

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Net;
5 using System.Text;
6 using System.Threading.Tasks;
7 using System.Windows;
8 using System.Windows.Controls;
9 using System.Windows.Data;
10 using System.Windows.Documents;
11 using System.Windows.Input;
12 using System.Windows.Media;
13 using System.Windows.Media.Imaging;
14 using System.Windows.Navigation;
15 using System.Windows.Shapes;
16
17 namespace QuiddichV2._0
18 {
19     /// <summary>
20     /// Interaction logic for AddTeam.xaml
21     /// </summary>
22     public partial class AddTeam : Page
23     {
24         public Team team;
25         public bool addTeam;
26
27         public AddTeam()
28         {
29             InitializeComponent();
30         }
31
32         private void btnSaveTeam_Click(object sender, RoutedEventArgs e)
33         {
34             if (IsValidData())
35             {
36                 if (addTeam)
37                 {
38                     team = new Team();
39                     this.PutTeamData(team);
40                     try
41                     {
42                         QuiddichDB.AddTeam(team);
43
44                         MessageBoxResult result = MessageBox.Show("In order to play a game, " + team.
45 TeamName.Trim() + " must have at least four players."
46 + "\nDo you wish to add players now?",
47 "Add Players?", MessageBoxButton.YesNo,
48 MessageBoxImage.Question);
49
50                     switch(result)
51                     {
52                         case MessageBoxResult.Yes:
53                             AddPlayer addPlayer = new AddPlayer(team.TeamID);
54                             addPlayer.addPlayer = true;
55                             NavigationService.Navigate(addPlayer);
56                             break;
57                     }
58                 }
59             }
60             catch (Exception ex)
61             {
62                 MessageBox.Show(ex.Message, ex.GetType().ToString());
63             }
64         }
65     }
66 }
```

```
65         newTeam.TeamID = team.TeamID;
66         newTeam.TeamName = team.TeamName;
67         newTeam.AbbName = team.AbbName;
68         newTeam.City = team.City;
69         newTeam.State = team.State;
70         newTeam.ZipCode = team.ZipCode;
71
72         this.PutTeamData(newTeam);
73         try
74         {
75             if (!QuiddichDB.UpdateTeam(team, newTeam))
76             {
77                 MessageBox.Show("Another user has updated or " +
78                     "deleted that team.", "Database Error",
79                     MessageBoxButton.OK, MessageBoxImage.Error);
80             }
81             else
82             {
83                 team = newTeam;
84             }
85         }
86         catch (Exception ex)
87         {
88             MessageBox.Show(ex.Message, ex.GetType().ToString());
89         }
90     }
91
92     ManageTeams manageTeams = new ManageTeams();
93     NavigationService.Navigate(manageTeams);
94
95 }
96
97
98 private void PutTeamData(Team team)
99 {
100     team.TeamID = Convert.ToInt32(txtTeamID.Text.Trim());
101     team.TeamName = txtTeamName.Text.Trim();
102     team.AbbName = txtAbbrName.Text.Trim();
103     team.City = txtCity.Text.Trim();
104     team.State = txtState.Text.Trim();
105     team.ZipCode = txtZipCode.Text.Trim();
106
107 }
108
109 private bool IsValidData()
110 {
111     return Validator.IsPresent(txtTeamName) &&
112         Validator.IsPresent(txtAbbrName) &&
113         Validator.IsPresent(txtCity) &&
114         Validator.IsPresent(txtState) &&
115         Validator.IsPresent(txtZipCode) &&
116         Validator.IsValidZip(txtZipCode) &&
117         Validator.IsPresent(txtTeamID) &&
118         Validator.IsInt32(txtTeamID);
119 }
120
121 private void btnCancel_Click(object sender, RoutedEventArgs e)
122 {
123     ManageTeams manageTeams = new ManageTeams();
124     NavigationService.Navigate(manageTeams);
125 }
126
127 private void GetTeam(int teamID)
128 {
129     try
130     {
```

```
131         team = QuiddichDB.GetTeam(teamID);
132     }
133     catch (Exception ex)
134     {
135         MessageBox.Show(ex.Message, ex.GetType().ToString());
136     }
137 }
138
139 private void btnFindTeam_Click(object sender, RoutedEventArgs e)
140 {
141     if (IsValidData())
142     {
143         int teamID = Convert.ToInt32(txtTeamID.Text);
144         this.GetTeam(teamID);
145         if (team == null)
146         {
147             MessageBox.Show("No team found with this name. " +
148                 "Please try again.", "Team Not Found");
149         }
150         else
151         {
152             this.DisplayTeam();
153         }
154     }
155 }
156
157 private bool IsValidData()
158 {
159     bool valid = false;
160
161     if (Validator.IsInt32(txtTeamID))
162     {
163         valid = true;
164     }
165     return valid;
166 }
167
168 private void DisplayTeam()
169 {
170     txtTeamID.Text = team.TeamID.ToString();
171     txtTeamName.Text = team.TeamName;
172     txtAbbrName.Text = team.AbbbrName;
173     txtCity.Text = team.City;
174     txtState.Text = team.State;
175     txtZipCode.Text = team.ZipCode;
176 }
177
178 private void txtZipCode_TextChanged(object sender, TextChangedEventArgs e)
179 {
180     //GetInfo();
181 }
182
183 private void GetInfo()
184 {
185     //string location = new WebClient().DownloadString("http://api.ipinfodb.com/v3/ip-city/?key
186     //9bf463d68b2c27af0366950c4e4a607100143e9c798b42959ee508f73f15b509");
187     //string[] locationParams = new string[10];
188
189     //locationParams = location.Split(';');
190
191     //string city = locationParams[6].ToLower().Trim();
192     //char.ToUpper(city[0]);
193
194     //string state = locationParams[5].Trim();
195     //state = state.Substring(0,2);
```

```
196         //txtCity.Text = city;
197         //txtState.Text = state;
198     }
199 }
200 }
201
```