управлении, изобретение и внедрение в нашу хозяйственно-бытовую деятельность современных типов экологической как техники, так и технологии, и многое-многое другое.

Учитывая изложенное, считаем, что экологизация бизнес-деятельности должна последовательно меняться и развиваться по следующим ключевым блокам: усовершенствование промышленно-технологических процессов, низкоотходных и неотходных нано-технологий и разработка такого оборудования, которое отличалось бы меньшим объемом выбросов вредных для природы и человека примесей и отходов в окружающую среду; повсеместное широкое присутствие экологической спецэкспертизы всех видов бизнес-производств и индустриально-массовой продукции; замена токсичных и неутилизируемых отходов на нетоксичные и утилизируемые.

Библиографический список

1. Основы государственной политики в области экологического развития России на период до 2030 года. М.: 30.04.2012.
2. Основные показатели охраны окружающей среды. Стат.сб. – М.: Росстат, 2013. – 112с.
3. Охрана окружающей среды в России. Стат.сб. – М.: Росстат, 2016. – 95с.
4. План действий по реализации Основ государственной политики в области экологического развития Российской Федерации на период до 2030 года. Утвержден распоряжением Правительства Российской Федерации от 18 декабря 2012 г. № 2423-р.
5. Российский статистический ежегодник. Стат.сб. – М.: Росстат, 2016. – 725 с. [Электронный ресурс]. Режим доступа: *https://ria.ru/ society/20161110/1481082290.html*
6. Стрелец, И.А. Влияние новых технологий на экономическое поведение потребителей и фирм / И.А. Стрелец // США и Канада: экономика, политика, культура. - 2008. - № 8. - С. 63-72.

М.А Карпов, Т.Н. Кульман

РАЗРАБОТКА САЙТА С ЦЕЛЬЮ ПОДГОТОВКИ ДАННЫХ ДЛЯ ПРИЛОЖЕНИЯ С ЭЛЕМЕНТАМИ ИСКУСТВЕННОГО ИНТЕЛЛЕКТА

Филиал «Протвино» государственного университета «Дубна»
Секция информационных технологий

В работе рассматривается разработка сайта с целью подготовки данных для приложения с элементами искусственного интеллекта. При создании сайта применяются: язык HTML, bootstrap 3, CSS, фреймворк Django, язык программирования Python и СУБД PostgreSQL.

Современные технологии характеризуются активным включением методов искусственного интеллекта (ИИ) в процессы решения задач. Работы в области искусственного интеллекта приводят к созданию принципиально новых информационных приложений. Одной из задач таких приложений является ввод данных, необходимых для работы искусственного интеллекта. В предлагаемой работе, источником данных будут служить данные, получаемые с сайта.

Объектом работы является разработка сайта и подготовка данных на нём с последующим использованием в приложении с элементами ИИ.

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

При создании приложения с элементами ИИ планируется использовать алгоритм Гроссберга и Карпентера, *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.

Первоначально была начата разработка приложения с элементами ИИ. Однако, в ходе разработки встал вопрос о подготовке данных для алгоритма ИИ.

В связи с этим, работа разбивается на несколько этапов. Первый этап – знакомство с алгоритмом *ART1*. Второй этап – подготовка данных для этого алгоритма. Третьим этапом является разработка небольшого приложения на базе алгоритма *ART1*.

В данной работе рассматривается подготовка данных для алгоритма *ART1*.

Для реализации этой цели предлагается создать сайт, подключить к нему базу данных, а затем эти данные передавать в алгоритм. Задача сайта – предоставлять данные для приложения.

Самым большим источником информации является интернет, а именно – сайты, поэтому было принято решение создания сайта – интернет-магазин. Предметной областью разрабатываемого сайта является описание компьютерных игр, более конкретно: типа игры, разработчика, покупателя, фотографий игры. Эта информация послужила основой для разработки базы данных.

Приложение с элементами ИИ будет анализировать, какие товары покупатель просматривает и покупает, будет выдавать рекомендации для конкретного пользователя.

В качестве инструментария для разработки сайта был использован фреймворк *Django* [3], язык программирования *Python* [4], язык разметки HTML, *CSS*, и фреймворк *bootstrap*. Рассмотрим кратко инструментарий, применяемый в работе.

Фреймворк – это программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта. *Django* находится в свободном доступе и 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.

Основными характеристиками *Django* являются:

- лёгкость в использовании,
- полная комплектация,
- безопасность,
- разносторонность.

Python – язык программирования общего назначения [5, 6], 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 языке *Python* можно написать практически всё, что угодно, без ощутимых проблем.

HTML – это стандартизованный язык разметки. Язык *HTML* интерпретируется браузерами, затем полученный в результате интерпретации форматированный текст отображается на экране монитора компьютера или мобильного устройства.

Bootstrap – свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя *HTML* и *CSS* – шаблоны оформления для типографии, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса.

CSS (каскадные таблицы стилей) – формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

Автоматически в *Django* создаётся база данных *SQLite*, но для ИИ её функционала будет недостаточно, поэтому в проекте решено использовать *PostgreSQL* [7].

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, 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* (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, разработчиков, описание, дату создания, дату обновления, картинки, специальную метку для упрощения поиска в браузере.

На рис. 1 можно увидеть описания таблицы *«Genre»*.

```
class Genre(models.Model):   #создаём класс(таблицу в БД)
        genre = models.CharField(
            'название жанра',
            max_length=256)  #создаём поле char с максимальной длинной 256 символов
        created = models.DateTimeField(auto_now_add=True) #дата создания записи
        updated = models.DateTimeField(auto_now=True) #дата изменения записи
        slug = Models.SugField(  #это короткое название-метка, улучшает поиск в браузере
            'slug',
            allow_unicode=True, max_length=256, blunk=True)

        def __str__(self):
                return self.genre

        class Meta:             # отображение таблицы БД на сайте в админке
                verbose_name='Жанр'
                verbose_name_pular='Жанры'

        def save(self, *args, **kwargs):  #функция, которая сохраняет изменения
                self.autoslug()  #делаем slug
                super (Genre, self).save(*args, **kwargs)  #сохранение изменений

        def autoslug(self): #создание короткой метки
                if self.name:
                        self.slug = slugify(self.name. allow_unicode=True)
```

Рис. 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* (рис. 2).

С помощью *Django* автоматически 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, 4).

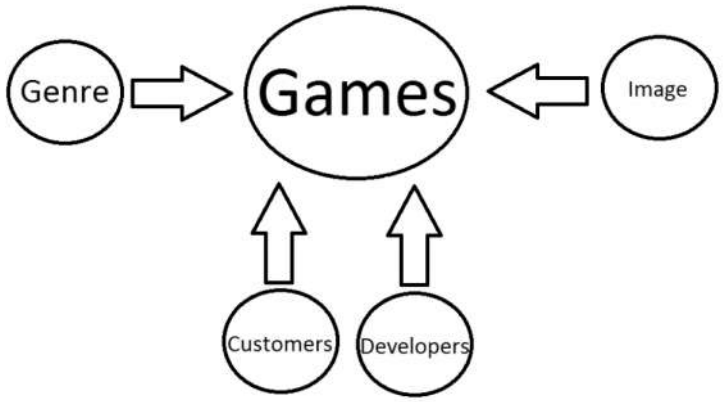

Рис. 2 Схема взаимодействия классов

Рис. 3 Таблицы БД

На рис. 4 можно увидеть какие данные уже имеются в БД и занести новые.

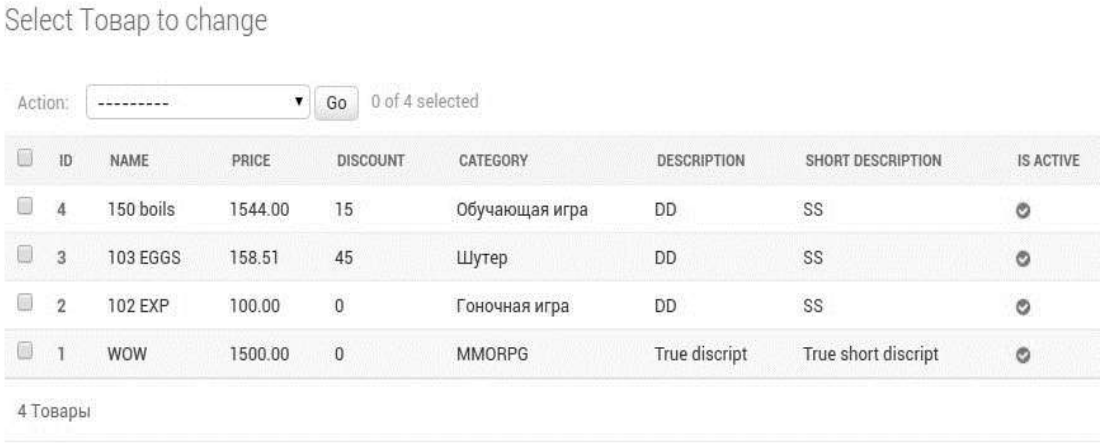

Рис. 4 Заполнение и просмотр данных в БД

Заключение

В итоге, был создан сайт интернет-магазина, рассмотрены создание БД с использованием языка *Python*, фреймворк *Django*, ОРСУБД *PostgreSQL, HTML, CSS, Bootstrap*. Введённые в БД данные подготовлены для передачи их в приложение с элементами ИИ.

В дальнейшем планируется разработка приложения с элементами ИИ, данные в которое будут передаваться из описанного в данной работе сайта.

Библиографический список

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

И.О. Ковцова, Д.Д. Попрыго

РАЗРАБОТКА КОМПЬЮТЕРНОЙ ВИДЕО ИГРЫ «КАМО ГРЯДЕШИ» В СРЕДЕ *UNITY*

Филиал «Протвино» государственного университета «Дубна»
Секция информационных технологий

В статье рассматриваются основные особенности игрового процесса, включающего в себя правила видеоигры, которым она подчиняется. Ключевые слова: видео игра, среда разработки Unity.

Unity - это инструмент для разработки двух - и трёхмерных приложений, и игр, работающий под операционными системами *Windows, Linux* и *OS X* [1]. Созданные с помощью *Unity* приложения работают под операционными системами *Windows, OS X, Windows Phone, Android, Apple iOS, Linux*, а также на игровых приставках *Wii, PlayStation 3, PlayStation 4, Xbox 360, Xbox One* и *MotionParallax3D* дисплеях (устройства для воспроизведения виртуальных голограмм), например, *Nettlebox*.

«Камо грядеши» - это двухмерная компьютерная видеоигра, в жанре «платформер». Основной чертой жанра платформер является ограниченное движения протагониста по оси *Z*, в игровом мире, прыгание по платформам, лазанье по лестницам и собирание предметов, обычно необходимых для завершения уровня или выживания персонажа. Сбор предметов осуществляется посредством обыска игроком различных объектов. Собираемые предметы делятся на несколько классов:

- *Weapon* – (оружие), предметы которые использует игрок, для защиты;
- *Consumable* – (потребляемый) – предметы которые игровой персонаж, потребляет с целью повышения, показателя голода и жажды.
- *Quest* – (предметы для задания) – предметы которые нужны, для выполнения определенных задач поставленных перед игроком.

Найденные объекты помещаются в инвентарь (рис 1), выполненный в виде поля, разделенного на двенадцать ячеек. В каждой ячейке размешен найденный предмет, при наведении на который отображаются некоторые характеристики.

С помощью метода оперирования элементами интерфейса «Бери-и-брось» игрок может менять расположение предметов в сетке, удалять и передавать их не игровым персонажам. Для взаимодействия с игровыми объектами в игре реализованы всплывающие иконки, появляющиеся в момент приближения игрока к объекту. В случае приближения игрока к двери в работу вступает следующий участок кода.

```
if (tags == "door")
{
if (doorScript.canOpen(moveScript.numberOfChooseCharacter) == false)
{
noticeObj.GetComponent<SpriteRenderer>().sprite = grayNoticeSpr;
}
else {
```