» университета «Дубна» Кульман Татьяна Николаевна

Аннотация

В работе рассматривается подготовка данных с использование сайта для приложения с элементами искусственного интеллекта. При создании сайта используется фреймворк Django, язык программирования Python и СУБД PostgreSQL.

The paper considers the preparation of data using the site for an application with elements of artificial intelligence. When you create a site, you use the Django framework, Python programming language, and PostgreSQL DBMS.

Технологии ближайшего будущего будут характеризоваться активным включением методов искусственного интеллекта (ИИ) в процессы решения задач. Работы в области искусственного интеллекта и моделирования приведут к созданию принципиально нового поколения интеллектуальных технологий. Для работы искусственного интеллекта, ему необходимы данные. В предлагаемой работе, источником данных будут служить данные, получаемые с сайта.

Объектом работы является подготовка данных на сайте с последующим использованием приложения с элементами ИИ.

Искусственный интеллект – наука и технология создания интеллектуальных машин, особенно интеллектуальных компьютерных программ [1].

При создании приложение с элементами ИИ планируется использовать алгоритм Гроссберга и Карпендера, ART1, который был первым в семье алгоритмов теории адаптивного резонанса (Adaptive Resonance Theory) [2]. Это простой алгоритм с обучением, основанный на биологической мотивации. Основная цель адаптивных систем – реализация управления процессом обучения с учётом индивидуальных особенностей пользователей. Адаптивные методы позволяют сократить время и повысить эффективность процесса обучения за счёт удержания пользователей в оптимальной зоне обучения, изменяя последовательность предъявления материала и заданий, темп обучения и нагрузку.

Алгоритмы кластеризации имеют биологическое происхождение, поскольку предоставляют возможность обучения посредством классификации. Человеческий мозг изучает новые понятия, сравнивая их с уже существующими знаниями. Мы классифицируем новое, пытаясь объединить его в одном кластере с чем-то, что нам уже

известно. Если новое понятие нельзя связать с тем, что мы уже знаем, нам приходится создавать новую структуру.

Алгоритм ART1 работает с объектами, которые называются векторами признаков. Вектор признаков является группой значений в двоичном коде.

Работа разбивается на несколько этапов. Первый этап – знакомство с алгоритмом ART1. После этого стало понятно, что необходим второй этап – подготовка данных для этого алгоритма. Третьим этапом является объединение результатов первых двух.

Целью данной работы является подготовка данных для алгоритма ART1.

Для реализации этой цели предлагается создать сайт, подключить к нему базу данных, а затем эти данные передавать в алгоритм.

В качестве инструментария для разработки сайта был использован фреймворк Django [3] и язык программирования Python [4]. Фреймворк – программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта. Django находится в свободном доступе, это дает возможность заметно упростить процесс создания сайта, так как программист может сфокусироваться на процессе дизайна и создании функционала приложения.

Django является быстрым решением в веб-разработке, включающим все необходимое для качественного написания кода.

Рассмотрим основные характеристики Django:

- Лёгкость в использовании: Django был создан, чтобы помочь программистам создать приложение настолько быстро, на сколько это возможно. Этот процесс включает в себя формирование идеи, разработку и выпуск проекта.
- Полная комплектация: Django работает с десятками дополнительных функций, которые заметно упрощают работу с аутентификацией пользователя, картами сайта, администрированием содержимого, RSS и многим другим.
- Безопасность: работая в Django, вы получаете защиту от ошибок, связанных с безопасностью и ставящих под угрозу проект.
- Разносторонность: научные вычислительные платформы, менеджмент контента, даже крупные организации – со всем этим можно эффективно справляться при помощи Django.

Python – это язык программирования общего назначения, нацеленный в первую очередь на повышение продуктивности самого программиста, нежели кода, который он пишет. Можно использовать динамическую типизацию для упрощения кода и встроенные функции языка, чтобы избавить себя от написания шаблонных кодов. На Python можно написать практически всё, что угодно без ощутимых проблем.

Предметной областью разрабатываемого сайта являются игры, более конкретно: тип игры, разработчик, покупатель, фотографии игры. Это и послужило основой для разработки базы данных.

Автоматически в Django создаётся база данных SQLite, но для ИИ её функционала будет недостаточно, поэтому в проекте решено использовать PostgreSQL.

PostgreSQL – это объектно-реляционная система управления базами данных (ОРСУБД).

Создание сайта выполняется на операционной системе Linux Ubuntu в редакторе кода Sublime Text 3 с версией языка Python3. Чтобы начать работать с Django, его необходимо установить и запустить, это можно сделать следующими командами:

- `sudo apt-get install python3-pip`
- `sudo pip3 install django`

- django-admin startproject «projectname»

В Django таблицы базы данных описываются как классы. В классе содержатся переменные и методы обработки. Благодаря этому, мы получаем не просто таблицы базы данных, но и код программы. Как и в любой базе данных, полям можно задать определённые свойства.

Описание самих таблиц производится на языке Python (см. рис.1).

В настоящее время созданы следующие таблицы:

- Genre – жанр игры, включает атрибуты жанра, даты создания и даты последнего изменения, специальная метка для упрощения поиска в браузере.
- Customers – покупатели, включает сведения о покупателях.
- Developers – разработчики, включает сведения о разработчиках, даты создания и даты последнего изменения, специальная метка для упрощения поиска в браузере.
- Image – содержит в себе картинки игр.
- Games – включает в себя все таблицы, название игры, разработчики, описание, дата создания, дата обновления, картинки, специальная метка для упрощения поиска в браузере.

На рисунке 1 можно увидеть описание таблицы «Genre».

```
class Genre(models.Model): #создаём класс(таблицу данных)
    genre = models.CharField('название жанра', max_length=256) # Создаём поле char с максимальной длиной 256 символов
    created = models.DateTimeField(auto_now_add=True) # Дата сощздания записи
    updated = models.DateTimeField(auto_now=True) # Дата изменения записи
    slug = models.SlugField(
        'slug',
        allow_unicode=True, max_length=254, blank=True)
        # это короткое название-метка, улучшает поиск в браузере.

    def __str__(self):
        return self.genre

    class Meta:
        verbose_name='Жанр' # отображение на сайте в админке
        verbose_name_plural='Жанры'

    def save(self, *args, **kwargs):
        self.autoslug() # Делаем slug
        super(Genre, self).save(*args, **kwargs) # сохраняем

    def autoslug(self):
        if self.name:
            self.slug = slugify(self.name, allow_unicode=True)
```

Рисунок 1. Код класса «Genre» на языке Python

Рассмотрим особенности создания таблицы БД в виде класса, на примере «Genre». Сначала описываются поля (genre, created, updated, slug), сохранение которых определяется в методах того же класса «Genre». В класс Genre включён ещё один класс Meta, задачей которого является отображение информации о классе Genre.

Все рассмотренные выше классы подключаются к классу Games.

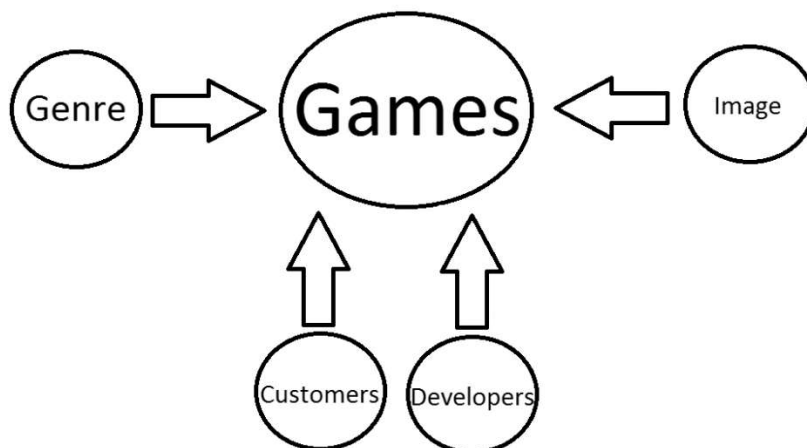


Рисунок 2

На данном этапе автоматически с помощью Django были созданы основные файлы для запуска сайта и подключена база данных, реализованная на PostgreSQL, таблицы которой описаны на языке Python. Кроме этого, Django создаёт сайт администратора для управления и заполнения данными разработанного сайта.

Теперь можно начинать заполнять БД, используя средства сайта администратора (рис.3).

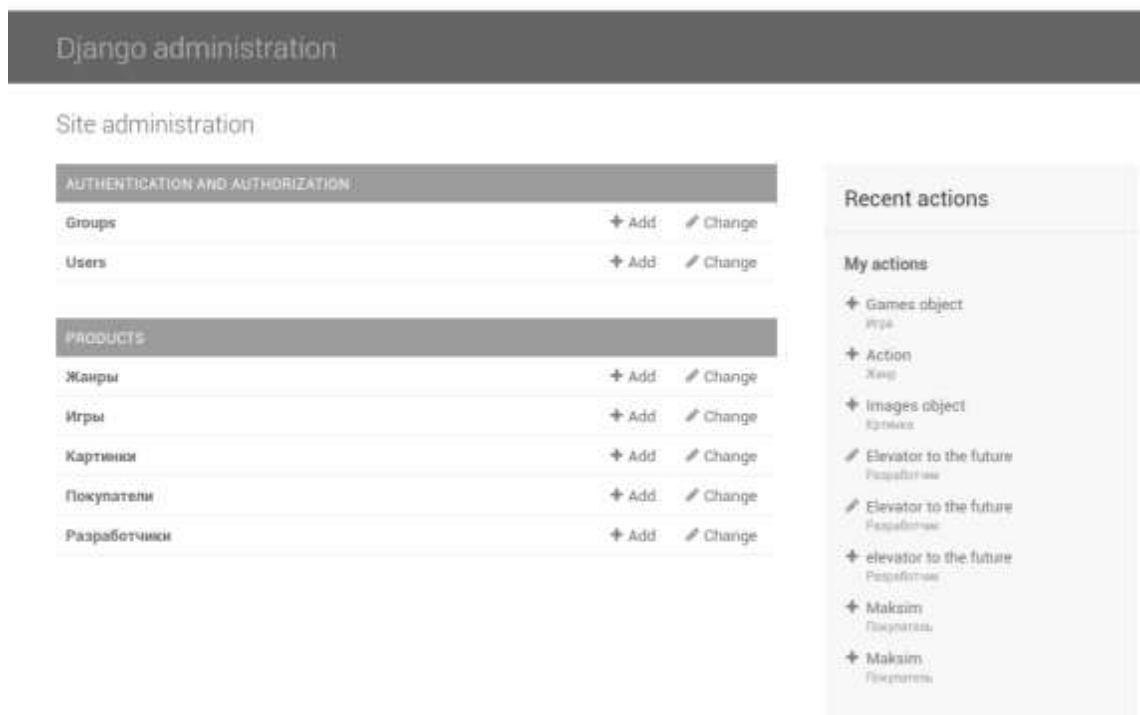


Рисунок 3. Ввод данных в БД и управление ими

Данные, введённые в БД можно просмотреть на сайте администратора.

В итоге, мы рассмотрели построение сайта, создание БД с использованием языка Python, фреймворка Django, ОРСУБД PostgreSQL. Введённые в БД данные подготовлены для передачи их в приложение с элементами ИИ.

В дальнейшем планируется разработка системы ИИ, данные в которую будут передаваться из описанного в данной работе сайта.

Список использованной литературы

1. Толковый словарь по искусственному интеллекту - <http://www.raai.org/library/tolk/aivoc.html#L208>
2. Программирование искусственного интеллекта в приложениях / М. Тим Джонс ; Пер. с англ. Осипова А. И. – М.: ДМК Пресс, 2004. – 312 с : ил.
3. Документация Django – документация Django 1.9 - <https://djangobook.ru/rel1.9/>
4. Прохоренок Н.А. Python 3 и PyQt. Разработка приложений. – СПб.: БХВ-Петербург, 2012. – 704 с.
5. Лутц М. Программирование на Python, том II, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с
6. Лутц М. Изучаем Python, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 1280 с
7. PostgreSQL: Документация: 9.6: 1. - <https://postgrespro.ru/docs/postgresql/9.6/intro-what-is.html>

ГЕНЕРАЦИЯ ЛАБИРИНТОВ ДЛЯ ДЕТЕЙ МЛАДШЕГО ШКОЛЬНОГО И ДОШКОЛЬНОГО ВОЗРАСТА

Автор: Кузина Ирина Александровна, студентка 3 курса филиала «Протвино» государственного университета «Дубна»

Научный руководитель: кандидат технических наук, доцент кафедры филиала «Протвино» государственного университета «Дубна» Кульман Татьяна Николаевна.

Аннотация.

Данное приложение создано в среде разработки программного обеспечения Microsoft Visual Studio 2015. Функциональная часть реализована на языке программирования C++ при помощи стандартной библиотеки шаблонов STL с использованием контейнерного класса stack. Для создания интерфейса также использовалась среда разработки Microsoft Visual Studio 2015, но вся реализация производилась на языке программирования C#.

В данной статье описывается ход создания проекта по генерации лабиринтов для детей младшего школьного и дошкольного возраста. Данная тема представляет интерес с различных точек зрения. Во-первых, это развитие мышления посредством углубленного изучения генерирования лабиринтов и учёта всех возможных условий для его создания. Во-вторых, обучение работе с различными алгоритмами для генерирования лабиринтов, прохода по ним с целью поиска оптимального пути. И в-третьих, приобретение навыков работы с различными языками программирования, умение переносить проект с одного языка программирования на другой, а также создание пользовательского интерфейсов под цели проекта.

Актуальность данного проекта обуславливаются её практической значимостью. Проект может быть полезен для развития логического мышления у детей младшего школьного и дошкольного возраста, ввиду того, что есть возможность генерировать лабиринты от самых простых и маленьких до более сложных и больших.

Если основываться на мнениях психологов, то также можно утверждать, что лабиринты помогают в развитие внимания у ребёнка, развивают моторику ребёнка, а также усидчивость и навыки достижения целей посредством обхода возникающих перед ребёнком препятствий.