Компьютерная академия «ШАГ»

Николаевский филиал

Кафедра Разработки программного обеспечения

КУРСОВОЙ ПРОЕКТ

«Разработка игрового приложения Домино .NET»

Студента группы 27ВП-1  
Юркова В.А.

Преподаватель:  
Устенко С.А.

Николаев 2018

Аннотация

В данном курсовом проекте рассматривается разработка алгоритма, а также пользовательского интерфейса, реализующие одноименную настольную игру «Домино». В результате построения алгоритма показан игровой механизм «игрок против игрока», а также объединение технологий для его построения.

Программная реализация алгоритма и пользовательского интерфейса осуществлена на языке программирования высокого уровня C#, с использованием технологий Windows Communication Foundation, Windows Presentation Foundation, ADO.NET, а также использование интегрированной среды разработки Visual Studio.

Содержание

Введение 4

Техническое задание 5

1. Выбор технологии реализации проекта 7

2. Разработка алгоритмов функционирования системы 8

3. Разработка базы данных и структуры системы 10

4. Разработка интерфейса системы 14

Выводы 16

Список использованной литературы 17

Введение

Домино́ — настольная логическая игра, в процессе которой выстраивается цепь костяшек («костей», «камней»), соприкасающихся половинками с одинаковым количеством точек, обозначающим число очков.

Корни игры в домино уходят в Индию и Китай, именно там появились кости в виде пластинок с точками красного и белого цвета. В XVIII веке игра была привезена в Италию и видоизменена. Современное домино также тесно связано с игрой в кости. Две половинки костяшки домино — это одна из возможных комбинаций, выпадающих при броске двух костей.

Стандартный набор традиционного (классического) западного домино включает в себя 28 костяшек. Костяшка домино представляет собой прямоугольную плитку, длина вдвое больше ширины. Её лицевая сторона разделена линией на две квадратные части. Каждая часть содержит от нуля до шести точек. В специализированных наборах домино возможное количество точек может доходить до девяти, двенадцати, пятнадцати или восемнадцати[1].

Техническое задание

Программное средство должно обеспечить следующее (функциональные характеристики):

* обеспечить пользователю возможность запуска игры и предоставить возможность работы с персональной учетной записью, если такая присутствует;
* обеспечить возможность подключиться у удаленному серверу для начала игры (если сервер свободен);
* обеспечить пользователю возможность сделать ход в интерактивном режиме;
* автоматически отображать текущее состояние игрового поля включая время на совершение хода, номер игры, текущий счёт;
* обеспечить пользователю возможность закончить игру.

Отказы программы вследствие некорректных действий пользователя при взаимодействии с программой недопустимы.

Программный продукт разрабатывается для персональной ЭВМ со следующими характеристиками:

* процессор с частотой не ниже 1 ГГц;
* объем ОЗУ не менее 256 Мб;
* графический адаптер;
* манипулятор типа «мышь»;
* клавиатура;
* место, занимаемое на жестком диске – 10 Мб.

Требования к информационной и программной совместимости.

ЭВМ должна работать под управлением операционной системы не ниже, чем Windows XP для запуска windows-совместимых приложений.

Стадии разработки

Разработка должна быть проведена в три стадии:

1. разработка технического задания – 5 дней;
2. рабочее проектирование – 1 неделя;
3. внедрение – 2дня.

1. Выбор технологии реализации проекта

Реализация игрового проекта «Домино» базируется на современных технологических решениях, выбор которых осуществляется на основе требований со стороны предполагаемых пользователей.

При внедрении и использовании были детально изучены такие технологии как:

* язык программирования C#;
* Windows Communication Foundation;
* Windows Presentation Foundation;
* ADO NET;
* платформа .NET[2].

Данные технологии тесно взаимосвязаны между собой, что даёт преимущество в выборе и их взаимодействии.

Критерии при руководстве выбора технологий реализации проекта:

* размер и тип проекта;
* сложность проекта;
* скорость разработки;
* доступные инструменты разработки;
* наличие готовых решений;
* гибкость решения;
* наличие широкого сообщества;
* отказоустойчивость решения;
* наличие подробной документации;
* требования к нагрузкам;
* требования к безопасности;
* возможности интеграции с другими решениями.

2. Разработка алгоритмов функционирования системы

Моделирования игровой ситуации «Домино» позволило разработать программную часть, в которой описывается основной алгоритм работы программы, а также приводится блок-схема этого алгоритма (рисунок 2.1).

Для лучшего понимания представленной блок-схемы, ниже приводится детальное описание, каждого из весомых блоков:

1. Программа запускается с возможностью входа в учётную запись.

2. После входа в учётную запись появляется возможность подключиться к игре, а также посмотреть свою статистику.

3 После подключения всех игроков начинается новая игра, сервер определяет возможность первого хода.

4. В процессе игры ход за ходом определяется кто будет ходить и нужно ли добирать кости из колоды.

5. Когда один из игроков выбросит все кости, или сервер выявит невозможность хода всех игроков, то определяется победитель в данной партии или игры в целом и подсчитывается количество набранных им очков.

6. Если после окончания партии счет игрока меньше 100, то игра продолжается и начинается новая партия.

7. Игра заканчивается, когда, или компьютер, или игрок набрали сумму, которая равна или превышает 100 очков.

9. После окончания игры обновляется статистика игроков.

8. В любой момент игрок может закрыть приложение. Если игрок находится в игре, ему запишут поражение.



Рисунок 2.1. Блок-схема алгоритма приложения

3. Разработка базы данных и структуры системы

При проектировании базы данных в проекте было использовано такое программное обеспечение как MS SQL Server 2017.

Основные пользователи базы данных это люди, которые зарегистрировались для участия в игре. При помощи БД они могут узнать сколько они сыграли игр и сколько побед одержали. Можно просмотреть минимальное количество очков, набранное в ходе игры.

В рассматриваемой предметной области создана одна сущность Accounts, куда вносятся все данные об пользователе.

Данная сущность содержит различные атрибуты. Атрибут – свойство сущности (заголовок столбца таблицы).

Перечислим атрибуты данной сущности: Login – имя пользователя; Password – пароль для доступа к данной сущности; ReminderText – для отображения пароля при его забытии; Games – общее количество сыгранных игр; Wins – количество побед; MinScore – минимальное количество набранных очков. Таблица Accounts представлена на рисунке 3-1.



Рисунок 3-1. Сущность Accounts базы данных

Проанализировать структуру системы нам помогут диаграммы классов приложения Клиент (рисунок 3-2) и приложения Сервер (рисунок 3-3).

Основные классы приложения Клиент:

* MainWindow – которое инициализирует игровое поле. В данном классе реализованы все основные поля приложения для взаимодействия пользователя с сервером. Так же здесь содержится основной обработчик события выбора кости для добавления на игровое поле. Из данного класса через делегат мы имеем возможность вызвать и добавить сведения об пользователе. Данный класс наследует интерфейс обратной связи с сервером и реализует его методы;
* LoginWindow – данный класс служит для инициализации, получения и добавления информации о пользователе;
* BoneButton – основной класс для графического представления кости.

Основные классы и интерфейсы приложения Сервер:

* Bone –содержит информацию о костяшке;
* Player – данный класс служит для инициализации пользователя;
* Game – класс для решения логических задач игрового процесса, таких как выявление первого хода партии, окончание партии или игры;
* Program – основной класс для запуска службы как сервера;
* IDominoes (интерфейс) и Dominoes (класс) – основной интерфейс и класс, который его реализует. Обрабатывает всю информацию поступающую с клиента и отправляющие методы обратного вызова;
* IDominoesCallback – интерфейс, в котором описываются методы обратного вызова, такие как обновление полей, информацию о смене хода или завершению партии или игры;

Рисунок 3-2. Диаграмма классов приложения Клиент

Рисунок 3-3. Диаграмма классов приложения Сервер

4. Разработка интерфейса системы

Общие принципы разработки интерфейса программы клиент:

* Интерфейс должен быть интуитивно понятным. Таким, чтобы пользователю не требовалось объяснять как им пользоваться;
* Чем быстрее человек увидит результат — тем лучше;
* Перед расположением, интерфейс следует упорядочить(сгруппировать) по значимости. Т.е. определить, какие наиболее важны, а какие — менее;
* Избегать двусмысленности;
* Монотонный интерфейс — это интерфейс, в котором какое-то действие, можно сделать только одним способом. Такой подход обеспечит быструю привыкаемость к программе и автоматизацию действий;
* Необходимо учитывать привычки пользователя. Например, если в Windows кнопка закрыть находится в правом верхнем углу, то программе аналогичную кнопку необходимо расположить там же. Т.е. интерфейс должен иметь как можно больше аналогий, с известными пользователю вещами;
* Основные компоненты (часто используемые) должны быть выделены. Например размером или цветом;
* Интерфейс должен быть ориентированным на человека, т.е. отвечать нуждам человека и учитывать его слабости. Нужно постоянно думать о том, с какими трудностями может столкнуться пользователь[4].

Макеты интерфейса приложения Клиент представлены на рисунках 4-1 и 4-2.



Рисунок 4-1. Основное окно приложения Клиент

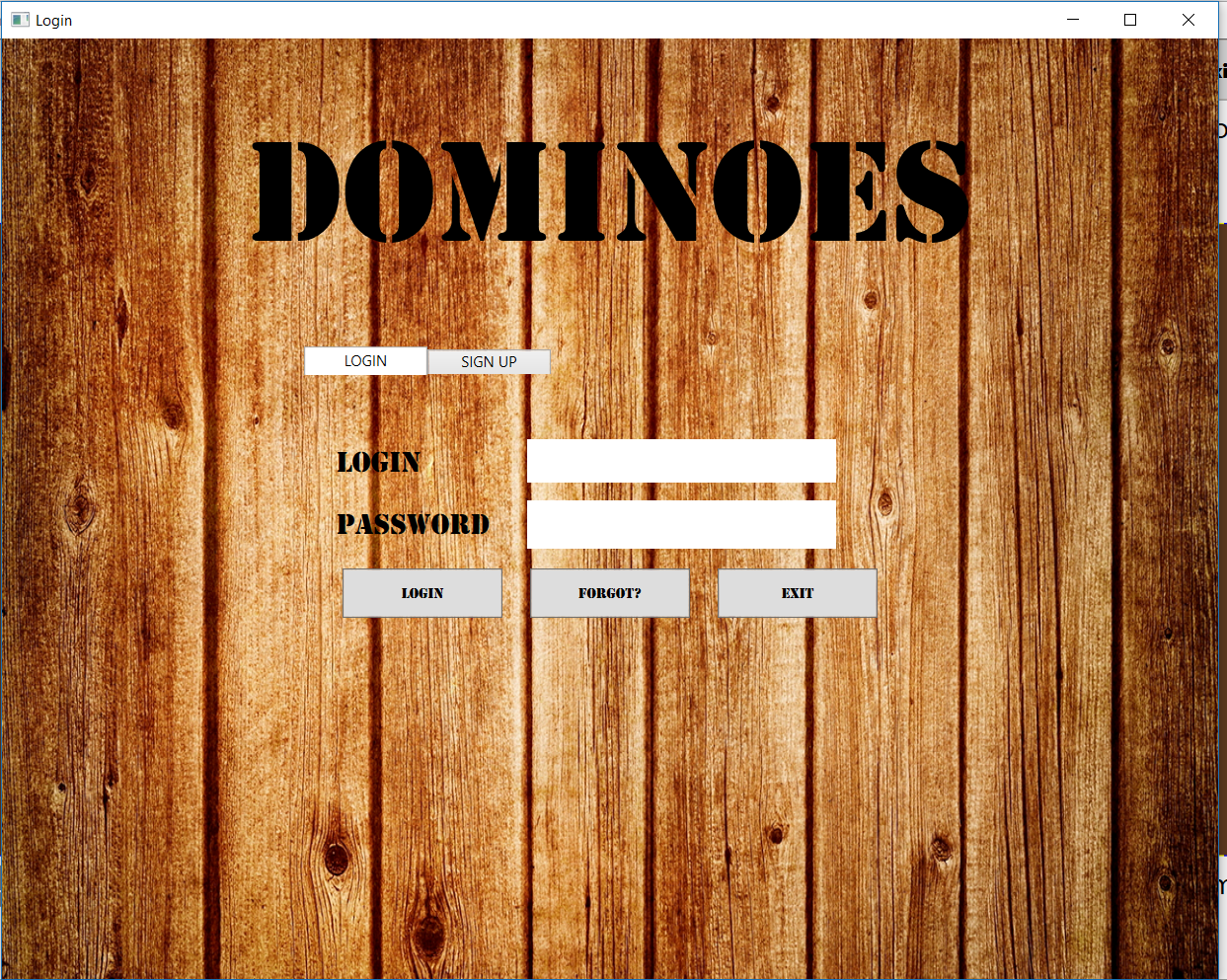


Рисунок 4-1. Окно для инициализации пользователя приложения Клиент

Выводы

В представленном курсовом проекте были рассмотрены основные понятия языка программирования C#, и прочих технологий, которые касаются графических интерфейсов, и проектирования серверных приложений.

Нами была реализована программа, которая реализует настольную игру «Домино».

Данная программа на текущий момент осуществляет игру, которой управляет несколько пользователей. В дальнейшем можно развить это игровое приложение и обеспечить возможность игры с компьютером, выбором количества соперником.

Список использованной литературы

1. Домино (настольная игра) <https://ru.wikipedia.org/wiki/%D0%94%D0%BE%D0%BC%D0%B8%D0%BD%D0%BE>
2. Сообщество разработчиков Microsoft <https://msdn.microsoft.com/uk-ua/dn308572.aspx>
3. Электронный ресурс о программировании <https://metanit.com/>
4. Проектирование графического интерфейса пользователя. [yadobr](https://habrahabr.ru/users/yadobr/). 2014 <https://habrahabr.ru/post/208966/>