**Topic of the project: “The Maze” game**

**Annotation:** We build a maze game, where the user has to move from start to finish through the walls as fast as possible. The database is created for keeping information about games history and results, which can be shared to social network using API.

**Members of the team and their roles:**

Osmanova Yana 146 - GUI, application logic  
Badaeva Elina 146 - DB, application logic

**Central repository address:** <https://github.com/d1stort/TheMaze.git>

**List of classes:**

1. public partial class MainWindow:

(private void closeButton\_Clicked, private void startButton\_Clicked, private void statsButton\_Clicked, static IQueryable<Player> LoadPlayersFromDB)

1. class PlayerViewModel:
2. public partial class RegistrationWindow : Window:

(private void CloseRegWindButton\_Click, public RegistrationWindow())

1. public partial class LoginWindow : Window:

(private void PlayButtonLW\_Click, public LoginWindow())

1. public partial class Level1 : Window:

(private static extern bool SetCursorPo, private static void SetCursor, public Level1(), private void mouseOver\_finish, private void mouseEnter\_label, private void Level\_Loaded, private void BackTMenu\_Click)

1. public partial class AccountWindow : Window:

(public AccountWindow(),private void YesButtonAc\_Click, private void NoButtonAc\_Click)

1. public class Player:

(private void OnPropertyChanged)

1. class Context: DbContext:

(public Context() : base("PlayersDb"), public DbSet<Player> Players, protected override void OnModelCreating)

1. internal sealed class Configuration : DbMigrationsConfiguration<TheMaze.Context>:

(public Configuration(),protected override void Seed, )

1. class PlayerUpdateCommand : ICommand:

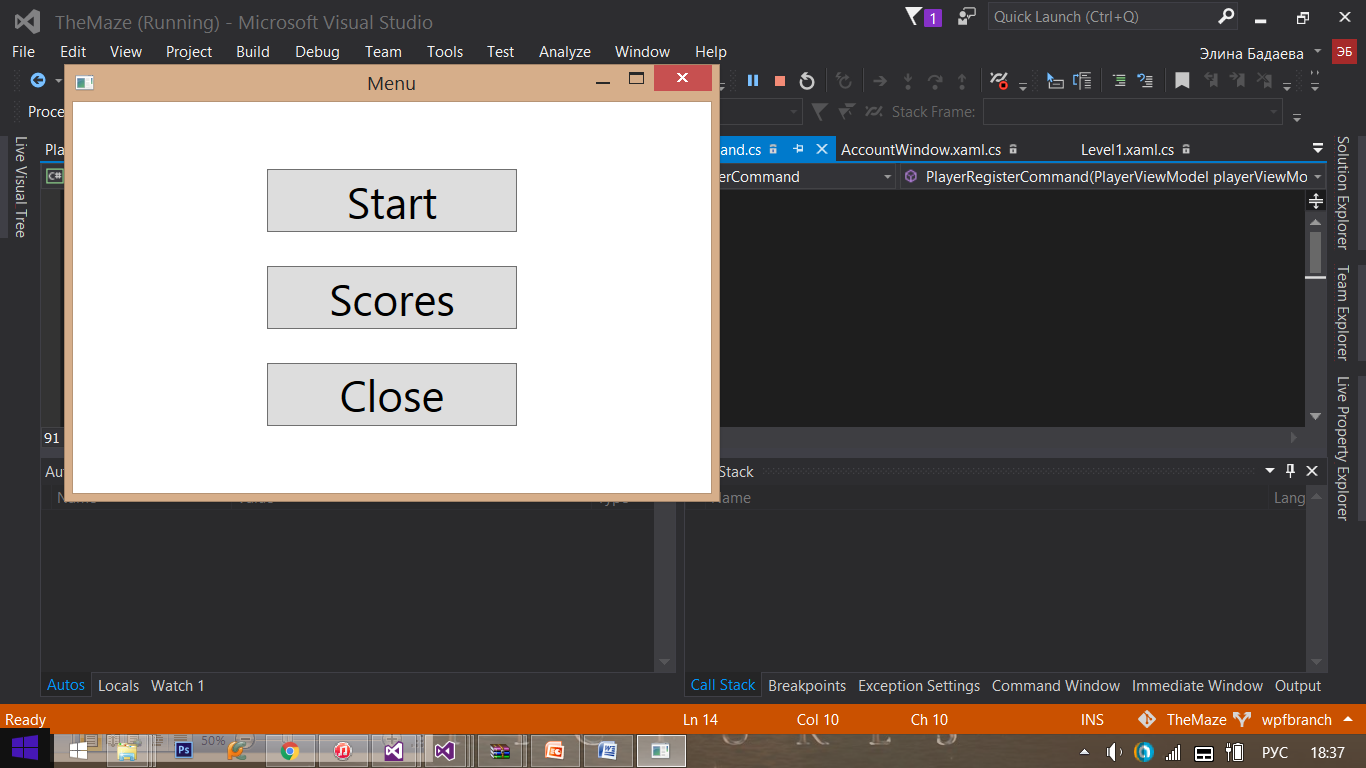
(public PlayerUpdateCommand, public bool CanExecute, public event EventHandler CanExecuteChanged, public void Execute)

1. internal class PlayerRegisterCommand : ICommand:

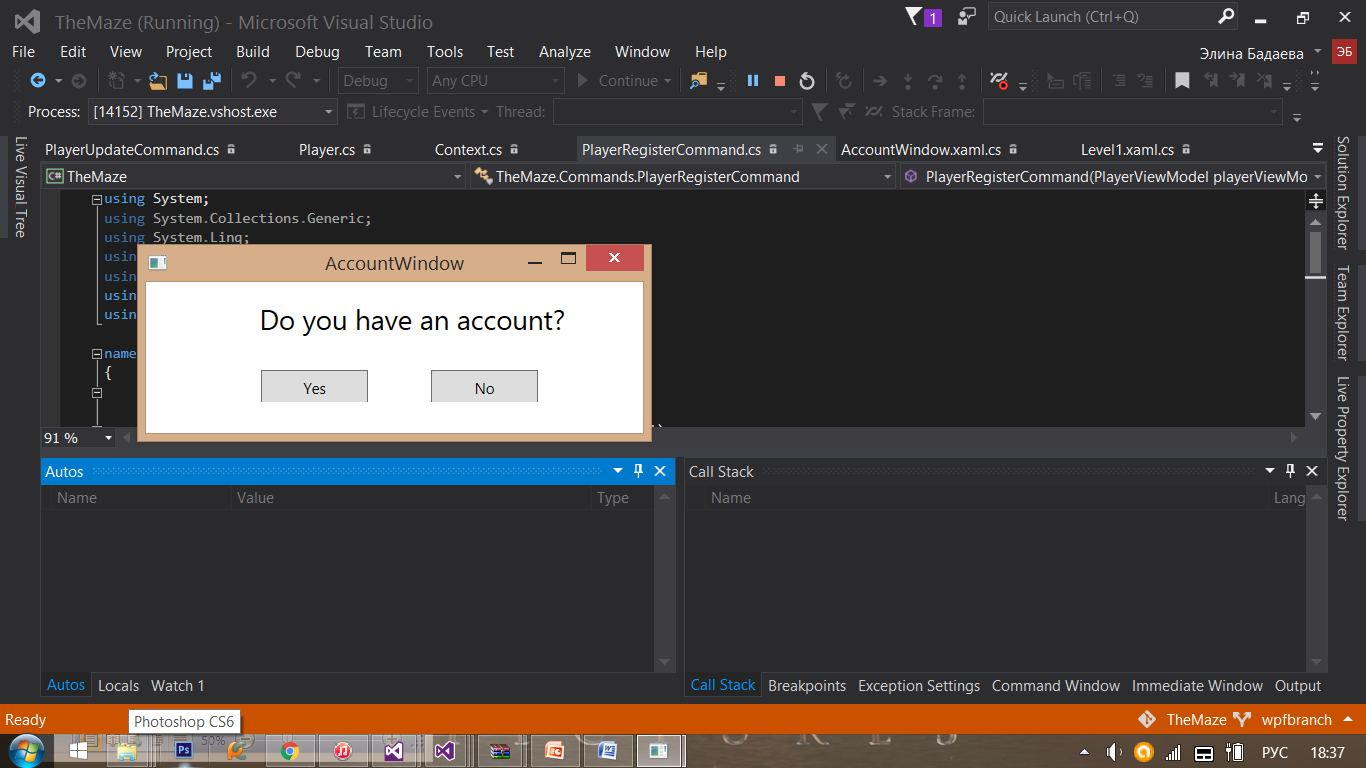
(public PlayerRegisterCommand, public bool CanExecute, public event EventHandler CanExecuteChanged, public void Execute)

**Program interface:**

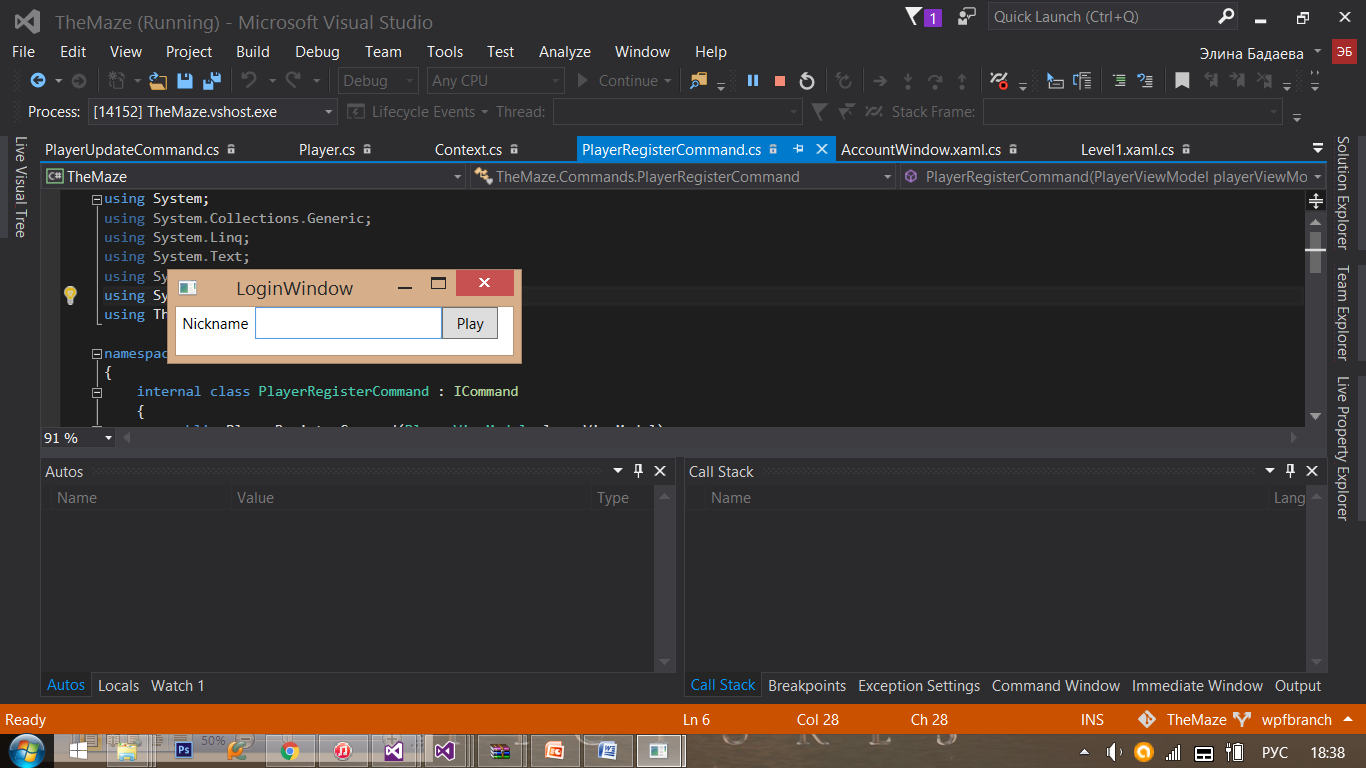
MainWindow allows to select one of three in the menu: Start, Scores, Close

****

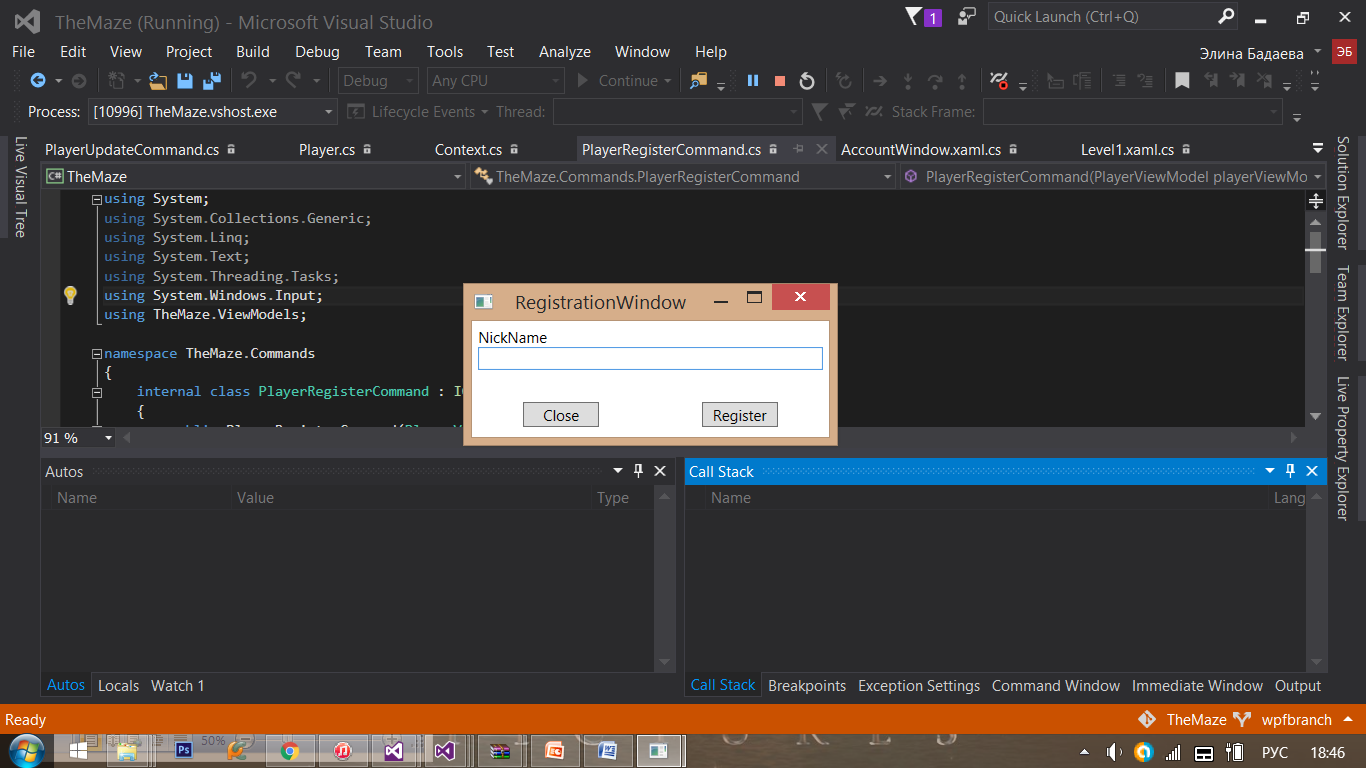
After selecting Start-button you are asked whether you have an account or not.

****

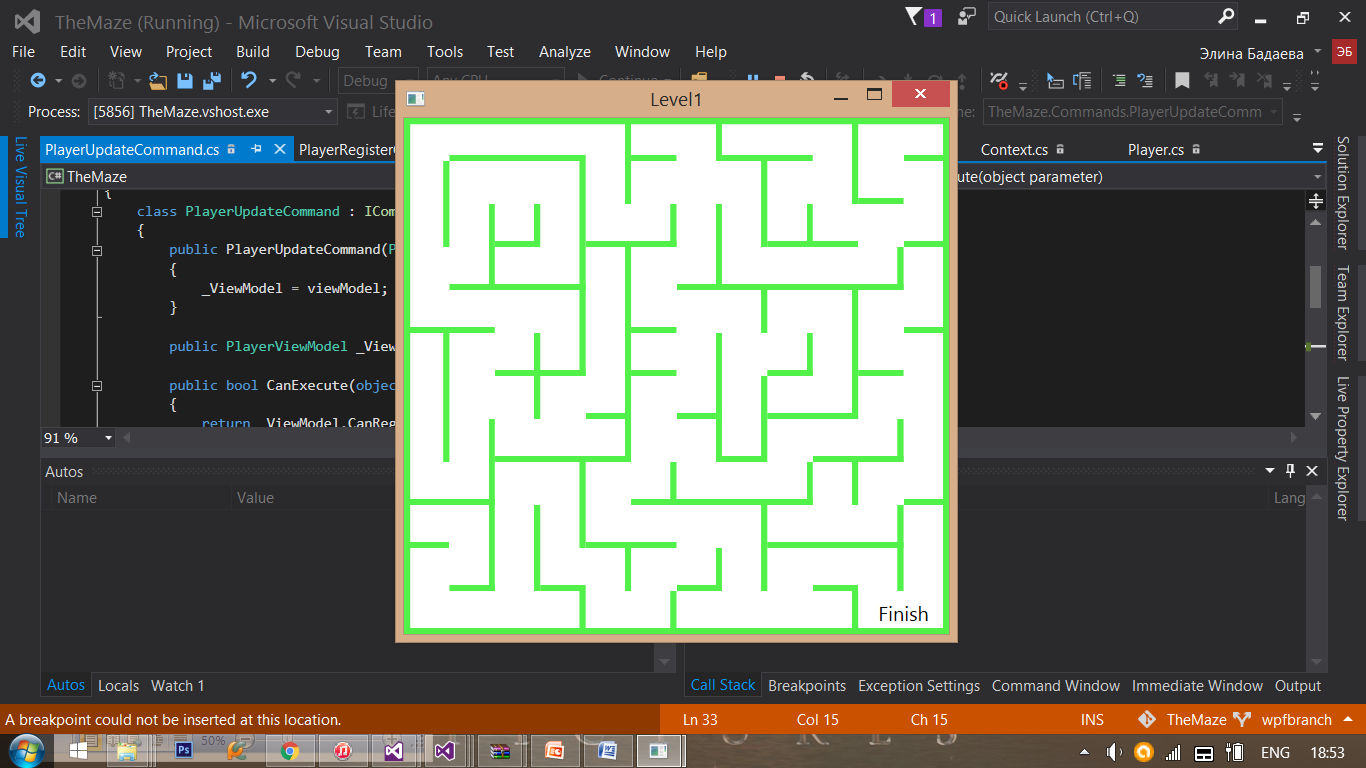
If yes:

****

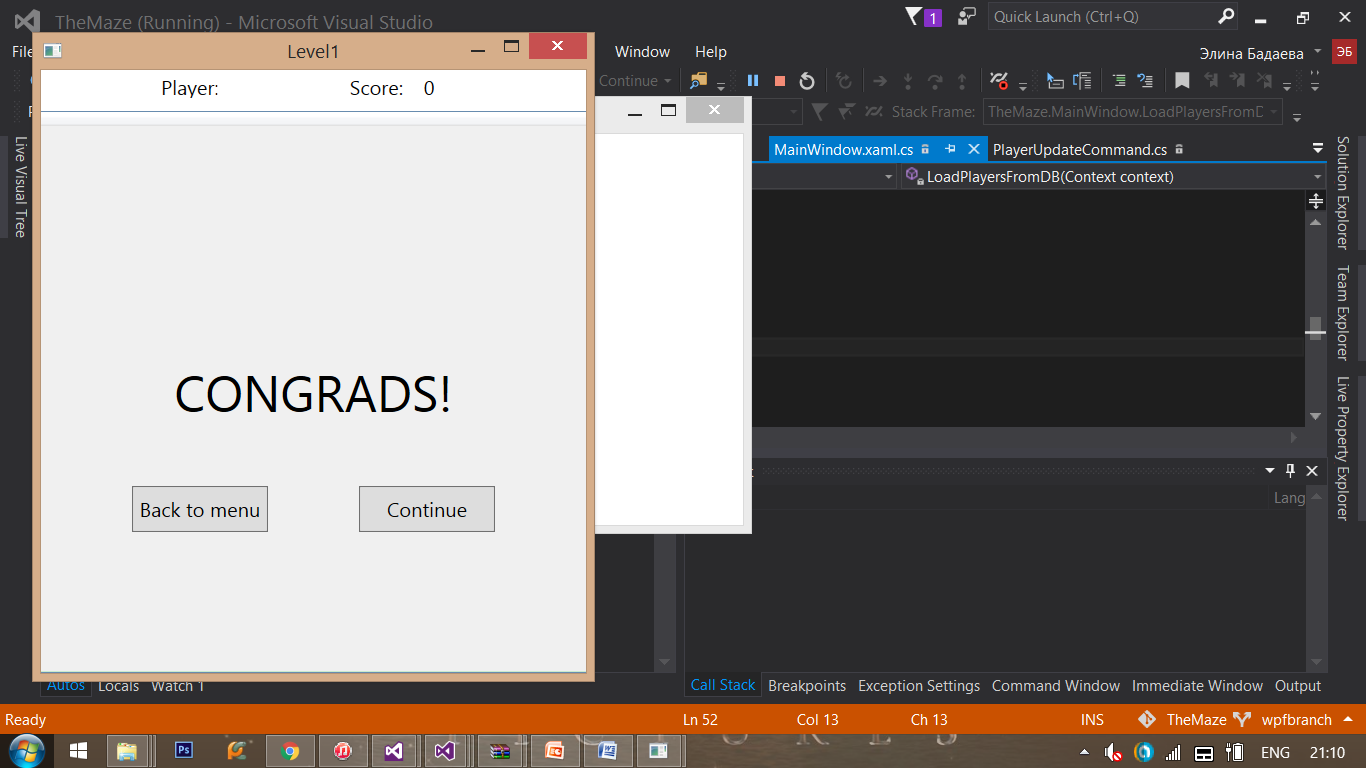
If no:

****

After successful logging in or registration you are suggested to try the first Level of the game.

****

If you win, you will get congratulations and opportunity to go back to menu or to continue the game

****