Проект

„Обектно – ориентирано програмиране с C#.NET“

на тема: „Судоку“

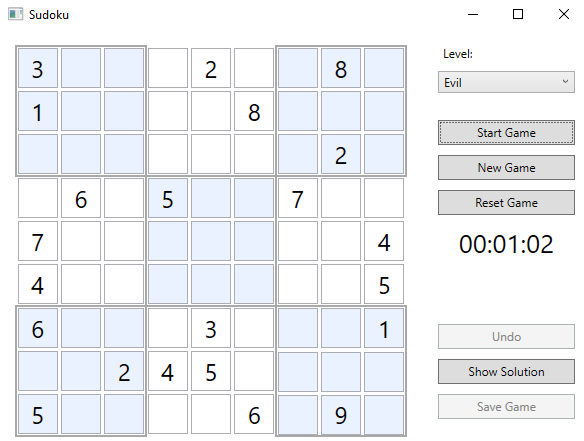
*Изготвил:*

*Ивелин Илиев, ФН: 71958, ИС 3 курс, 1-ва група*

1. **Кратко описание на проекта**

В проекта предстои да видите представянето на една популярна игра Судоку. Ребус състоящ се от 81 квадратчета, представляващи игралното поле, с предварително попълнени няколко правилни места с цифри. Целта на играта е да се запълнят останалите празни квадратчета, като във всяка група от 3х3, всеки ред и колона числата от 1 до 9 не трябва да се повтарят. В играта са зададени четири нива на трудност (лесно, средно, трудно и дяволско), за всеки един играч спрямо неговите възможности. Всеки играч по всяко време може да рестартира нивото си, да започне ново или в случай, че не може да го приключи в момента, да запази файла си и да продължи от там до където е стигнал при следващо стартиране на приложението. При затруднение с решението на ребуса винаги много да се види неговото решение, както и да се следи времето, за което играча решава пъзела. Също така всеки потребител може да зареди свой собствен пъзел от файл на неговия компютър. Приложението е направено така че при грешка от човека, неволна или не, да му се оцветява числото в зелено за вярно и в червено за грешно.

1. **XAML Файлове**

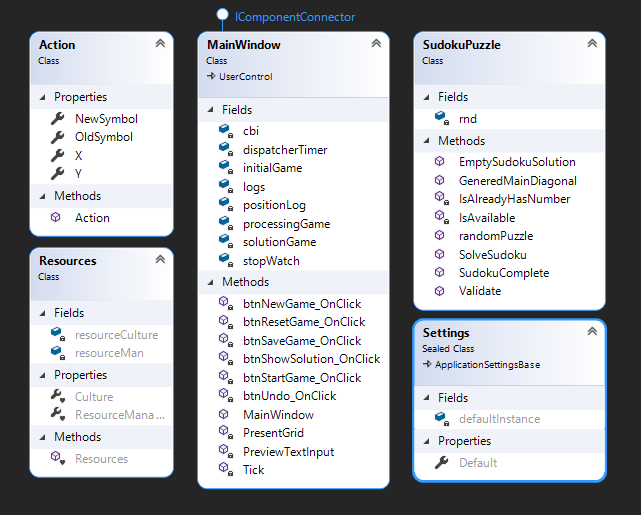


Фигура 1 - Дизайн на приложението

На снимка може да видите дизайна на приложението при стартиран на програмата и зареждането на пъзел готов за решаване от потребителя. Даден пъзел приключва при вярно решение или до зареждането на нов.

1. **Класове**

* SudokuPuzzle
  + private static Random rnd – променлива използвана за генериране на произволни числа
  + public static bool EmptySudokuSolution – метод за проверяване дали матрицата пазеща решението на пъзела е празна
  + public static bool SudokuComplete – метод за проверка дали решението до момента на потребителя е приключило и проверка дали е правилно
  + private static bool IsAlreadyHasNumber – метод използван за проверка на число дали съществува в 3х3 подматриците по главния диагонал при генериране на случаен ребус
  + public static void GeneredMainDiagonal – метод за генериране на случаен ребус
  + public static void randomPuzzle – метод, който задава определен брой случайни правилни числа зависимост от трудността на нивото.
  + public static bool SolveSudoku – метод за решаване на даден ребус
  + private static bool IsAvailable – метод за проверка на число, в 3х3 подмратрицата, реда и колоната, в които се намира, за постигане на правилно решение да даден пъзел
  + public static bool Validate – метод за валидиране на зареден файл от компютъра с готов пъзел

1. **UML Диаграма на UsetControl**

Фигура 2 - UML Diagram UserControl