

## InteractionsService

---

```
1 using MGM.Web.Domain;
2 using System;
3 using System.Collections.Generic;
4 using System.Data;
5 using System.Data.SqlClient;
6 using MGM.Data;
7 using System.Linq;
8 using System.Web;
9 using MGM.Web.Models.Requests.Interactions;
10
11 namespace MGM.Web.Services
12 {
13     public class InteractionsService : BaseService, IInteractionService,
14         IInteractionMapper
15     {
16         public Int32 Add(AddInteractionRequest model, string userId, int
17             accountId)
18         {
19             int Id = 0; //000-0000-0000-0000
20
21             DataProvider.ExecuteNonQuery(GetConnection, "dbo.Interactions_Insert"
22                 , inputParamMapper: delegate (SqlParameterCollection
23                     paramCollection)
24                 {
25                     paramCollection.AddWithValue("@Name", model.Name);
26                     paramCollection.AddWithValue("@Icon", model.Icon);
27                     paramCollection.AddWithValue("@Description",
28                         model.Description);
29                     paramCollection.AddWithValue("@UserId", userId);
30                     paramCollection.AddWithValue("@AccountId", accountId);
31
32                     SqlParameter p = new SqlParameter("@Id",
33                         System.Data.SqlDbType.Int);
34                     p.Direction = System.Data.ParameterDirection.Output;
35
36                     paramCollection.Add(p);
37
38                 }, returnParameters: delegate (SqlParameterCollection param)
39                 {
40                     Int32.TryParse(param["@Id"].Value.ToString(), out Id);
41                 }
42             );
43
44             return Id;
45         }
46
47         public void AddByClient(int interactionId, string clientId)
48         {
49             DataProvider.ExecuteNonQuery(GetConnection,
```

## InteractionsService

---

```
48         "dbo.ClientInteractionRecord_Insert"
49         , inputParamMapper: delegate (SqlParameterCollection
50         paramCollection)
51         {
52             paramCollection.AddWithValue("@ClientId", clientId);
53             paramCollection.AddWithValue("@InteractionId", interactionId);
54         }
55     };
56 }
57 public void UpdateInteraction(UpdateInteractionRequest model, string
58     userId, int accountId)
59 {
60     DataProvider.ExecuteNonQuery(GetConnection, "dbo.Interactions_Update"
61     , inputParamMapper: delegate (SqlParameterCollection
62     paramCollection)
63     {
64         paramCollection.AddWithValue("@Id", model.Id);
65         paramCollection.AddWithValue("@Name", model.Name);
66         paramCollection.AddWithValue("@Icon", model.Icon);
67         paramCollection.AddWithValue("@Description",
68         model.Description);
69         paramCollection.AddWithValue("@UserId", userId);
70         paramCollection.AddWithValue("@AccountId", accountId);
71     }
72     );
73 }
74
75 public List<Interaction> Get()
76 {
77     List<Interaction> list = null;
78     DataProvider.ExecuteCmd(GetConnection, "dbo.Interactions_GetAll"
79     , inputParamMapper: null
80     , map: delegate (IDataReader reader, short set)
81     {
82         Interaction p = MapInteraction(reader);
83
84         if (list == null)
85         {
86             list = new List<Interaction>();
87         }
88
89         list.Add(p);
90     }
91     );
92
93     return list;
94 }
```

## InteractionsService

---

```
95     }
96
97     public Interaction MapInteraction(IDataReader reader, int ordinal = 0)
98     {
99         Interaction p = new Interaction();
100         int StartingIndex = ordinal;
101         p.Id = reader.GetSafeInt32(StartingIndex++);
102         p.Name = reader.GetSafeString(StartingIndex++);
103         p.Icon = reader.GetSafeString(StartingIndex++);
104         p.Description = reader.GetSafeString(StartingIndex++);
105         p.AccountId = reader.GetSafeInt32(StartingIndex++);
106         return p;
107     }
108
109     public List<Interaction> Get(int accountId)
110     {
111         List<Interaction> list = null;
112
113
114         DataProvider.ExecuteCmd(GetConnection,                                ↗
115             "dbo.Interactions_GetByAccountId"
116             , inputParamMapper: delegate (SqlParameterCollection paramCollection)    ↗
117             {
118                 paramCollection.AddWithValue("@AccountId", accountId);
119             }
120             , map: delegate (IDataReader reader, short set)
121             {
122                 Interaction p = MapInteraction(reader);
123
124                 if (list == null)
125                 {
126                     list = new List<Interaction>();
127                 }
128
129                 list.Add(p);
130
131             }
132             );
133
134         return list;
135     }
136
137
138     public void DeleteContact(int id)
139     {
140
141         DataProvider.ExecuteNonQuery(GetConnection, "dbo.Interactions_Delete"
142             , inputParamMapper: delegate (SqlParameterCollection paramCollection)    ↗
143             {
```

## InteractionsService

```
144         paramCollection.AddWithValue("@Id", id);
145     }
146 );
147
148 }
149
150 public List<ClientInteractionsByClient> GetAClientInteractions(string  ↗
    clientId)
151 {
152     List<ClientInteractionsByClient> list = null;
153     DataProvider.ExecuteCmd(GetConnection,  ↗
        "ClientInteractionRecord_GetByClientId"
154     , inputParamMapper: delegate (SqlParameterCollection  ↗
        paramCollection)
155     {
156         paramCollection.AddWithValue("@ClientId", clientId);
157
158     }, map: delegate (IDataReader reader, short set)
159     {
160         ClientInteractionsByClient r = new ClientInteractionsByClient  ↗
            ();
161         int StartingIndex = 0;
162         r.InteractionName = reader.GetSafeString(StartingIndex++);
163         r.Times = reader.GetSafeInt32(StartingIndex++);
164         r.Date = reader.GetSafeDateTime(StartingIndex++);
165
166
167         if (list == null)
168         {
169             list = new List<ClientInteractionsByClient>();
170         }
171
172         list.Add(r);
173
174     });
175     return list;
176 }
177
178
179
180
181 public void AddClientAppInteraction(InsertClientAppInteractionRequest  ↗
    model, string clientId)
182 {
183     DataProvider.ExecuteNonQuery(GetConnection,  ↗
        "dbo.ClientAppInteraction_Insert"
184     , inputParamMapper: delegate (SqlParameterCollection  ↗
        paramCollection)
185     {
186         paramCollection.AddWithValue("@ClientId", clientId);
187         paramCollection.AddWithValue("@InteractionType",  ↗
            model.InteractionType);
```

## InteractionsService

---

```
188
189     });
190 }
191
192 public List<InteractionUserReportData> GetClientAppInteractionReport(int accountId) ➤
193 {
194     List<InteractionUserReportData> list = null;
195     DataProvider.ExecuteCmd(GetConnection, ➤
196         "dbo.ClientAppInteraction_Report" ➤
197         , inputParamMapper: delegate (SqlParameterCollection ➤
198             paramCollection) ➤
199         {
200             paramCollection.AddWithValue("@AccountId", accountId);
201         }, map: delegate (IDataReader reader, short set)
202
203         {
204             InteractionUserReportData r = new ➤
205             InteractionUserReportData();
206             int StartingIndex = 0;
207             r.Id = reader.GetSafeInt32(StartingIndex++);
208             r.UserId = reader.GetSafeString(StartingIndex++);
209             r.FirstName = reader.GetSafeString(StartingIndex++);
210             r.LastName = reader.GetSafeString(StartingIndex++);
211             r.Email = reader.GetSafeString(StartingIndex++);
212             r.DateLastInteracted = reader.GetSafeDateTimeNullable ➤
213             (StartingIndex++);
214             r.DaysWithoutInteraction = reader.GetSafeInt32Nullable ➤
215             (StartingIndex++);
216
217             if (list == null)
218             {
219                 list = new List<InteractionUserReportData>();
220             }
221
222             list.Add(r);
223
224         });
225     return list;
226 }
```