

Точную дату завершения этого процесса сложно предсказать, однако можно уверенно сказать, что он будет продолжаться и развиваться в ближайшие годы.

Разработанный прототип имеет возможности для масштабирования и расширения функционала.

Список использованных источников:

1. Земсков А.И. Электронная информация и электронные ресурсы: публикации и документы, фонды и библиотеки / А.И. Земсков, Я.Л. Шрайберг ; под ред. Л.А. Казаченковой. – М., 2007. – С. 329–331. – URL: https://www.studmed.ru/view/zemskov-ai-elektronnaya-informaciya-i-elektronnye-resursy-publikacii-i-dokumenty-fondy-bibliotek_b37beff4386.html (дата обращения: 24.02.2024). – Текст: электронный.

2. Болотов А.В. Перспективы использования программного продукта для автоматизации библиотек крупных университетов корпорации VTLS (США) в научной библиотеке ТГУ / А.В. Болотов // Автоматизированные информационные системы в образовании и науке. – Томск, 2006. – 7 с. – URL: <https://vital.lib.tsu.ru/vital/access/services/Download/vital:3036/SOURCE01> (дата обращения: 24.02.2024). – Текст: электронный.

3. Гололобова А.В. Программное обеспечение автоматизированных библиотечных систем в России / А.В. Гололобова // Культурная жизнь Юга России. – 2011. – № 3. – URL: <https://cyberleninka.ru/article/n/programmnoe-obespechenie-avtomatizirovannyh-bibliotechnyh-sistem-v-rossii> (дата обращения: 24.02.2024). – Текст: электронный.

4. Никуличева Н.В. Анализ современных средств автоматизации библиотечно-информационной деятельности общеобразовательных организаций / Н.В. Никуличева, Е.А. Сармаева, С.С. Хапаева // Библиосфера. – 2015. – № 4. – URL: <https://cyberleninka.ru/article/n/analiz-sovremennyh-sredstv-avtomatizatsii-bibliotechno-informatsionnoy-deyatelnosti-obscheobrazovatelnyh-organizatsiy> (дата обращения: 24.02.2024). – Текст: электронный.

5. Нещерет М.Ю. Цифровизация процессов обслуживания в библиотеках – это уже реальность / М.Ю. Нещерет // Библиосфера. – 2019. – № 2. – URL: <https://cyberleninka.ru/article/n/tsifrovizatsiya-protsesov-obsluzhivaniya-v-bibliotekah-eto-uzhe-realnost> (дата обращения: 24.02.2024). – Текст: электронный.

СОЗДАНИЕ ИГРЫ ПАСМАН НА ЯЗЫКЕ C#

Д.А. Колмыков^а, студент гр. 17В11

*Научный руководитель: Колегова О.А.^б, ассистент
Юргинский технологический институт (филиал)*

Национального исследовательского Томского политехнического университета

652055, Кемеровская обл., г. Юрга, ул. Ленинградская, 26

E-mails: ^а26kda@tpu.ru, ^бOlga01@tpu.ru

Аннотация: В статье рассматривается процесс разработки игры Пасман на языке программирования C# с использованием библиотеки Windows Forms. Освещаются ключевые шаги создания игры, включая проектирование архитектуры, реализацию игровой логики, разработку пользовательского интерфейса и тестирование программы. Результатом работы является функциональная игра Пасман, способная запускаться и работать на платформе Windows.

Ключевые слова: игра Пасман, C#, Windows Forms, разработка игр, пользовательский интерфейс, тестирование.

Abstract: The article discusses the process of developing a Pacman game in the C# programming language using the Windows Forms library. The key steps of creating a game are highlighted, including architecture design, implementation of game logic, user interface development and program testing. The result is a functional Pacman game that can run and run on the Windows platform.

Keywords: Pacman game, C#, Windows Forms, game development, user interface, testing.

Введение

В современном мире компьютерные игры являются популярным развлечением и создание собственной игры может быть увлекательным и интересным проектом для разработчиков. В этой статье мы рассмотрим процесс создания игры Пасман на языке программирования C# с использованием библиотеки Windows Forms.

1. Проектирование игры

Проектирование игры включает определение основных компонентов, игровых механик и структуры программы. В случае игры Распан необходимо определить классы игровых объектов (игрок, привидения, монетки), логику и правила игры, а также пользовательский интерфейс.

2. Реализация игровой логики

После проектирования начинается реализация игровой логики. Это включает в себя написание кода для перемещения игровых объектов, обработки столкновений, подсчета очков и управления игровым процессом.

3. Разработка пользовательского интерфейса

Создание пользовательского интерфейса игры Распан включает в себя размещение элементов управления на форме (окне) игры, отображение игрового поля, игровых объектов и отображение информации о текущем состоянии игры.

4. Схема движения информационных потоков



Рис. 1. Схема движения информационных потоков

Схема показывает движение информационных потоков от игрового окна до обработчика событий клавиш, далее к логике игры и наконец к отображению состояния игры. Ввод клавиш игроком передается в обработчик событий клавиш, который изменяет состояние и положение игрока. Логика игры затем использует это состояние и положение для проверки столкновений и условий победы/поражения. Информация о текущем состоянии игры передается от логики игры к модулю отображения, который отображает соответствующую информацию на игровом окне.

5. Блок-схема программы

При создании программного алгоритма «Распан» на начальном этапе была разработана блок-схема игры. В ней была описана последовательность работы алгоритма игры.

5. Тестирование и отладка

После завершения разработки следует провести тестирование игры, чтобы выявить и исправить ошибки, а также убедиться в правильной работе всех игровых механик и элементов пользовательского интерфейса.



Рис. 2. Блок схема

Результат

В результате выполнения всех шагов создается полноценная игра Расман, способная запускаться и работать на компьютерах под управлением операционной системы Windows. Игра может предоставлять игрокам увлекательный игровой опыт, соответствующий классическим правилам и механикам игры Расман.



Рис. 3. Скриншот экрана игры

Заключение

Создание игры Расман на языке программирования C# является увлекательным и познавательным проектом, который позволяет разработчикам углубиться в мир разработки компьютерных игр и приобрести практические навыки программирования. Результатом такой работы становится функциональная игра, способная принести удовольствие и развлечение многим игрокам.

Список использованных источников:

1. Буйначев С.К. Основы программирования на языке Python : учебное пособие / С.К. Буйначев, Н.Ю. Боклаг. – Екатеринбург : Изд-во Урал. ун-та, 2014. – 91 с.

2. Войниканис Е.А. База данных как объект правового регулирования: учебное пособие для вузов / Е.А. Войниканис, В.О. Калятин. – Москва : Статут, 2011. – 174 с. – URL: <https://znanium.com/catalog/product/314835> (дата обращения: 28.11.2023). – Режим доступа: по подписке. – Текст: электронный.
3. Пушников А.Ю. Введение в системы управления базами данных. Часть 1. Реляционная модель данных: учебное пособие / А.Ю. Пушников. – Уфа : Изд-е Башкирского ун-та, 1999. – 108 с.
4. Разработка Windows-приложений на основе Visual C#/ Intuit. – Москва : Electronic pub, 2005. – 424 с.
5. Тюкачев, Н.А. С#. Основы программирования : учебное пособие для вузов / Н.А. Тюкачев, В.Г. Хлебостроев. – 4-е изд., стер. – Санкт-Петербург : Лань, 2021. – 272 с. – Текст: электронный // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/158960> (дата обращения: 22.11.2022). – Режим доступа: для авториз. пользователей.

РАЗРАБОТКА АРХИТЕКТУРЫ ВОПРОСНО-ОТВЕТНОЙ СИСТЕМЫ НА БАЗЕ БОЛЬШОЙ ЯЗЫКОВОЙ МОДЕЛИ

Я.А. Шенцов^а, магистрант гр. ПИ-23.01м

Научный руководитель: Чернышева Т.Ю., к.т.н., доц.

Тюменский государственный университет,

625003, Россия, Тюменская обл., г. Тюмень, ул. Володарского, 6

E-mail: ^а yaroslavshentsov@yandex.ru

Аннотация: В настоящее время является актуальной задача автоматизации различных сфер жизни человека. Одним из популярных подходов в данной задаче является внедрение вопросно-ответных систем в бизнес-процессы. Современные вопросно-ответные системы основаны на использовании больших языковых моделей (LLM), которые, в свою очередь, обучаются на больших объемах данных для умения вести диалог с пользователем. Как правило, доменные области автоматизируемых бизнес-процессов являются специфичными, поэтому предобученные большие языковые модели не всегда способны корректно дать ответ на заданный вопрос. Отсюда появляется необходимость дообучения языковой модели под заданную область знаний, но проблемой в данном случае является тот фактор, что для данного процесса необходим большой объем данных и мощные вычислительные ресурсы. Для решения данной проблемы в большую языковую модель интегрируются методы извлечения и генерации информации RAG (Retrieval Augmented Generation). В статье описывается реализация вопросно-ответной системы с использованием RAG-подхода для такой области знаний, как «Нейронные сети».

Ключевые слова: вопросно-ответная система, большие языковые модели, модель искусственного интеллекта, RAG, архитектура системы.

Abstract: Currently, the task of automating various aspects of human life is highly relevant. One popular approach in this task is the integration of question-answering systems into business processes. Modern question-answering systems are based on the use of large language models (LLM), which, in turn, are trained on large datasets to engage in dialogues with users. Typically, the domain areas of automated business processes are specific, so pre-trained large language models may not always provide accurate answers to the given questions. This leads to the need for fine-tuning language models for a specific knowledge domain. However, the challenge in this case is the requirement for a large volume of data and powerful computational resources for the fine-tuning process. To address this issue, methods of information retrieval and generation, such as Retrieval Augmented Generation (RAG), are integrated into large language models. This article describes the implementation of a question-answering system using the RAG approach for the knowledge domain of «Neural Networks».

Keywords: question-answering systems, large language models, artificial intelligence model, RAG, system architecture.

Большая языковая модель (Large Language Model) – это модель искусственного интеллекта, которая обучается на больших объемах текстовых данных и имеет большое количество параметров. LLM предназначены для обработки, понимания и генерации текста. Наиболее популярным представителем данной модели является GPT. В настоящее время различные LLM активно внедряются во многие бизнес-процессы: чат-боты, поисковые системы, роботизированные операторы в call-центрах, системы генерации контента и многое другое. Это помогает существенно сократить время обработки и решения поставленного вопроса, а также упрощает работу и снимает часть рутинных обязанностей с человека.