

Developer Guide – RateItAll.com API Platform

Version 1 (<http://api.rateitall.com/v1>)

Last updated on 5/14/2009



Table of Contents

Introduction	4
Recent Updates.....	5
API Architecture	6
Base Structure of Services.....	7
List of all Services	8
Supported Protocols (Request/Response Formats)	8
Sign Up Process.....	9
Sandbox/Testing Environment.....	9
Test Tool for XML POST.....	9
How to Request New Functionalities and Report bugs	10
Concepts	10
Version Support	10
How to read the API Response	10
Understanding the Access issues on Create/Updates	12
REST Metadata.....	12
Understanding the REST Metadata.....	12
Important Information Regarding the XML POST.....	13
SOAP WSDL	14
Understanding various Objects.....	14
Object – User.....	14
Object – Subject	15
Object – Topic	15
Object – Item	17
Object – ItemLink [There is no separate service for ItemLinks, but will be used as part of Items object in ItemService)	19

Object – Business [Business object is an Extension of Item Object. So it has all the properties of Item along with the below ones too]	20
Object – ItemRating	21
Object – ItemRatingComment	22
Sample Request and Response (In XML).....	24
Service: UserService	24
Service: Subject Service	28
Service: Topic Service.....	29
Service: Item Service	33
Service: ItemRating Service.....	41
Service: ItemRating Service (Vote Functionality).....	49
Service: ItemRatingComment Service.....	51

Introduction

Rateitall.com exposes some of its functionality through the API (Application Programmable Interface). This document will explain all the available features along with the specifications, which will help the developer to build tools using RateItAll.com.

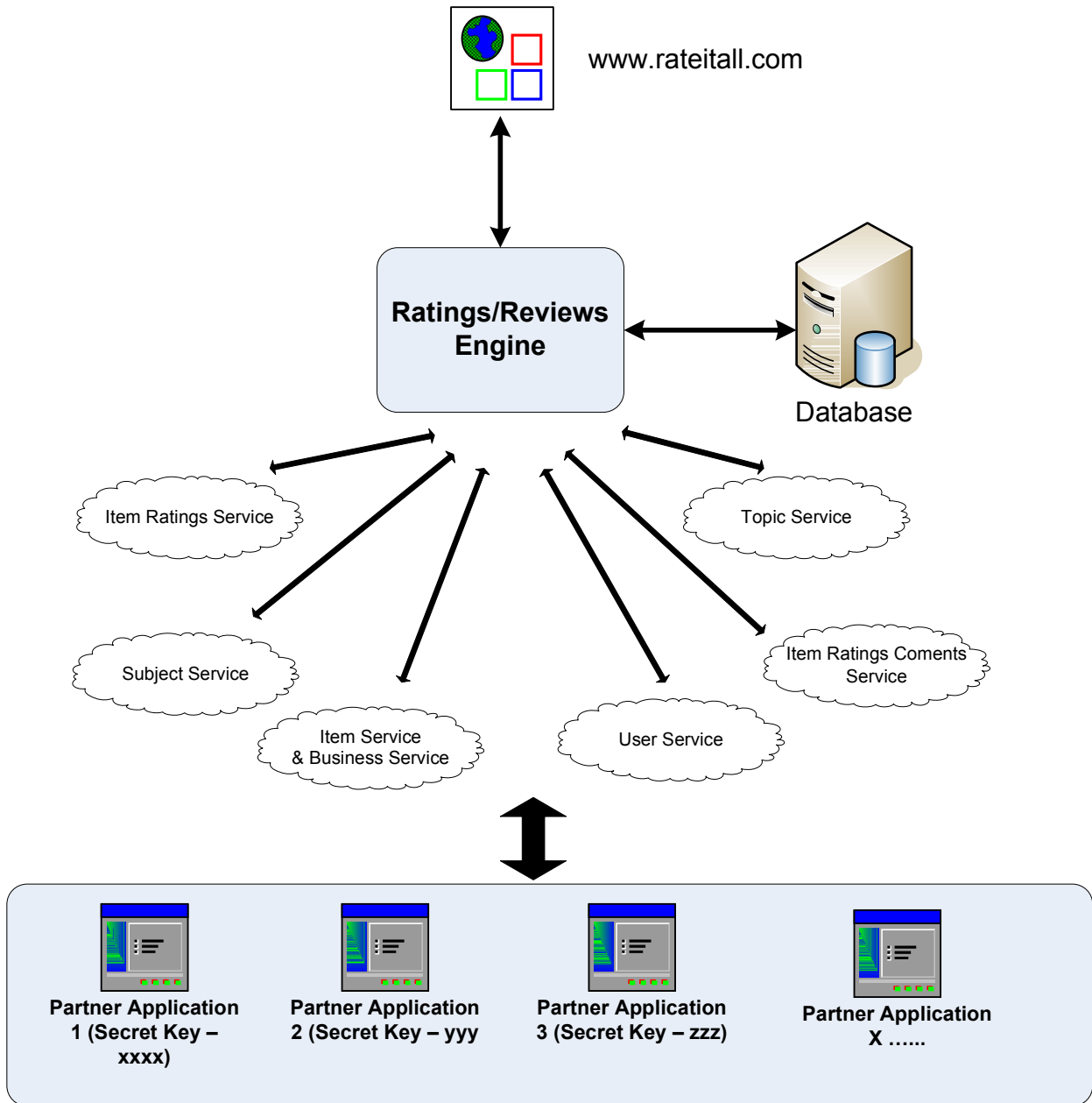
The API has been built so that the developers will have a standard complaint to access RateItAll data. The API will facilitate to access the existing Topics, Items, Reviews, Ratings, Comments, and Votes etc. The API will also have a writeable option at all levels.

Recent Updates

05/14/2008 – (Geo Location Details) Latitude and Longitude is now part of the Business object. New Business Inserts or Updates can include the Geo details. The Geo Location details will also be available on the fetch operations.

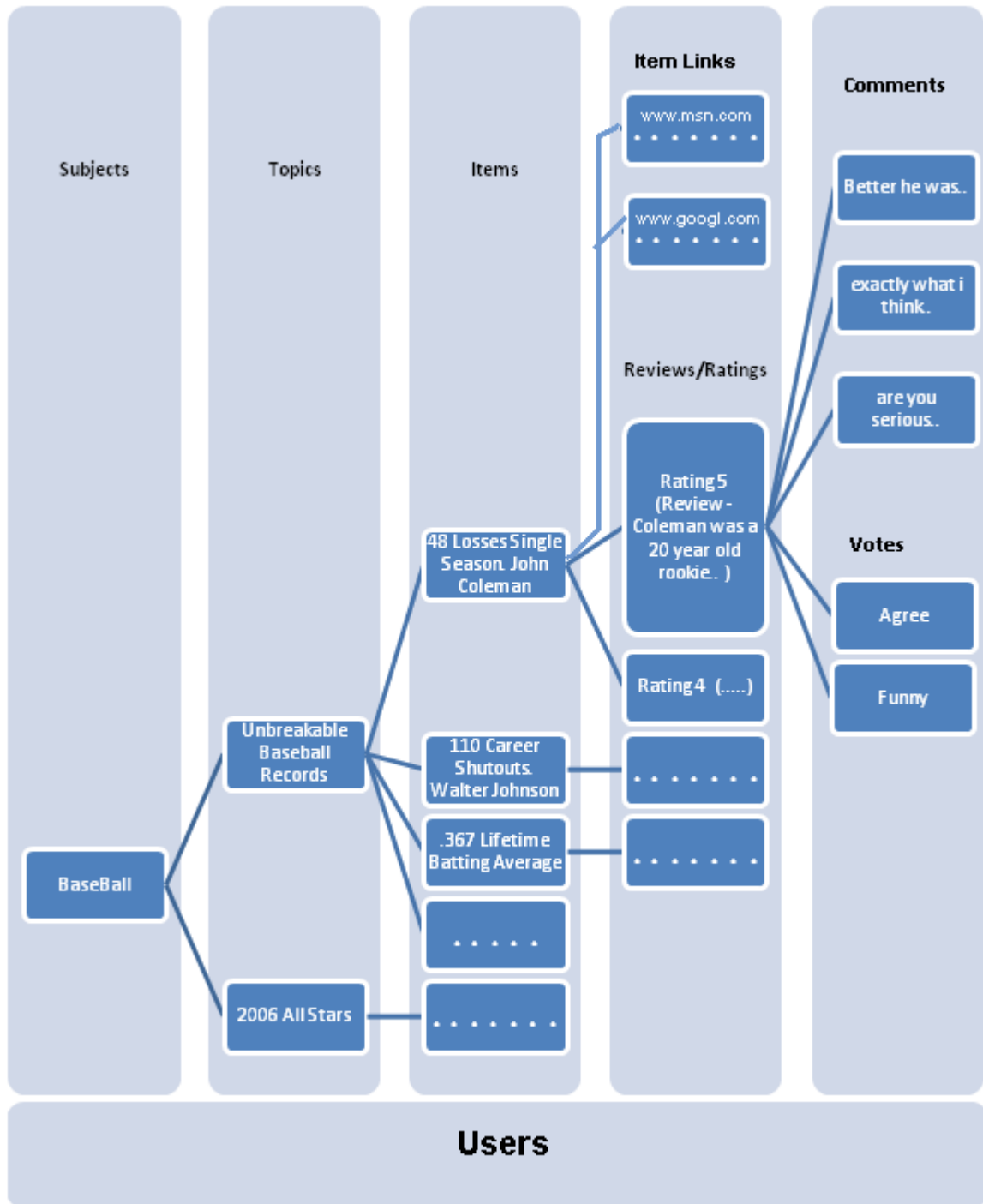
05/12/2008 – Implemented the Auto Topic Lookup Functionality for Business Service. So if you want to add a new Business, you just need to specify the Business Type (ex: “pizza restaurant” or “gym”) along with the City, State and Country (ex: San Francisco, CA, USA). API will do an auto lookup for the matching Topic for the Business Type (ex: “San Francisco Pizza Restaurants”). This will eliminate the step for creating a new Topic.

API Architecture



Base Structure of Services

The below image will explain how the data is distributed across RateItAll.com, the relationship between data elements. Each data element is tagged with Application ID, so the applications need not to worry on the data integrity issues.



List of all Services

By design RateItAll.com hosts multiple services, with each service to handle specific data. Below is the list of available services.

1. User Service (Complete User Administration)
2. Subject Service (List of Available Subjects on which a Topic can be Created)
3. Topic Service (Manage Topics)
4. Item Service (Manage Items, for ex: Pizza is the Item for the Topic "Italian Foods"). This service also includes methods to handle the operations for Local Business Too.
5. ItemRating Service (Maintains the Ratings/Reviews for each Item). This service can also handle the Voting for each Item Review/Rating.
6. ItemRatingComment Service (Maintain the Comments for the Item Reviews/Ratings)

Supported Protocols (Request/Response Formats)

Right now the API platform has been designed to Support

1. SOAP (with XML) – The SOAP/XML will be primary for all .Net and JAVA Applications. The API platform itself has been built on .Net WCF framework.
2. REST (XML/JSON) - The RateItAll API conforms mostly to the design principles of Representational State Transfer (REST). We currently XML/JSON data formats on the REST.

The document will give a Detailed walkthrough on each of these protocols and the access strategies.

Base URL structure will look like this

<http://api.rateitall.com/v1/{SOAP/REST/{XML/JSON}}{Users/Topics/Items/Itemratings/ItemRatingComments/Subjects}?SecretKey={key}>



Sign Up Process

To Signup for RateItAll API, please go to <http://api.rateitall.com>.

On after the Completion of Signup process, each Application will be provided with a Developer Secret Key and Application Key. They are unique and of 32 char length.

A Sample Secret Key and Application Key will look as follows

Secret Key: DD451469-F26E-4403-95E0-145DB4B51CFF

Application Key: 5FEA19F2-1061-4E8D-A358-9E070DE8B2CC

The Secret Key has to be used as part of the Service URL and the authentication is based on the Secret Key. No special headers are necessary. The Application key is used to tag the records, so that each Item, Reviews, Ratings etc created by Application will be marked with the Application ID.

The developer can also use this Application ID to fetch all the records Created/Updated by the same.

Sandbox/Testing Environment

We have a fully functional Sandbox setup. Please get your application working on the sandbox first before using the Live setup.

Sandbox URL: <http://api.rateitall.com/sandbox>

The Base URL Structure for Sandbox will look like

<http://api.rateitall.com/sandbox/{SOAP/REST/{XML/JSON}}/{Users/Topics/Items/Itemratings/ItemRatingComments/Subjects}?SecretKey={key}>

The Application Key and Secret Key created as part of Signup process will remain same for both Sandbox and Live environments.

Test Tool for XML POST

We currently have a Test Tool for Debugging XML POST. This page will help you to test API by posting the XML requests for Create and Update requests using JavaScript. Please make sure to enter your secret key first to proceed.

<http://api.rateitall.com/sandbox/debug.aspx>



How to Request New Functionalities and Report bugs

Most of the issues have been discussed on the Google Groups - RateItAll API Developer support.
<http://groups.google.com/group/rateitall-api>

Please go through the posts before posting a new bug.

Please write to Lawrence@rateitall.com for any new feature requests.

Concepts

Version Support

At any point of time RateItAll.com API can supports versions, the Service URL will differ for each versions. Right now API has only one version which is V1.

SOAP/XML for Users Service in Version 1:

<http://api.rateitall.com/v1/SOAP/XML/Users?SecretKey={SecretKey}>

REST/XML for Fetch User By Email Address in Version 1

<http://api.rateitall.com/v1/REST/XML/Users/EmailAddress/yyy@email.com?SecretKey={SecretKey}>

How to read the API Response

[Version 1]

Irrespective of the type of access (SOAP/REST), the response will follow the standard format.

This is how the Response looks in XML

```
<Response xmlns="http://api.rateitall.com/types"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <User>
      <UserId>26266713</UserId>
      <UserName>vijai</UserName>
      <IsDisabled>>false</IsDisabled>
      <ImageUrl>http://ria-user.s3.amazonaws.com/11b25abc-0248-405d-bea3-
        5decb5e29f1b-x80.jpg</ImageUrl>
      <EmailAddress i:nil="true" />
      <NewPassword i:nil="true" />
      <CurrentPassword i:nil="true" />
    </User>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```



```
{"Value": [{"UserId": 26266713, "UserName": "vijai", "IsDisabled": false, "ImageUrl": "http://ria-user.s3.amazonaws.com/11b25abc-0248-405d-bea3-5decb5e29f1b-x80.jpg", "EmailAddress": null, "NewPassword": null, "CurrentPassword": null}], "Message": null, "IsSuccess": true}
```

Elements on the Response Object

1. Value – This will hold the returned object. In the previous example the User object has been returned as part of the response. This value object can also hold collection of Objects which could be Item, User, or Topic etc.
2. IsSuccess – If the API successfully processed the request, then the IsSuccess flag will be set to true.
3. Message – If there is an Error or Exception on the process then the Message field will hold the Exception information. Most of the time this element will be available with Response if the IsSuccess is false.

Note: For Creating New Business, Update Business and Search Business, the XML response will look like below (The Output object will be “BusinessResponse” instead of “Response”). There will no change in the JSON response.

```
<BusinessResponse xmlns="http://api.rateitall.com/types"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <User>
      <UserId>26266713</UserId>
      <UserName>vijai</UserName>
      <IsDisabled>>false</IsDisabled>
      <ImageUrl>http://ria-user.s3.amazonaws.com/11b25abc-0248-405d-bea3-
        5decb5e29f1b-x80.jpg</ImageUrl>
      <EmailAddress i:nil="true" />
      <NewPassword i:nil="true" />
      <CurrentPassword i:nil="true" />
    </User>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</BusinessResponse>
```

Understanding the Access issues on Create/Updates

[Version 1]

Although RateItAll.com API allows to create unlimited number of new Objects, the Updates are restricted to only the objects which are created by the current application.

So if 'Application1' creates the User 'User1', then that user cannot be updated from 'Application2'. All the Create and Updates from the API are tracked using the ApplicationId.

REST Metadata

[Version 1]

As per REST standards the RateItAll.com API supports both GET/POST version of REST. For Get requests the Parameters will be passed in the URL to perform the specific operations. To understand all the operations supported for each service, please go through the following URL

REST URL/Operation Combinations:

UserService: <http://api.rateitall.com/v1/REST/XML/Users/help>

SubjectService: <http://api.rateitall.com/v1/REST/XML/subjects/help>

TopicService: <http://api.rateitall.com/v1/REST/XML/topics/help>

ItemService: <http://api.rateitall.com/v1/REST/XML/Items/help>

Note: Item Service handles the Local Business operations too

ItemRatingsService: <http://api.rateitall.com/v1/REST/XML/ItemRatings/help>

ItemRatingsCommentService: <http://api.rateitall.com/v1/REST/XML/itemratingcomments/help>

Understanding the REST Metadata

[Version 1]

Each REST metadata document will explain the list of all the available operations along with the URL format.



For example the ItemRatingComments Metadata has an Operation format

Example 1:

*Operation Type(GET) : REST url format :
ItemRatingComments/ApplicationId/{sApplicationId}/UserId/{sUserId}, Response
Format: Xml*

Below are the specs of the above operation

1. Operation Type – GET, which is also the Request type through Http.
2. Response Format - XML
3. REST URL Format – Explains what function this particular operation performs. In the above example, it returns all the ItemRatingComments for the ApplicationId and UserId.

The base URL will also look like

<http://api.rateitall.com/v1/REST/XML/ItemRatingComments/ApplicationId/{xxx}/UserId/{yy}?SecretKey={SecretKey}>

Example 2:

*Operation Type(Invoke) : REST url format : ItemRatingComments/Update,
RequestFormat : Json, Response Format: Json, Operation: POST*

Below are the specs of the above operation

1. Operation Type – POST, which is also the Request type through Http.
2. Response Format -JSON
3. REST URL Format – Updates ItemRatingComments object on POST.

The base URL will also look like

<http://api.rateitall.com/v1/REST/XML/ItemRatingComments/ApplicationId/{xxx}/UserId/{yy}?SecretKey={SecretKey}>

Important Information Regarding the XML POST

[Version 1]

The RateItAll API is strict on the sequence of the elements on the XML during the POST operation. Please use the Objects table on the later part of the document to identify the element sequence.



SOAP WSDL

[Version 1]

The WSDL (Web Service Description Language) file can be generated for each service from the below URL's. Before Generating the WSDL file, make sure you have the proper secret Key, since the SOAP Address location in each WSDL file is specific for each Client.

UserService: <http://api.rateitall.com/v1/SOAP/XML/Users?SecretKey={SecretKey}&wsdl>

SubjectService: <http://api.rateitall.com/v1/REST/XML/subjects?SecretKey={SecretKey}&wsdl>

TopicService: <http://api.rateitall.com/v1/REST/XML/topics?SecretKey={SecretKey}&wsdl>

ItemService (BusinessService):

<http://api.rateitall.com/v1/REST/XML/Items?SecretKey={SecretKey}&wsdl>

ItemRatingsService: <http://api.rateitall.com/v1/REST/XML/ItemRatings?SecretKey={SecretKey}&wsdl>

ItemRatingsCommentService:

<http://api.rateitall.com/v1/REST/XML/itemratingcomments?SecretKey={SecretKey}&wsdl>

Understanding various Objects

[Version1]

This section will give a complete overview on different objects used across the RateItAll.com services and how to assign data to these objects (or read through these objects). These objects will be primary for the RateItAll.com API, since the Service responses comprises of either one of these objects or a collection of objects. Any Create operation should have one of these objects on the POST operation.

Object - User

Name	Data Type	Description	Writable	Is Mandatory (Create/Updates)	Max Length
UserId	Int	Represents the ID of the User. This is auto-generated value	No	No-Create Yes - Updates	
UserName	string	Unique User Name assigned to Each User. This value is unique across all RateItAll.com Users.	Yes	Yes	50
IsDisabled	Bool	Specified whether the User is Active or not.	Yes	Yes	
ImageUrl	string	The profile Image url of the user. If the ImageUrl has	Yes	No	



		been Specified on Create/Update operations, then the image will be streamed and stored into our Server. So the URL in the Response will be different			
EmailAddress	String	Email Address of the User	Yes	Yes	150
New Password	String	On Creating new Users, the New Password will set the password the users. On Update operations the New password has to be specified along with the Current password value.	Yes	Yes	50
Current Password	String	This value has to be set on Update User process, where the Current Password will be matched against the existing password before the New password is set.	No	No-Create Yes - Update	50

Object – Subject

(As of now the Subjects cannot be created or updated through service)

Name	Data Type	Description	Writable
SubjectId	Int	Represents the ID of the Subject. This is auto-generated value	No
Name	string	Subject Name	No

Object – Topic

Name	Data Type	Description	Writable	Is Mandatory (Create/Updates)	Max Length
TopicId	Int	Represents the ID of the Topic. This is auto-generated value	No	No-Create Yes-Update	
TopicIdentifierId	string	A custom identifier ID can be specified	Yes	No	450

		on each Topic. This value should be unique across the Application.			
SubjectId	Int	The Subject ID to which this Topic belongs	Yes	Yes	
Name	String	Name of the Topic	Yes	Yes	100
Description	String	Topic Description	Yes	Yes	No Limit
ImageUrl	String	Each topic can have Image. If the ImageUrl has been Specified on Create/Update operations, then the image will be streamed and stored into our Server. So the URL in the Response will be different	Yes	No	
IsDisabled	Bool	Specifies whether the Topic is active or not	Yes	Yes	
OwnerUserName	String	The Owner User who creates the Topic. The OwnerUserName will be UserName in the User Object. If this value is specified then it refers the User exists in RateItAll.com Users list	Yes	Either OwnerUserName or OwnerUserIdentifierId Should exist	50
OwnerUserIdentifierId	String	Incase if you don't want to deal with the unique User Name or Creating Users through UserService, then this value should be set. This is a unique value for each application. This value can be Users	Yes	Either OwnerUserName or OwnerUserIdentifierId Should exist	450

		Email Address or IP-Address etc.			
TitleTags	String	SEO (Search Engine optimization)-This value will be used for search engine queries and be assigned in the URL.	Yes	No	No limit
Metadescription	String	SEO-Description of the Topic which will be indexed by Search Engines	Yes	No	No Limit
MetaKeywords	string	SEO-Keywords used on the Search Engine Search	Yes	No	No Limit
TopicUrl	string	Landing page for the specific topic on Rateitall.com	No	No	No Limit

Object - Item

Name	Data Type	Description	Writable	Is Mandatory (Create/Updates)	Max Length
ItemId	Int	Represents the ID of the Item. This is auto-generated value	No	No-Create Yes-Update	
ItemIdentifierId	string	A custom identifier ID can be specified on each Item. This value should be unique across the Application.	Yes	No	450
TopicID	Int	The Topic ID to which this Item belongs	Yes	Yes	
Name	String	Name of the Item	Yes	Yes	250
Description	String	Item Description	Yes	Yes	No Limit
ImageUrl	String	Each item can have Image. If the ImageUrl has been Specified on Create/Update operations, then	Yes	No	

		the image will be streamed and stored into our Server. So the URL in the Response will be different			
IsDisabled	Bool	Specifies whether the item is active or not	Yes	Yes	
OwnerUserName	String	The Owner User who creates the item. The OwnerUserName will be UserName in the User Object. If this value is specified then it refers the User exists in RateItAll.com Users list	Yes	Either OwnerUserName or OwnerUserIdentifierId Should exist	50
OwnerUserIdentifierId	String	Incase if you don't want to deal with the unique User Name or Creating Users through UserService, then this value should be set. This is a unique value for each application. This value can be Users Email Address or IP-Address etc.	Yes	Either OwnerUserName or OwnerUserIdentifierId Should exist	450
TitleTags	String	SEO (Search Engine optimization)-This value will be used for search engine queries and be assigned in the URL.	Yes	No	No-Limit
Metadescription	String	SEO-Description of the Topic which will be indexed by Search Engines	Yes	No	No-Limit
MetaKeywords	string	SEO-Keywords used on the Search Engine Search	Yes	No	No-Limit

AverageRating	String	Details the Average Rating based on the Number of Ratings. The Format will be “<<AverageRating>> out of <<Number of Ratings>> Ratings” Ex: “4.98 out of 10 Ratings”	No	No	No-Limit
ItemLinks	Collection	This contains a collection of ItemLink, which specifies the related links/urls for the specified item. Please refer to the next ItemLink object for further details	Yes	No	
ItemUrl	String	Landing page for the specific item on RateItAll.com	No	No	No Limit

Object – ItemLink [There is no separate service for ItemLinks, but will be used as part of Items object in ItemService]

Name	Data Type	Description	Writable	Is Mandatory (Create/Updates)	Max Length
Url	string	Url of the related Item	Yes	Yes	1000
Name	string	Corresponding Name tag for the specified url	Yes	Yes	100

Object - Business [Business object is an Extension of Item Object. So it has all the properties of Item along with the below ones too]

Name	Data Type	Description	Writable	Is Mandatory (Create/Updates)	Max Length
Type	String	Business Type (ex: 'gym', 'greek restaurants')	Yes	Either TopicId or Type can be specified on Business Creation or Updates	300
Address1	string	Address1 of the Business Location	Yes	Yes	255
Address2	string	Address2 of the Business Location	Yes	No	255
Suite	string	Suite # of the Business Location	Yes	No	100
City	string	City of the Business Location	Yes	Yes	255
State	string	State of the Business Location	Yes	Yes	100
ZipCode	string	ZipCode of the Business Location	Yes	No	10
Country	String	Country	Yes	Yes	100
Phone	String	Business Phone Number	Yes	No	50
Fax	String	Business Fax Number	Yes	No	50
Email	String	Business Email Address	Yes	No	100
Latitude	Decimal	Latitude of the Business, to locate on Maps	Yes	No	Range(-90 to 90) with max of 7 decimals
Longitude	Decimal	Longitude of the Business, to locate on Maps	Yes	No	Range(-180 to 180) with max of 7 decimals

Object - ItemRating

Name	Data Type	Description	Writable	Is Mandatory (Create/Updates)	Max Length
RatingId	int	Represents the ID of the Rating. This is auto-generated value	No	No-Create Yes-Update	
ItemId	Int	The ItemId to which the Rating has been added to.	Yes	Yes	
ItemIdentifierId	string	The custom identifier ID which is used on the Item creation.	Yes	No	450
Item	Object	If the Item has to be created/Updated along with this ItemRating, then use this Object.	Refer to Item object	Refer to Item object	
Rating	int	Rating for the Item (should be between 1 to 5)	Yes	Yes	
Comments	Strings	Review	Yes	Yes	No Limit
OwnerUserName	String	The Owner User who creates/updates the ItemRating. The OwnerUserName will be UserName in the User Object. If this value is specified then it refers the User exists in RateItAll.com Users list	Yes	Either OwnerUserName or OwnerUserIdentifierId Should exist	50
OwnerUserIdentifierId	String	Incase if you don't want to deal with the unique User Name or Creating Users through UserService, then this value should be	Yes	Either OwnerUserName or OwnerUserIdentifierId Should exist	450

		set. This is a unique value for each application. This value can be Users Email Address or IP-Address etc.			
FunnyVotes	Int	No of votes for this Item Rating voted as "Funny"	No	No	
AgreeVotes	int	No of votes for this Item Rating voted as "Agree"	No	No	
HelpfulVotes	int	No of votes for this Item Rating voted as "Helpful"	No	No	
DisagreeVotes	Int	No of Votes for this Item Rating voted as "Disagree"	No	No	
RatingUrl	String	Landing Page for the specific Item Rating on RateItAll.com	No	No	No Limit

Object - ItemRatingComment

Name	Data Type	Description	Writable	Is Mandatory (Create/Updates)	Max Length
CommentId	int	Represents the ID of the Comment. This is auto-generated value	No	No-Create Yes-Update	
RatingId	Int	Since the Comments are created for the ItemRatings, RatingID will hold the value of the parent ItemRating/Review	Yes	Yes	
Comments	String	Comments	Yes	Yes	No Limit
OwnerUserName	String	The Owner User who creates/updates the	Yes	Either OwnerUserName or	50

		ItemRating. The OwnerUserName will be UserName in the User Object. If this value is specified then it refers the User exists in RateItAll.com Users list		OwnerUserIdentifierId Should exist	
OwnerUserIdentifierId	String	Incase if you don't want to deal with the unique User Name or Creating Users through UserService, then this value should be set. This is a unique value for each application. This value can be Users Email Address or IP-Address etc.	Yes	Either OwnerUserName or OwnerUserIdentifierId Should exist	450

Sample Request and Response (In XML)

[Version1]

The below section will explain all the Sample Request and Response XML formats for each service.

Service: UserService

The below REST URL fetch requests

Get User by UserID - <http://api.rateitall.com/v1/SOAP/XML/Users/UserId/{UserId}?SecretKey={SecretKey}>

Get User by UserName-

<http://api.rateitall.com/v1/SOAP/XML/Users/UserName/{UserName}?SecretKey={SecretKey}>

Get User by Email Address -

<http://api.rateitall.com/v1/SOAP/XML/Users/EmailAddress/{EmailAddress}?SecretKey={SecretKey}>

will have the response XML in the below format

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <User>
      <UserId>26266713</UserId>
      <UserName>usera</UserName>
      <IsDisabled>>false</IsDisabled>
      <ImageUrl>http://ria-user.s3.amazonaws.com/11b25abc-0248-405d-bea3-5decb5e29f1b-x80.jpg</ImageUrl>
      <EmailAddress i:nil="true" />
      <NewPassword i:nil="true" />
      <CurrentPassword i:nil="true" />
    </User>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```

Note: The details of all elements inside the XML element <User> have been discussed in Error! Bookmark not defined..

The REST URL Fetch request

Get all Users in Application Id -

<http://api.rateitall.com/v1/SOAP/XML/Users/ApplicationId/{ApplicationId}?SecretKey={SecretKey}>

will have the response XML in the below format. There will be multiple <User> elements inside the <Value> element.




```

<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <User>
      <UserId>26266713</UserId>
      <UserName>usera</UserName>
      <IsDisabled>>false</IsDisabled>
      <ImageUrl>http://ria-user.s3.amazonaws.com/11b25abc-0248-405d-bea3-5decb5e29f1b-
        x80.jpg</ImageUrl>
      <EmailAddress i:nil="true" />
      <NewPassword i:nil="true" />
      <CurrentPassword i:nil="true" />
    </User>
    <User>
      <UserId>26266714</UserId>
      <UserName>userb</UserName>
      <IsDisabled>>false</IsDisabled>
      <ImageUrl>http://ria-user.s3.amazonaws.com/11b25abc-2322-405d-bea3-5decb5e29f1b-
        x80.jpg</ImageUrl>
      <EmailAddress i:nil="true" />
      <NewPassword i:nil="true" />
      <CurrentPassword i:nil="true" />
    </User>
    <User>.....
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>

```

The REST POST Request

Create a New User - <http://api.rateitall.com/v1/SOAP/XML/Users/Create?SecretKey={SecretKey}>

will create a new user with the User Information specified in the XML POST request.

Below is the sample XML Request for Creating a new User

Note: The details of all required and optional elements inside the XML element <User> have been discussed in Error! Bookmark not defined..

```

<User xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <UserName>userb</UserName>
  <EmailAddress>userb@rateitall.com</EmailAddress>
  <IsDisabled>>false</IsDisabled>
  <NewPassword>kaliko$5</NewPassword>
  <ImageUrl>http://example.com/sample.jpg</ImageUrl>
</User>

```



and below is the Sample XML response for creating the New User

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <User>
      <UserId>29917683</UserId>
      <UserName>userb</UserName>
      <IsDisabled>>false</IsDisabled>
      <ImageUrl>http://ria-user.s3.amazonaws.com/11b25abc-2322-405d-bea3-5dec5e29f1b-
x80.jpg </ImageUrl>
      <EmailAddress i:nil="true" />
      <NewPassword i:nil="true" />
      <CurrentPassword i:nil="true" />
    </User>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```

- Note:
1. All the ImageUrls specified on User creation and fetched and stored in our repository. The Response will have the url of the Image stored in our repository
 2. EmailAddress and Password will not be available in the Response XML's
 3. This Service can create only one user at a time

The REST POST Request

Update an Existing User - <http://api.rateitall.com/v1/SOAP/XML/Users/Update?SecretKey={SecretKey}>

will update the existing user information. Below is the sample XML request for updating an existing user

```
<User xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <UserId>29917683</UserId>
  <UserName>userbx</UserName>
  <EmailAddress>userb@rateitall.com</EmailAddress>
  <IsDisabled>>false</IsDisabled>
  <NewPassword i:nil="true" />
  <CurrentPassword>kaliko$5</CurrentPassword>
  <ImageUrl>http://example.com/sample2.jpg</ImageUrl>
</User>
```



and below is the sample XML response for the User updates

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <User>
      <UserId>29917683</UserId>
      <UserName>userbx</UserName>
      <IsDisabled>>false</IsDisabled>
      <ImageUrl>http://ria-user.s3.amazonaws.com/11b25abc-2322-405d-bea3-5decb5e29f1b-x80.jpg </ImageUrl>
      <EmailAddress i:nil="true" />
      <NewPassword i:nil="true" />
      <CurrentPassword i:nil="true" />
    </User>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```

Note: 1. On User Updates the Current Password should be specified

2. The Updates can be performed on UserName, EmailAddress, IsDisabled, NewPassword and ImageUrl fields

3. This Service can update only one user at a time.

The REST POST Request

Search Users - <http://api.rateitall.com/v1/SOAP/XML/Users/Search?SecretKey={SecretKey}>

will search users based on the fields specified on the XML request

Below is the sample XML request for searching users. The search criteria is specified in the form of XML, with atleast one or more elements in the Request XML can be specified.

```
<User xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <UserId>29917683</UserId>
  <UserName i:nil="true" />
  <EmailAddress i:nil="true" />
  <IsDisabled i:nil="true" />
  <NewPassword i:nil="true" />
  <CurrentPassword i:nil="true" />
  <ImageUrl i:nil="true" />
</User>
```



and below is the sample XML response for the User search. There will be multiple <User> elements inside the <Value> element depending on the Search request.

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <User>
      <UserId>29917683</UserId>
      <UserName>userb</UserName>
      <IsDisabled>>false</IsDisabled>
      <ImageUrl>http://ria-user.s3.amazonaws.com/11b25abc-2322-405d-bea3-5decb5e29f1b-
x80.jpg </ImageUrl>
      <EmailAddress i:nil="true" />
      <NewPassword i:nil="true" />
      <CurrentPassword i:nil="true" />
    </User>
    <User>.....
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```

Service: Subject Service

Subject Service now has only one fetch operation, which will list down all the subjects in RateItAll

Fetch All Subjects - <http://api.rateitall.com/v1/SOAP/XML/Subjects?SecretKey={SecretKey}>

Below is the Response XML format. Response will have multiple <Subject> elements

Note: The details of all elements inside the XML element <Subject> have been discussed in Error! Bookmark not defined..

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Subject>
      <SubjectId>1</SubjectId>
      <Name>Baseball</Name>
    </Subject>
    <Subject>
      <SubjectId>2</SubjectId>
      <Name>BaskeyBall</Name>
    </Subject>
    <Subject>.....
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```



Service: Topic Service

The below REST URL fetch requests

Get Topic by TopicId - <http://api.rateitall.com/v1/SOAP/XML/Topics/TopicId/{TopicId}?SecretKey={SecretKey}>

Get Topic by TopicIdentifierId and Application Id -

<http://api.rateitall.com/v1/SOAP/XML/Topics/TopicIdentifierId/{TopicIdentifierId}/ApplicationId/{ApplicationId}?SecretKey={SecretKey}>

will have the response in the below format

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Topic>
      <TopicId>1182700</TopicId>
      <TopicIdentifierId i:nil="true" />
      <SubjectId>1</SubjectId>
      <Name>Games Dime</Name>
      <Description>Sample Games</Description>
      <ImageUrl />
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>usera</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags i:nil="true" />
      <MetaDescription i:nil="true" />
      <MetaKeywords i:nil="true" />
      <TopicUrl>http://www.rateitall.com/t-1182700-games-dime.aspx</TopicUrl>
    </Topic>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```

Note: The details of all elements inside the XML element <Topic> have been discussed in Error! Bookmark not defined..

The below REST URL Fetch requests

Get Topics with Name (Contains) -

<http://api.rateitall.com/v1/SOAP/XML/Topics/Name/{Name}?SecretKey={SecretKey}>

Get Topics with Name (contains) and SubjectId-

<http://api.rateitall.com/v1/SOAP/XML/Topics/Name/{Name}/SubjectId/{SubjectId}?SecretKey={SecretKey}>

Get Topics in ApplicationId -

<http://api.rateitall.com/v1/SOAP/XML/Topics/ApplicationId/{ApplicationId}?SecretKey={SecretKey}>

Get Topics created by UserId -

<http://api.rateitall.com/v1/SOAP/XML/Topics/UserId/{UserId}?SecretKey={SecretKey}>

Get Topics created by User with UserIdentifierId -

<http://api.rateitall.com/v1/SOAP/XML/Topics/UserIdentifierId/{UserIdentifierId}?SecretKey={SecretKey}>

Search Topics with Search Query in the Url -

<http://api.rateitall.com/v1/SOAP/XML/Topics/Search/{Query}?SecretKey={SecretKey}>



will have the response in the below XML format. The Response will have multiple <Topic> elements

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Topic>
      <TopicId>1182700</TopicId>
      <TopicIdentifierId i:nil="true" />
      <SubjectId>1</SubjectId>
      <Name>Games Dime</Name>
      <Description>Sample Games</Description>
      <ImageUrl />
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>usera</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags i:nil="true" />
      <MetaDescription i:nil="true" />
      <MetaKeywords i:nil="true" />
      <TopicUrl>http://www.rateitall.com/t-1182700-games-dime.aspx</TopicUrl>
    </Topic>
    <Topic>
      <TopicId>434534</TopicId>
      <TopicIdentifierId i:nil="true" />
      <SubjectId>1</SubjectId>
      <Name>Hack Dime</Name>
      <Description>Hacker Games</Description>
      <ImageUrl />
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>userb</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags i:nil="true" />
      <MetaDescription i:nil="true" />
      <MetaKeywords i:nil="true" />
      <TopicUrl>http://www.rateitall.com/t-434534-hack-dime.aspx</TopicUrl>
    </Topic>
    <Topic>.....
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```

The REST POST Request

Create a new Topic - <http://api.rateitall.com/v1/SOAP/XML/Topics/Create?SecretKey={SecretKey}>

will create a new Topic with the Topic Information specified in the XML POST request.

Below is the sample XML Request for Creating a new Topic.

Note: The details of all required and optional elements inside the XML element <Topic> have been discussed in Error! Bookmark not defined..



```

<Topic xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <TopicIdentifierId i:nil="true" />
  <SubjectId>1</SubjectId>
  <Name>Powerful Games</Name>
  <Description>List of Powerful games</Description>
  <ImageUrl>http://www.gamer.com/sample4.jpg</ImageUrl>
  <IsDisabled>>false</IsDisabled>
  <OwnerUserName>usera</OwnerUserName>
  <OwnerUserIdentifierId i:nil="true" />
  <TitleTags>Games,Gamer,Powerful </TitleTags>
  <MetaDescription>List of Power ful games</MetaDescription>
  <MetaKeywords> Games,Gamer,Powerful </MetaKeywords>
</Topic>

```

and below is the Sample XML response for creating the New Topic

```

<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Topic>
      <TopicId>434534</TopicId>
      <TopicIdentifierId i:nil="true" />
      <SubjectId>1</SubjectId>
      <Name>Powerful Games</Name>
      <Description>List of Powerful games</Description>
      <ImageUrl> http://ria-item.s3.amazonaws.com/11b25abc-2322-405d-bea3-5decb5e29f1b-x80.jpg </ImageUrl>
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>usera</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags>Games,Gamer,Powerful </TitleTags>
      <MetaDescription>List of Power ful games</MetaDescription>
      <MetaKeywords> Games,Gamer,Powerful </MetaKeywords>
      <TopicUrl>http://www.rateitall.com/t-434534-hack-dime.aspx</TopicUrl>
    </Topic>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>

```

- Note: 1. All the ImageUrls specified on Topic creation and fetched and stored in our repository. The Response will have the url of the Image stored in our repository
2. This Service can create only one Topic at a time

The REST POST Request



Update an Existing Topic - <http://api.rateitall.com/v1/SOAP/XML/Topics/Update?SecretKey={SecretKey}>

will update the existing Topic information. Below is the sample XML request for updating an existing Topic

```
<Topic xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <TopicId>434534</TopicId>
  <SubjectId>1</SubjectId>
  <Name>Powerful Games for PC/MAC</Name>
  <Description>List of Powerful games</Description>
  <IsDisabled>>false</IsDisabled>
  <OwnerUserName>userb</OwnerUserName>
</Topic>
```

and below is the sample XML response for the Topic updates

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Topic>
      <TopicId>434534</TopicId>
      <TopicIdentifierId i:nil="true" />
      <SubjectId>1</SubjectId>
      <Name> Powerful Games for PC/MAC </Name>
      <Description>List of Powerful games</Description>
      <ImageUrl> http://ria-item.s3.amazonaws.com/11b25abc-2322-405d-bea3-5decb5e29f1b-x80.jpg </ImageUrl>
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>userb</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags>Games,Gamer,Powerful </TitleTags>
      <MetaDescription>List of Power ful games</MetaDescription>
      <MetaKeywords> Games,Gamer,Powerful </MetaKeywords>
      <TopicUrl>http://www.rateitall.com/t-434534-hack-dime.aspx</TopicUrl>
    </Topic>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```

- Note: 1. On Topic Updates either TopicId or TopicIdentifierId should be specified
2. This Service can update only one Topic at a time.

The REST POST Request

Search Topics - <http://api.rateitall.com/v1/SOAP/XML/Topics/Search?SecretKey={SecretKey}>

will search Topics based on the fields specified on the XML request



Below is the sample XML request for searching Topics. The search criteria are specified in the form of XML, with at least one or more elements in the Request XML can be specified.

```
<Topic xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <TopicId>434534</TopicId>
  <SubjectId i:nil="true" />
  <Name>Games</Name>
  <Description i:nil="true" />
  <IsDisabled i:nil="true" />
  <OwnerUserName i:nil="true" />
</Topic>
```

and below is the sample XML response for the Topic search. There will be multiple <Topic> elements inside the <Value> element.

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Topic>
      <TopicId>434534</TopicId>
      <TopicIdentifierId i:nil="true" />
      <SubjectId>1</SubjectId>
      <Name> Powerful Games for PC/MAC </Name>
      <Description>List of Powerful games</Description>
      <ImageUrl> http://ria-item.s3.amazonaws.com/11b25abc-2322-405d-bea3-5decb5e29f1b-x80.jpg </ImageUrl>
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>userb</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags>Games,Gamer,Powerful </TitleTags>
      <MetaDescription>List of Power ful games</MetaDescription>
      <MetaKeywords> Games,Gamer,Powerful </MetaKeywords>
      <TopicUrl>http://www.rateitall.com/t-434534-hack-dime.aspx</TopicUrl>
    </Topic>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```

Note: 1. The Description Element in the Search XML will not be considered.

2. You can specify only TopicId, SubjectId, Name, Image, OwnerUserId and ApplicationId in the Topic Search XML.

Service: Item Service

The below REST URL fetch requests

Get Item by ItemId - <http://api.rateitall.com/v1/SOAP/XML/Items/ItemId/{ItemId}?SecretKey={SecretKey}>



Get Item by ItemIdentifierId and Application Id -

<http://api.rateitall.com/v1/SOAP/XML/Items/ItemIdentifierId/{ItemIdentifierId}/ApplicationId/{ApplicationId}?SecretKey={SecretKey}>

will have the response in the below format

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Item>
      <ItemId>2138143</ItemId>
      <ItemIdentifierId i:nil="true" />
      <TopicId>1182706</TopicId>
      <Name>Resistance</Name>
      <Description>War Game - Resistance</Description>
      <ImageUrl />
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>usera</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags i:nil="true" />
      <MetaDescription i:nil="true" />
      <MetaKeywords i:nil="true" />
      <AverageRating>0 based on 0 ratings</AverageRating>
      <ItemLinks i:nil="true" />
      <ItemUrl>http://www.rateitall.com/i-2138143- Resistance.aspx</ItemUrl>
    </Item>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```

Note: The details of all elements inside the XML element <Item> have been discussed in Error! Bookmark not defined..

The below REST URL Fetch requests

Get Items with Name (Contains) -

<http://api.rateitall.com/v1/SOAP/XML/Items/Name/{Name}?SecretKey={SecretKey}>

Get Items in Topic with TopicId -

<http://api.rateitall.com/v1/SOAP/XML/Items/TopicId/{TopicId}?SecretKey={SecretKey}>

Get Items in ApplicationId -

<http://api.rateitall.com/v1/SOAP/XML/Items/ApplicationId/{ApplicationId}?SecretKey={SecretKey}>

Get Items created by UserId -

<http://api.rateitall.com/v1/SOAP/XML/Items/UserId/{UserId}?SecretKey={SecretKey}>

Get Items created by User with UserIdentifierId -

<http://api.rateitall.com/v1/SOAP/XML/Items/UserIdentifierId/{UserIdentifierId}?SecretKey={SecretKey}>

Search Items with Search Query in the Url -

<http://api.rateitall.com/v1/SOAP/XML/Items/Search/{Query}?SecretKey={SecretKey}>

will have the response in the below XML format. The Response will have multiple <Items> elements



```

<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Item>
      <ItemId>2138143</ItemId>
      <ItemIdentifierId i:nil="true" />
      <TopicId>1182706</TopicId>
      <Name>Resistance</Name>
      <Description>War Game - Resistance</Description>
      <ImageUrl />
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>usera</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags i:nil="true" />
      <MetaDescription i:nil="true" />
      <MetaKeywords i:nil="true" />
      <AverageRating>0 based on 0 ratings</AverageRating>
      <ItemLinks i:nil="true" />
      <ItemUrl>http://www.rateitall.com/i-2138143- Resistance.aspx</ItemUrl>
    </Item>
    <Item>
      <ItemId>2138145</ItemId>
      <ItemIdentifierId i:nil="true" />
      <TopicId>1182706</TopicId>
      <Name>Resistance II</Name>
      <Description>War Game - Resistance II</Description>
      <ImageUrl />
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>usera</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags i:nil="true" />
      <MetaDescription i:nil="true" />
      <MetaKeywords i:nil="true" />
      <AverageRating>0 based on 0 ratings</AverageRating>
      <ItemLinks i:nil="true" />
      <ItemUrl>http://www.rateitall.com/i-2138143- Resistance.aspx</ItemUrl>
    </Item>
    <Item>.....
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>

```

The REST POST Request

Create a new Item - <http://api.rateitall.com/v1/SOAP/XML/Items/Create?SecretKey={SecretKey}>

will create a new Item with the Item Information specified in the XML POST request.

Below is the sample XML Request for Creating a new Item.

Note: The details of all required and optional elements inside the XML element <Item> have been discussed in Error! Bookmark not defined..



```

<Item xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ItemIdentifierId>WII123</ItemIdentifierId>
  <TopicId>1182706</TopicId>
  <Name>WII Tennis</Name>
  <Description>Games On WII</Description>
  <ImageUrl>http://www.sample.com/image.jpg</ImageUrl>
  <IsDisabled>>false</IsDisabled>
  <OwnerUserName>userc</OwnerUserName>
  <OwnerUserIdentifierId i:nil="true" />
  <TitleTags>Games,WII </TitleTags>
  <MetaDescription> Play games on WII </MetaDescription>
  <MetaKeywords>Games,WII,Play </MetaKeywords>
  <ItemLinks>
    <ItemLink>
      <Url>http://www.wii.com</Url>
      <Name>WII Website</Name>
    </ItemLink>
  </ItemLinks>
</Item>

```

and below is the Sample XML response for creating the New Item

```

<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Item>
      <ItemId>112345</ItemId>
      <ItemIdentifierId>WII123</ItemIdentifierId>
      <TopicId>1182706</TopicId>
      <Name>WII Tennis</Name>
      <Description>Games On WII</Description>
      <ImageUrl>http://www.sample.com/image.jpg</ImageUrl>
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>userc</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags>Games,WII </TitleTags>
      <MetaDescription> Play games on WII </MetaDescription>
      <MetaKeywords>Games,WII,Play </MetaKeywords>
      <AverageRating>0 based on 0 ratings</AverageRating>
      <ItemUrl>http://www.rateitall.com/i-112345-WII-Tennis.aspx</ItemUrl>
      <ItemLinks>
        <ItemLink>
          <Url>http://www.wii.com</Url>
          <Name>WII Website</Name>
        </ItemLink>
      </ItemLinks>
    </Item>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>

```

Note: 1. All the ImageUrls specified on Item creation and fetched and stored in our repository. The Response will have the url of the Image stored in our repository

2. This Service can create only one Item at a time



3. Multiple <ItemLinks> XML elements can exist on the Item element on Creation

The REST POST Request

Update an Existing Item - <http://api.rateitall.com/v1/SOAP/XML/Items/Update?SecretKey={SecretKey}>

will update the existing Item information. Below is the sample XML request for updating an existing Item

```
<Item xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ItemId>112345</ItemId>
  <ItemIdentifierId>WII123</ItemIdentifierId>
  <TopicId>1182706</TopicId>
  <Name>WII Tennis</Name>
  <Description>Crack WII with WII Tennis</Description>
</Item>
```

and below is the sample XML response for the Item updates

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Item>
      <ItemId>112345</ItemId>
      <ItemIdentifierId>WII123</ItemIdentifierId>
      <TopicId>1182706</TopicId>
      <Name>WII Tennis</Name>
      <Description> Crack WII with WII Tennis </Description>
      <ImageUrl>http://www.sample.com/image.jpg</ImageUrl>
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>userc</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags>Games,WII </TitleTags>
      <MetaDescription> Play games on WII </MetaDescription>
      <MetaKeywords>Games,WII,Play </MetaKeywords>
      <AverageRating>0 based on 0 ratings</AverageRating>
      <ItemUrl>http://www.rateitall.com/i-112345-WII-Tennis.aspx</ItemUrl>
      <ItemLinks>
        <ItemLink>
          <Url>http://www.wii.com</Url>
          <Name>WII Website</Name>
        </ItemLink>
      </ItemLinks>
    </Item>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```

Note: 1. On Item Updates either ItemId or ItemIdentifierId should be specified

2. This Service can update only one Item at a time.



The REST POST Request

Search Items - <http://api.rateitall.com/v1/SOAP/XML/Items/Search?SecretKey={SecretKey}>

will search Items based on the fields specified on the XML request

Below is the sample XML request for searching Items. The search criteria are specified in the form of XML, with at least one or more elements in the Request XML can be specified.

```
<Item xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ItemIdentifierId>WII123</ItemIdentifierId>
  <TopicId>1182706</TopicId>
  <Name>WII Tennis</Name>
  <Description>Crack WII with WII Tennis</Description>
</Item>
```

and below is the sample XML response for the Item search. There will be multiple <Item> elements inside the <Value> element.

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Item>
      <ItemId>112345</ItemId>
      <ItemIdentifierId>WII123</ItemIdentifierId>
      <TopicId>1182706</TopicId>
      <Name>WII Tennis</Name>
      <Description> Crack WII with WII Tennis </Description>
      <ImageUrl>http://www.sample.com/image.jpg</ImageUrl>
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>userc</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags>Games,WII </TitleTags>
      <MetaDescription> Play games on WII </MetaDescription>
      <MetaKeywords>Games,WII,Play </MetaKeywords>
      <AverageRating>0 based on 0 ratings</AverageRating>
      <ItemUrl>http://www.rateitall.com/i-112345-WII-Tennis.aspx</ItemUrl>
      <ItemLinks>
        <ItemLink>
          <Url>http://www.wii.com</Url>
          <Name>WII Website</Name>
        </ItemLink>
      </ItemLinks>
    </Item>
    <Item>.....
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>
```

- Note: 1. The Description Element in the Search XML will not be considered.
 2. You can specify only ItemId, ItemIdentifierId, TopicId, Name, Image, OwnerUserName, OwnerUserIdentifierId and ApplicationId in the Topic Search XML.

The below REST URL fetch requests

Get Business by ItemId- <http://api.rateitall.com/v1/SOAP/XML/Business/ItemId/{ItemId}?SecretKey={SecretKey}>

will have the response in the below format.

```
<BusinessResponse xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Item>
      <ItemId>121212</ItemId>
      <ItemIdentifierId i:nil="true" />
      <TopicId>1182706</TopicId>
      <Name>Pizzeria</Name>
      <Description>Best Pizza In San francisco</Description>
      <ImageUrl />
      <IsDisabled>>false</IsDisabled>
      <OwnerUserName>usera</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <TitleTags i:nil="true" />
      <MetaDescription i:nil="true" />
      <MetaKeywords i:nil="true" />
      <AverageRating>5 based on 5 ratings</AverageRating>
      <ItemLinks i:nil="true" />
      <ItemUrl>http://www.rateitall.com/i-121212-pizzeria.aspx</ItemUrl>
      <Type />
      <Address1>100 mission street</Address1>
      <Address2 />
      <Suite />
      <City>San francisco</City>
      <State>CA</State>
      <ZipCode />
      <Country>USA</Country>
      <Phone />
      <Fax />
      <Email />
      <Latitude i:nil="true" />
      <Longitude i:nil="true" />
    </Item>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</BusinessResponse>
```

Note: The details of all elements inside the XML element <Item> have been discussed in Error! Bookmark not defined..



2. The Business Object is just an extension for Item XML with additional fields like "Suite, City, State, ZipCode, Country, Phone, Fax, Email, Address1 and Address2"

The REST POST Request

Create a new Business - <http://api.rateitall.com/v1/SOAP/XML/Business/Create?SecretKey={SecretKey}>

Creating a new Business will be same as creating a New Item, with the Additional Address Fields. Please refer to the "Create a New Item" section above.

The Business Creation has the Auto Topic Lookup functionality, so for example if you want to add a Business of Type 'dentist' in 'San Francisco, CA', API will make an automatic lookup for the relevant Topic. So you don't need to create a new Topic for this Item. The Topic Id doesn't need to be set if the Business Type exists.

Below is the XML sample for creating new Business, (here we used the Business Type instead of the Topic If)

```
<Item xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Name>Pizzeria</Name>
  <Description>Best Pizza In San francisco</Description>
  <ImageUrl />
  <IsDisabled>>false</IsDisabled>
  <OwnerUserName/>
  <OwnerUserIdentifierId>pizzaowner</OwnerUserIdentifierId>
  <TitleTags i:nil="true" />
  <MetaDescription i:nil="true" />
  <MetaKeywords i:nil="true" />
  <AverageRating>5 based on 5 ratings</AverageRating>
  <ItemLinks i:nil="true" />
  <ItemUrl>http://www.rateitall.com/i-121212-pizzeria.aspx</ItemUrl>
  <Type>Pizza</Type>
  <Address1>100 mission street</Address1>
  <Address2 />
  <Suite />
  <City>San francisco</City>
  <State>CA</State>
  <ZipCode />
  <Country>USA</Country>
  <Phone />
  <Fax />
  <Email />
  <Latitude i:nil="true" />
  <Longitude i:nil="true" />
</Item>
```


The REST POST Request

Update an Existing Business -

<http://api.rateitall.com/v1/SOAP/XML/Business/Update?SecretKey={SecretKey}>

Its same as updating an Existing Item, please refer to the “Update an existing Item” section for more information. All the Address fields can be updated above.

The Auto Topic Lookup functionality works for Business Updates as well. Please specify the Business Type instead of the Topic Id.

The REST POST Request

Search Business - <http://api.rateitall.com/v1/SOAP/XML/Business/Search?SecretKey={SecretKey}>

It’s same as Searching an item with Search Parameters specified in the XML. Please refer to “Search Items” Section for More Information. The Address fields City, State, Country and Zipcode can be included in the Search business query.

Service: ItemRating Service

The below REST URL fetch request

Get by RatingId - <http://api.rateitall.com/v1/SOAP/XML/ItemRatings/RatingId/{RatingId}?SecretKey={SecretKey}>

will have the response in the below sample format



```

<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <ItemRating>
      <RatingId>12367</RatingId>
      <ItemId>18319</ItemId>
      <ItemIdentifierId i:nil="true" />
      <Item>
        <ItemId>18319</ItemId>
        <ItemIdentifierId i:nil="true" />
        <TopicId>495</TopicId>
        <Name>Fahrenheit 451 (Ray Bradbury)</Name>
        <Description />
        <ImageUrl />
        <IsDisabled>>false</IsDisabled>
        <OwnerUserName i:nil="true" />
        <OwnerUserIdentifierId i:nil="true" />
        <TitleTags>fahrenheit-451-ray-bradbury</TitleTags>
        <MetaDescription i:nil="true" />
        <MetaKeywords i:nil="true" />
        <AverageRating>3.76 based on 84 ratings</AverageRating>
        <ItemLinks>
          <ItemLink>
            <Url>http://www.amazon.com/exec/obidos/redirect-home/rateitallcom</Url>
            <Name>Amazon.com</Name>
          </ItemLink>
          <ItemLink>
            <Url>http://rateitall.pricegrabber.com/home_books.php?ut=</Url>
            <Name>Find the lowest prices on books!</Name>
          </ItemLink>
        </ItemLinks>
        <ItemUrl>http://www.rateitall.com/i-18319-fahrenheit-451-ray-bradbury.aspx</ItemUrl>
      </Item>
      <Rating>3</Rating>
      <Comments i:nil="true" />
      <OwnerUserName>ben@2608om</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <IsDisabled>>false</IsDisabled>
      <FunnyVotes>0</FunnyVotes>
      <AgreeVotes>0</AgreeVotes>
      <HelpfulVotes>0</HelpfulVotes>
      <DisagreeVotes>0</DisagreeVotes>
      <ItemRatingUrl>http://www.rateitall.com/r-12367.aspx</ItemRatingUrl>
    </ItemRating>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>

```

Note: The details of all elements inside the XML element <ItemRating> have been discussed in Error! Bookmark not defined.. The <ItemRating> will always include the parent <Item> information in the XML.

The below REST URL Fetch requests

Get all ItemRatings by ItemId -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/ItemId/{ItemId}?SecretKey={SecretKey}>



Get all ItemRatings by ItemId with Comments -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/ItemId/{ItemId}/Comments?SecretKey={SecretKey}>

Get all ItemRatings by ItemIdentifierId and ApplicationId -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/ItemIdentifierId/{ItemIdentifierId}/ApplicationId/{ApplicationId}?SecretKey={SecretKey}>

Get all ItemRatings with Comments contains -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/Comments/{Comments}?SecretKey={SecretKey}>

Get all ItemRatings in TopicId -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/TopicId/{TopicId}?SecretKey={SecretKey}>

Get all ItemRatings in ApplicationId -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/ApplicationId/{ApplicationId}?SecretKey={SecretKey}>

Get all ItemRatings in ApplicationId and UserId -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/ApplicationId/{ApplicationId}/UserId/{UserId}?SecretKey={SecretKey}>

Get all ItemRatings in UserIdentifierId and ApplicationId -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/UserIdentifierId/{UserIdentifierId}/ApplicationId/{ApplicationId}?SecretKey={SecretKey}>

Get all ItemRatings in ApplicationId, ItemId and UserId-

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/ApplicationId/{ApplicationId}/ItemId/{ItemId}/UserId/{UserId}?SecretKey={SecretKey}>

Get all ItemRatings in ApplicationId, ItemIdentifierId and UserId-

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/ApplicationId/{ApplicationId}/ItemIdentifierId/{ItemIdentifierId}/UserId/{UserId}?SecretKey={SecretKey}>

Get all ItemRatings in ApplicationId, TopicId and UserId-

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/ApplicationId/{ApplicationId}/TopicId/{TopicId}/UserId/{UserId}?SecretKey={SecretKey}>

Get all ItemRatings in ApplicationId, TopicId, ItemIdentifierId and UserId-

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/ApplicationId/{ApplicationId}/TopicId/{TopicId}/ItemIdentifierId/{ItemIdentifierId}/UserId/{UserId}?SecretKey={SecretKey}>

will have the response in the below XML format. The Response will have multiple <ItemRating> elements



```

<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <ItemRating>
      <RatingId>12367</RatingId>
      <ItemId>18319</ItemId>
      <ItemIdentifierId i:nil="true" />
      <Item>
        <ItemId>18319</ItemId>
        <ItemIdentifierId i:nil="true" />
        <TopicId>495</TopicId>
        <Name>Fahrenheit 451 (Ray Bradbury)</Name>
        <Description />
        <ImageUrl />
        <IsDisabled>>false</IsDisabled>
        <OwnerUserName i:nil="true" />
        <OwnerUserIdentifierId i:nil="true" />
        <TitleTags>fahrenheit-451-ray-bradbury</TitleTags>
        <MetaDescription i:nil="true" />
        <MetaKeywords i:nil="true" />
        <AverageRating>3.76 based on 84 ratings</AverageRating>
        <ItemLinks>
          <ItemLink>
            <Url>http://www.amazon.com/exec/obidos/redirect-home/rateitallcom</Url>
            <Name>Amazon.com</Name>
          </ItemLink>
          <ItemLink>
            <Url>http://rateitall.pricegrabber.com/home_books.php?ut=</Url>
            <Name>Find the lowest prices on books!</Name>
          </ItemLink>
        </ItemLinks>
        <ItemUrl>http://www.rateitall.com/i-18319-fahrenheit-451-ray-bradbury.aspx</ItemUrl>
      </Item>
      <Rating>3</Rating>
      <Comments i:nil="true" />
      <OwnerUserName>ben@2608om</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <IsDisabled>>false</IsDisabled>
      <FunnyVotes>0</FunnyVotes>
      <AgreeVotes>0</AgreeVotes>
      <HelpfulVotes>0</HelpfulVotes>
      <DisagreeVotes>0</DisagreeVotes>
      <ItemRatingUrl>http://www.rateitall.com/r-12367.aspx</ItemRatingUrl>
    </ItemRating>
    <ItemRating>.....
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</Response>

```

The REST POST Request

Create a new Item Rating -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/Create?SecretKey={SecretKey}>

will create a new ItemRating with the ItemRating Information specified in the XML POST request.

Below is the sample XML Request for Creating a new ItemRating.



Note: The details of all required and optional elements inside the XML element <ItemRating> have been discussed in Error! Bookmark not defined..

```
<ItemRating xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ItemId>18319</ItemId>
  <Rating>4</Rating>
  <Comments>I really like this one</Comments>
  <OwnerUserName>userc</OwnerUserName>
  <IsDisabled>>false</IsDisabled>
</ItemRating>
```

and below is the Sample XML response for creating the New ItemRating

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <ItemRating>
      <RatingId>12368</RatingId>
      <ItemId>18319</ItemId>
      <ItemIdentifierId i:nil="true" />
      <Item>
        <ItemId>18319</ItemId>
        <ItemIdentifierId i:nil="true" />
        <TopicId>495</TopicId>
        <Name>Fahrenheit 451 (Ray Bradbury)</Name>
        <Description />
        <ImageUrl />
        <IsDisabled>>false</IsDisabled>
        <OwnerUserName i:nil="true" />
        <OwnerUserIdentifierId i:nil="true" />
        <TitleTags>fahrenheit-451-ray-bradbury</TitleTags>
        <MetaDescription i:nil="true" />
        <MetaKeywords i:nil="true" />
        <AverageRating>3.76 based on 84 ratings</AverageRating>
        <ItemLinks>
          <ItemLink>
            <Url>http://www.amazon.com/exec/obidos/redirect-home/rateitallcom</Url>
            <Name>Amazon.com</Name>
          </ItemLink>
          <ItemLink>
            <Url>http://rateitall.pricegrabber.com/home_books.php?ut=</Url>
            <Name>Find the lowest prices on books!</Name>
          </ItemLink>
        </ItemLinks>
        <ItemUrl>http://www.rateitall.com/i-18319-fahrenheit-451-ray-bradbury.aspx</ItemUrl>
      </Item>
      <Rating>4</Rating>
      <Comments>I really like this one</Comments>
      <OwnerUserName>ben@2608om</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <IsDisabled>>false</IsDisabled>
      <FunnyVotes>0</FunnyVotes>
      <AgreeVotes>0</AgreeVotes>
      <HelpfulVotes>0</HelpfulVotes>
      <DisagreeVotes>0</DisagreeVotes>
      <ItemRatingUrl>http://www.rateitall.com/r-12367.aspx</ItemRatingUrl>
    </ItemRating>
  </Value>
```



```
<Message i:nil="true" />
<IsSuccess>true</IsSuccess>
</Response>
```

Note: 1. This Service can create only one Item Rating at a time.

The ItemRating can either specify an existing ItemId or Create a new Item on the Fly. Below is the Sample XML request which includes the New Item Information along with the ItemRating. The New Item will be created before the ItemRating.

Item Rating Create Request XML with New Item

```
<ItemRating xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Item>
    <TopicId>1182710</TopicId>
    <Name>PS3</Name>
    <Description>Sony PS3 Playstation</Description>
    <IsDisabled>>false</IsDisabled>
    <OwnerUserName>usera</OwnerUserName>
  </Item>
  <Rating>4</Rating>
  <Comments>Good one</Comments>
  <OwnerUserName>userb</OwnerUserName>
  <IsDisabled>>false</IsDisabled>
</ItemRating>
```

The REST POST Request

Update an Existing ItemRating -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/Update?SecretKey={SecretKey}>

will update the existing ItemRating information. Below is the sample XML request for updating an existing ItemRating

```
<ItemRating xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <RatingId>12368</RatingId>
  <Rating>5</Rating>
  <Comments>Great</Comments>
  <OwnerUserName>userc</OwnerUserName>
  <IsDisabled>>false</IsDisabled>
</ItemRating>
```

and below is the sample XML response for the ItemRating updates

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <ItemRating>
      <RatingId>12368</RatingId>
      <ItemId>18319</ItemId>
      <ItemIdentifierId i:nil="true" />
      <Item>
```



```

<ItemId>18319</ItemId>
<ItemIdentifierId i:nil="true" />
<TopicId>495</TopicId>
<Name>Fahrenheit 451 (Ray Bradbury)</Name>
<Description />
<ImageUrl />
<IsDisabled>>false</IsDisabled>
<OwnerUserName i:nil="true" />
<OwnerUserIdentifierId i:nil="true" />
<TitleTags>fahrenheit-451-ray-bradbury</TitleTags>
<MetaDescription i:nil="true" />
<MetaKeywords i:nil="true" />
<AverageRating>3.76 based on 84 ratings</AverageRating>
<ItemLinks>
  <ItemLink>
    <Url>http://www.amazon.com/exec/obidos/redirect-home/rateitallcom</Url>
    <Name>Amazon.com</Name>
  </ItemLink>
  <ItemLink>
    <Url>http://rateitall.pricegrabber.com/home_books.php?ut=</Url>
    <Name>Find the lowest prices on books!</Name>
  </ItemLink>
</ItemLinks>
<ItemUrl>http://www.rateitall.com/i-18319-fahrenheit-451-ray-bradbury.aspx</ItemUrl>
</Item>
<Rating>5</Rating>
<Comments>Great</Comments>
<OwnerUserName>ben@2608om</OwnerUserName>
<OwnerUserIdentifierId i:nil="true" />
<IsDisabled>>false</IsDisabled>
<FunnyVotes>0</FunnyVotes>
<AgreeVotes>0</AgreeVotes>
<HelpfulVotes>0</HelpfulVotes>
<DisagreeVotes>0</DisagreeVotes>
<ItemRatingUrl>http://www.rateitall.com/r-12367.aspx</ItemRatingUrl>
</ItemRating>
</Value>
<Message i:nil="true" />
<IsSuccess>>true</IsSuccess>
</Response>

```

- Note: 1. On ItemRating updates, RatingId should be specified
 2. This Service can update only one Item Rating at a time.

The REST POST Request

Search Items - <http://api.rateitall.com/v1/SOAP/XML/ItemRatings/Search?SecretKey={SecretKey}>

will search ItemRatings based on the fields specified on the XML request

Below is the sample XML request for searching ItemRatings. The search criteria are specified in the form of XML, with at least one or more elements in the Request XML can be specified.



```

<ItemRating xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <RatingId>12368</RatingId>
  <Rating>5</Rating>
  <Comments>Great</Comments>
  <OwnerUserName>userc</OwnerUserName>
  <IsDisabled>>false</IsDisabled>
</ItemRating>

```

and below is the sample XML response for the Item Ratings search. There may be multiple <ItemRatings> elements inside the <Value> element.

```

<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <ItemRating>
      <RatingId>12368</RatingId>
      <ItemId>18319</ItemId>
      <ItemIdentifierId i:nil="true" />
      <Item>
        <ItemId>18319</ItemId>
        <ItemIdentifierId i:nil="true" />
        <TopicId>495</TopicId>
        <Name>Fahrenheit 451 (Ray Bradbury)</Name>
        <Description />
        <ImageUrl />
        <IsDisabled>>false</IsDisabled>
        <OwnerUserName i:nil="true" />
        <OwnerUserIdentifierId i:nil="true" />
        <TitleTags>fahrenheit-451-ray-bradbury</TitleTags>
        <MetaDescription i:nil="true" />
        <MetaKeywords i:nil="true" />
        <AverageRating>3.76 based on 84 ratings</AverageRating>
        <ItemLinks>
          <ItemLink>
            <Url>http://www.amazon.com/exec/obidos/redirect-home/rateitallcom</Url>
            <Name>Amazon.com</Name>
          </ItemLink>
          <ItemLink>
            <Url>http://rateitall.pricegrabber.com/home_books.php?ut=</Url>
            <Name>Find the lowest prices on books!</Name>
          </ItemLink>
        </ItemLinks>
        <ItemUrl>http://www.rateitall.com/i-18319-fahrenheit-451-ray-bradbury.aspx</ItemUrl>
      </Item>
      <Rating>5</Rating>
      <Comments>Great</Comments>
      <OwnerUserName>ben@2608om</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
      <IsDisabled>>false</IsDisabled>
      <FunnyVotes>0</FunnyVotes>
      <AgreeVotes>0</AgreeVotes>
      <HelpfulVotes>0</HelpfulVotes>
      <DisagreeVotes>0</DisagreeVotes>
      <ItemRatingUrl>http://www.rateitall.com/r-12367.aspx</ItemRatingUrl>
    </ItemRating>
    <ItemRating>...
  </Value>
  <Message i:nil="true" />

```




```
<IsSuccess>true</IsSuccess>
</Response>
```

Note: 1. You can specify only RatingId, Rating, UserId, ApplicationId, ItemId, TopicId, SubjectId, ItemIdentifierId, FunnyVotes, AgreeVotes, HelpfulVotes, IsDisabled and Comments in the Search XML.

Service: ItemRating Service (Vote Functionality)

Each ItemRating can be Voted by User as 'Helpful', 'Funny', 'Agree' or 'Disagree'

To cast a new Vote, follow the below REST URL fetch request

Cast a Vote as "Helpful" -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/RatingId/{RatingId}/VotingUserId/{VotingUserId}/Vote/Helpful?SecretKey={SecretKey}>

will have the response in the below sample format

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <ItemRating>
      <RatingId>12367</RatingId>
      <ItemId>18319</ItemId>
      <ItemIdentifierId i:nil="true" />
      <Item>
        <ItemId>18319</ItemId>
        <ItemIdentifierId i:nil="true" />
        <TopicId>495</TopicId>
        <Name>Fahrenheit 451 (Ray Bradbury)</Name>
        <Description />
        <ImageUrl />
        <IsDisabled>>false</IsDisabled>
        <OwnerUserName i:nil="true" />
        <OwnerUserIdentifierId i:nil="true" />
        <TitleTags>fahrenheit-451-ray-bradbury</TitleTags>
        <MetaDescription i:nil="true" />
        <MetaKeywords i:nil="true" />
        <AverageRating>3.76 based on 84 ratings</AverageRating>
        <ItemLinks>
          <ItemLink>
            <Url>http://www.amazon.com/exec/obidos/redirect-home/rateitallcom</Url>
            <Name>Amazon.com</Name>
          </ItemLink>
          <ItemLink>
            <Url>http://rateitall.pricegrabber.com/home_books.php?ut=</Url>
            <Name>Find the lowest prices on books!</Name>
          </ItemLink>
        </ItemLinks>
        <ItemUrl>http://www.rateitall.com/i-18319-fahrenheit-451-ray-bradbury.aspx</ItemUrl>
      </Item>
    </ItemRating>
  </Value>
</Response>
```



```

    <Rating>3</Rating>
    <Comments i:nil="true" />
    <OwnerUserName>ben@2608om</OwnerUserName>
    <OwnerUserIdentifierId i:nil="true" />
    <IsDisabled>>false</IsDisabled>
    <FunnyVotes>0</FunnyVotes>
    <AgreeVotes>0</AgreeVotes>
    <HelpfulVotes>1</HelpfulVotes>
    <DisagreeVotes>0</DisagreeVotes>
    <ItemRatingUrl>http://www.rateitall.com/r-12367.aspx</ItemRatingUrl>
  </ItemRating>
</Value>
<Message i:nil="true" />
<IsSuccess>>true</IsSuccess>
</Response>

```

To cast a ItemRating as “Helpful” and “Agree”,

First

Cast a Vote as “Helpful” -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/RatingId/{RatingId}/VotingUserId/{VotingUserId}/Vote/Helpful?SecretKey={SecretKey}>

Then

Cast a Vote as “Agree” -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/RatingId/{RatingId}/VotingUserId/{VotingUserId}/Vote/Agree?SecretKey={SecretKey}>

the below REST URL fetch request

Get Vote information Voted by UserId -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatings/RatingId/{RatingId}/VotingUserId/{VotingUserId}?SecretKey={SecretKey}>

Will have the XML Response as follows

```

<VoteResponse xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <Helpful>>false</Helpful>
    <Funny>>true</Funny>
    <Agree>>true</Agree>
    <Disagree>>false</Disagree>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>>true</IsSuccess>
</VoteResponse>

```



Service: ItemRatingComment Service

The below REST URL fetch request

Get Comment on a ItemRating by CommentId -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatingComments/CommentId/{CommentId}?SecretKey={SecretKey}>

will have the response in the below sample format

```
<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <ItemRatingComment>
      <CommentId>1</CommentId>
      <RatingId>1462177</RatingId>
      <Comments>Wow, that is inspirational. I must see this place!</Comments>
      <OwnerUserName>CzechMate</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
    </ItemRatingComment>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>true</IsSuccess>
</Response>
```

Note: The details of all elements inside the XML element <ItemRatingComment> have been discussed in [Error! Bookmark not defined.](#)

The below REST URL Fetch requests

Get all Comments for an ItemRating by RatingId -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatingComments/RatingId/{RatingId}?SecretKey={SecretKey}>

Get all Comments created from an Application with ApplicationId -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatingComments/ApplicationId/{ApplicationId}?SecretKey={SecretKey}>

Get all Comments created from an Application with ApplicationId and by User with UserId-

<http://api.rateitall.com/v1/SOAP/XML/ItemRatingComments/ApplicationId/{ApplicationId}/UserId/{UserId}?SecretKey={SecretKey}>

Get all Comments created from an Application with ApplicationId and by User with UserIdentifierId-

<http://api.rateitall.com/v1/SOAP/XML/ItemRatingComments/ApplicationId/{ApplicationId}/UserIdentifierId/{UserIdentifierId}?SecretKey={SecretKey}>

Will have the Response XML in the below sample format. There will be multiple XML <ItemRatingComment> elements in the Response depending on the request.



```

<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <ItemRatingComment>
      <CommentId>4</CommentId>
      <RatingId>2180229</RatingId>
      <Comments>Yes, but the safety of others is as important as conducting
      business.</Comments>
      <OwnerUserName>BlueOrchid</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
    </ItemRatingComment>
    <ItemRatingComment>
      <CommentId>5</CommentId>
      <RatingId>2180229</RatingId>
      <Comments>Actually, safety of others is MORE important.</Comments>
      <OwnerUserName>BlueOrchid</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
    </ItemRatingComment>
    <ItemRatingComment>
      <CommentId>17338</CommentId>
      <RatingId>2180229</RatingId>
      <Comments>Blue your right. In a world where nothing is worthy except dineros or a
      career, it's time to prioritize. Sorry but my friends and family take first place (over my
      hamburger slinging job).</Comments>
      <OwnerUserName>NothingButNet</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
    </ItemRatingComment>
    <ItemRatingComment>....
  </Value>
  <Message i:nil="true" />
  <IsSuccess>true</IsSuccess>
</Response>

```

The REST POST Request

Create a new Item Rating Comment -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatingComments/Create?SecretKey={SecretKey}>

will create a new ItemRatingComments with Information specified in the XML POST request.

Below is the sample XML Request for Creating a new ItemRatingComment.

Note: The details of all required and optional elements inside the XML element <ItemRatingComment> have been discussed in Error! Bookmark not defined..

```

<ItemRatingComment xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
  <RatingId>3993257</RatingId>
  <Comments>You are right</Comments>
  <OwnerUserName>userc</OwnerUserName>
</ItemRatingComment>

```

and below is the Sample XML response for creating the New ItemRatingComment



```

<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <ItemRatingComment>
      <CommentId>124</CommentId>
      <RatingId>3993257</RatingId>
      <Comments>You are right</Comments>
      <OwnerUserName>userc</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
    </ItemRatingComment>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>true</IsSuccess>
</Response>

```

Note: 1. This Service can create only one Item Rating comment at a time.

The REST POST Request

Update an Existing ItemRatingComment -

<http://api.rateitall.com/v1/SOAP/XML/ItemRatingComments/Update?SecretKey={SecretKey}>

will update the existing ItemRatingComment information. Below is the sample XML request for updating an existing ItemRatingComment

```

<ItemRatingComment xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <CommentId>124</CommentId>
  <RatingId>3993257</RatingId>
  <Comments>You are right, probably the best review read</Comments>
  <OwnerUserName>userc</OwnerUserName>
</ItemRatingComment>

```

and below is the sample XML response for the ItemRating updates

```

<Response xmlns="http://api.rateitall.com/types" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Value>
    <ItemRatingComment>
      <CommentId>124</CommentId>
      <RatingId>3993257</RatingId>
      <Comments> You are right, probably the best review read </Comments>
      <OwnerUserName>userc</OwnerUserName>
      <OwnerUserIdentifierId i:nil="true" />
    </ItemRatingComment>
  </Value>
  <Message i:nil="true" />
  <IsSuccess>true</IsSuccess>
</Response>

```

Note: 1. On ItemRatingComment updates, CommentId should be specified

2. This Service can update only one Item RatingComment at a time.

