­­­­

|  |  |
| --- | --- |
| **Project Sponsor:** |  |
| **Product Name:** | STNConnect API |
| **Product Version:** | 0.5 |
|  |  |
| **Documentation Type:** | User Guide |
| **Prepared By:** | Hem Upreti |
| **Project Manager:** | M. Erick Smith |

**COPYRIGHT NOTICE**

No portion of this document may be used or reproduced in any form or by any means, or stored in a database or retrieval system, without the written permission of Station Casinos LLC.

Table of Contents

[Document Control 3](#_Toc450291831)

[Convention Used In This Document 3](#_Toc450291832)

[Glossary 3](#_Toc450291833)

[Introduction 3](#_Toc450291834)

[About STNConnect API 3](#_Toc450291835)

[Patron API Service 3](#_Toc450291836)

[Ratings API Service 4](#_Toc450291837)

[Field Mappings 5](#_Toc450291838)

[In-house ratings 5](#_Toc450291839)

[Pari-mutuel ratings 11](#_Toc450291840)

[STNConnect API Security 13](#_Toc450291841)

[API Access 13](#_Toc450291842)

[API Store 15](#_Toc450291843)

[API Credentials 16](#_Toc450291844)

[Appendix A: Workflow Diagram 17](#_Toc450291845)

[Appendix B: Swagger API Documentation 18](#_Toc450291846)

[Appendix C: Property ID 18](#_Toc450291847)

[Appendix D: Location ID 19](#_Toc450291848)

[Appendix E: Game ID (Race) 19](#_Toc450291849)

[Appendix F: Game ID (Sports) 21](#_Toc450291850)

# Document Control

## Convention Used In This Document

The following table shows the conventions used in this document.

|  |  |
| --- | --- |
| Convention | Meaning |
|  | This is a note. |

## Glossary

|  |  |
| --- | --- |
| Term | Definition |
| STNConnect API | STNConnect API is a rest end point API developed to provide external system access to retrieve Patron mag-stripe lookup & post Patron Ratings Information. |
| Patrons | Provides Patron information for provided mag-stripe information related to a Boarding Pass. |
| Ratings | Processing of Points given to a Patron for specific wagers placed on in-house\pari-mutuel games. |

# Introduction

## About STNConnect API

STNConnect API is a rest end point API developed to provide external system access to retrieve Patron mag-stripe lookup & post Patron Ratings Information.

It exposes two endpoints: one for retrieving the Patron and the other for posting the Ratings to Station’s rating system.

# Patron API Service

The Patron end point provides functionality to retrieve Patron information based on the provided mag-stripe number. This will return the patron entity within the response for the provided mag-stripe number. If the provided mag-stripe is not found within the Stations’ CMS, an error code ‘404 Not Found’ will be returned, indicating the provided mag-stripe number was not found within the Stations’ system.

|  |  |
| --- | --- |
|  | Details |
| Base URL (Cloud WSO2) | <http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Patrons/> |
| Base URL (Internal WSO2) | <http://24.234.60.38:8280/stadiumpatron/v1/api/Patrons/> |
| Method Used | **GET** |
| Entity | Patron |
|  | {  "PatronId":"string",  "PatronProfile": {  "FirstName": "string",  "LastName": "string",  "Dob": {  "DT": "DateTime",  "UtcOffsetMinutes": short  }  },  "BoardingPass": [  {  "MagStripe": "string",  "CardNumber": "string"  }  ]  } |
| Sample URL | [http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Patrons/{MagStripeNumber}](http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Patrons/%7bMagStripeNumber%7d)  EXAMPLE:  <http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Patrons/013042151503012508>  <http://24.234.60.38:8280/stadiumpatron/v1/api/Patrons/013042151503012508> |
| Sample Response | {  "PatronId":"3014735",  "PatronProfile": {  "FirstName": "Sarah",  "LastName": "Lebel",  "Dob": {  "DT": "1963-03-03T00:00:00",  "UtcOffsetMinutes": 0  }  },  "BoardingPass": [  {  "MagStripe": "013042151512556431",  "CardNumber": "3014735"  }  ]  } |

# Ratings API Service

The Ratings end point will provide functionality to post\update\void ratings. There are two different URIs defined: in-house and pari-mutuel ratings.

To use this end point, please provide the proper VERB and embed the message in the body (or payload) of the request. **\*Note: Do not provide the rating json information as a parameter of the URI.**

# Field Mappings

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Description** | **Required For** | **Source** |
| propertycode | Id of the property where wager placed | POST, PUT | ID Field from Appendix C |
| locationId | Id of the location where wager placed | POST, PUT | ID Field from Appendix D |
| patronId | Unique ID | POST, PUT | Unique ID received from Patron Data using Patron API call. |
| pointsEarned | Points as decimal | POST, PUT |  |
| transDateTime | Date Time of transaction | POST, PUT | DateTime of the current transaction |
| ratedBy | Client Source Id of the ratings | POST, PUT | Employee ID from the wagering system |
| ratingHostId | Transaction ID | PUT  (Not required for Post) | With PUT transaction Id need to be provided which originally returned with response of POST. |
| cpvBuyIn | Chip\Voucher buy in | POST, PUT | Must provide one or more values as part of request |
| otherBuyIn | Other Buy In Value | POST, PUT |
| cashBuyIn | Cash Buy In Value | POST, PUT |
| markerBuyIn | Marker Buy In Value | POST, PUT |
| watIn | WATIn Buy In Value | POST, PUT |
| actualWin | (CoinOut) | PUT | 0 for Initial Rating Creation. Value of Win/Loss + Handle |
| totalBuyIn | Total Buy In Value | POST, PUT | Must sum to the above provided Buy In Values |
| gameId | Game Id based on wagers placed | POST, PUT | Game Code from Appendix D or Appendix E |
| ratingSourceId | Rating source id passed from wagering system | POST, PUT | Must provide the string value from source of rating |
| ratingStatus | Representation of rating action: "Open", "Update", "Void" | POST, PUT | Values from "Open", "Update", "Void" |

## In-house ratings

|  |  |
| --- | --- |
|  | Details |
| Base URL for Race (Cloud WSO2) | <http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Ratings/inhouserace> |
| Base URL for Race (Internal WSO2) | <http://24.234.60.38:8280/stadiumratings/v1/api/ratings/inhouserace> |
| Base URL for Sports (Cloud WSO2) | <http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Ratings/>sports |
| Base URL for Sports (Internal WSO2) | <http://24.234.60.38:8280/stadiumratings/v1/api/ratings/sports> |
| Method | Post |
| Entity | EventRatingInhouse |
| Sample Request | {  "Rating": {  "Session": {  "locationId": "1",  "gameId": "1"  },  "Action": {  "ratedBy": "sourceReference",  "actualWin": 4500,  "markerBuyIn": 0,  "cpvBuyIn": 10,  "watIn": 5,  "otherBuyIn": 0,  "cashBuyIn": 0,  "totalBuyIn": 15  },  "ratingHostId": "",  "ratingStatus": "Open",  "patronId": "5109926",  "pointsEarned": 100,  "ratingSourceId":"sourceId"  },  "transDateTime": "2016-04-26T00:01:51.526Z",  "PropertyCode": "1"  } |
| Sample Response | {  "Rating": {  "Session": {  "locationId": "1",  "gameId": "1",  "gameDate": "0001-01-01T00:00:00",  "shiftId": null,  "deptId": null,  "areaId": null,  "startDatetime": "0001-01-01T00:00:00"  },  "Action": {  "inhouseWon": 0,  "ratingType": 0,  "ratingFinal": false,  "endDateTime": "0001-01-01T00:00:00",  "timePlayed": null,  "ratedBy": "sourceReference",  "calculationId": null,  "calcTheoPct": null,  "games": null,  "aveBet": 0,  "turnOver": 0,  "actualWin": 4500,  "theoWin": 0,  "markerBuyIn": 0,  "cpvBuyIn": 10,  "watIn": 5,  "otherBuyIn": 0,  "cashBuyIn": 0,  "totalBuyIn": 15,  "progressiveWon": 0  },  "ratingHostId": "41248951873000",  "ratingSourceId": "sourceId",  "ratingStatus": 1,  "patronId": "5109926",  "pointsEarned": 100,  "Other": null  },  "transAction": 0,  "transDateTime": "2016-04-26T00:01:51.526Z",  "transStation": null,  "postedBy": null,  "approvedBy": null,  "Source": null,  "PrioritySpecified": false,  "PropertyCode": "1",  "Timestamp": "0001-01-01T00:00:00",  "Notes": null,  "IsTestMessage": false,  "TestMessageExpirationDateTime": "0001-01-01T00:00:00"  } |

|  |  |
| --- | --- |
|  | Details |
| Base URL for Race (Cloud WSO2) | <http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Ratings/inhouserace> |
| Base URL for Race (Internal WSO2) | <http://24.234.60.38:8280/stadiumratings/v1/api/ratings/inhouserace> |
| Base URL for Sports (Cloud WSO2) | <http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Ratings/>sports |
| Base URL for Sports (Internal WSO2) | <http://24.234.60.38:8280/stadiumratings/v1/api/ratings/sports> |
| Method | Put |
| Entity | EventRatingInhouse |
| Sample Request | {  "Rating": {  "Session": {  "locationId": "1",  "gameId": "1"  },  "Action": {  "ratedBy": "sourceReference",  "actualWin": 0,  "markerBuyIn": 0,  "cpvBuyIn": 10,  "watIn": 5,  "otherBuyIn": 0,  "cashBuyIn": 0,  "totalBuyIn": 15  },  "ratingHostId": "41248951874000",  "ratingStatus": "Update",  "patronId": "5109926",  "pointsEarned": 100,  "ratingSourceId":"sourceId"  },  "transDateTime": "2016-04-26T00:01:51.526Z",  "PropertyCode": "1"  } |
| Sample Response | {  "Rating": {  "Session": {  "locationId": "1",  "gameId": "1",  "gameDate": "0001-01-01T00:00:00",  "shiftId": null,  "deptId": null,  "areaId": null,  "startDatetime": "0001-01-01T00:00:00"  },  "Action": {  "inhouseWon": 0,  "ratingType": 0,  "ratingFinal": false,  "endDateTime": "0001-01-01T00:00:00",  "timePlayed": null,  "ratedBy": "sourceReference",  "calculationId": null,  "calcTheoPct": null,  "games": null,  "aveBet": 0,  "turnOver": 0,  "actualWin": 0,  "theoWin": 0,  "markerBuyIn": 0,  "cpvBuyIn": 10,  "watIn": 5,  "otherBuyIn": 0,  "cashBuyIn": 0,  "totalBuyIn": 15,  "progressiveWon": 0  },  "ratingHostId": "41248951874001",  "ratingSourceId": "sourceId",  "ratingStatus": 3,  "patronId": "5109926",  "pointsEarned": 100,  "Other": null  },  "transAction": 0,  "transDateTime": "2016-04-26T00:01:51.526Z",  "transStation": null,  "postedBy": null,  "approvedBy": null,  "Source": null,  "PrioritySpecified": false,  "PropertyCode": "1",  "Timestamp": "0001-01-01T00:00:00",  "Notes": null,  "IsTestMessage": false,  "TestMessageExpirationDateTime": "0001-01-01T00:00:00"  } |

|  |  |
| --- | --- |
|  | Details |
| Base URL for Race (Cloud WSO2) | <http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Ratings/inhouserace> |
| Base URL for Race (Internal WSO2) | <http://24.234.60.38.160:8280/stadiumratings/v1/api/ratings/inhouserace> |
| Base URL for Sports (Cloud WSO2) | <http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Ratings/>sports |
| Base URL for Sports (Internal WSO2) | <http://24.234.60.38:8280/stadiumratings/v1/api/ratings/sports> |
| Method | Put |
| Entity | EventRatingInhouse |
| Sample Request | {  "Rating": {  "Session": {  "locationId": "1",  "gameId": "1"  },  "Action": {  "ratedBy": "sourceReference",  "actualWin": 0,  "markerBuyIn": 0,  "cpvBuyIn": 10,  "watIn": 5,  "otherBuyIn": 0,  "cashBuyIn": 0,  "totalBuyIn": 15  },  "ratingHostId": "41248951874001",  "ratingStatus": "Void",  "patronId": "5109926",  "pointsEarned": 100,  "ratingSourceId":"sourceId"  },  "transDateTime": "2016-04-26T00:01:51.526Z",  "PropertyCode": "1"  } |
| Sample Response | {  "Rating": {  "Session": {  "locationId": "1",  "gameId": "1",  "gameDate": "0001-01-01T00:00:00",  "shiftId": null,  "deptId": null,  "areaId": null,  "startDatetime": "0001-01-01T00:00:00"  },  "Action": {  "inhouseWon": 0,  "ratingType": 0,  "ratingFinal": false,  "endDateTime": "0001-01-01T00:00:00",  "timePlayed": null,  "ratedBy": "sourceReference",  "calculationId": null,  "calcTheoPct": null,  "games": null,  "aveBet": 0,  "turnOver": 0,  "actualWin": 0,  "theoWin": 0,  "markerBuyIn": 0,  "cpvBuyIn": 10,  "watIn": 5,  "otherBuyIn": 0,  "cashBuyIn": 0,  "totalBuyIn": 15,  "progressiveWon": 0  },  "ratingHostId": "41248951874000",  "ratingSourceId": "sourceId",  "ratingStatus": 4,  "patronId": "5109926",  "pointsEarned": 100,  "Other": null  },  "transAction": 0,  "transDateTime": "2016-04-26T00:01:51.526Z",  "transStation": null,  "postedBy": null,  "approvedBy": null,  "Source": null,  "PrioritySpecified": false,  "PropertyCode": "1",  "Timestamp": "0001-01-01T00:00:00",  "Notes": null,  "IsTestMessage": false,  "TestMessageExpirationDateTime": "0001-01-01T00:00:00"  } |

## Pari-mutuel ratings

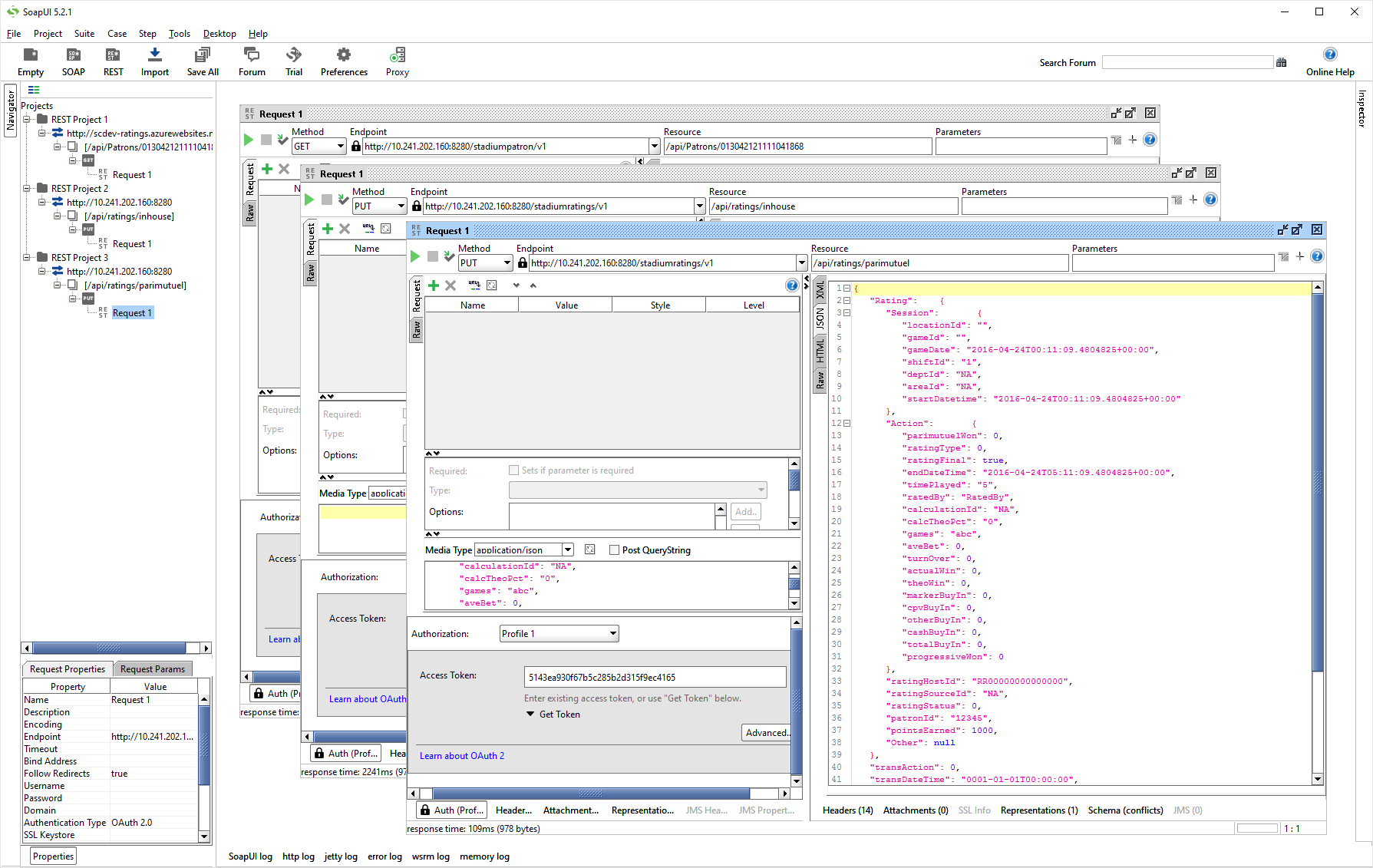
|  |  |
| --- | --- |
|  | Details |
| Base URL (Cloud WSO2) | <http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Ratings/>parimutuel |
| Base URL (Internal WSO2) | <http://24.234.60.38:8280/stadiumratings/v1/api/ratings/parimutuel> |
| Method Available | Post |
| Entity | EventRatingParimutuel |
| Sample Request | {  "Rating": {  "Session": {  "locationId": "1",  "gameId": "35"  },  "Action": {  "ratedBy": "sourceReference",  "actualWin": 0,  "markerBuyIn": 0,  "cpvBuyIn": 10,  "watIn": 5,  "otherBuyIn": 0,  "cashBuyIn": 0,  "totalBuyIn": 15  },  "ratingHostId": "",  "ratingStatus": "Open",  "patronId": "5109926",  "pointsEarned": 100,  "ratingSourceId":"sourceId"  },  "transDateTime": "2016-04-26T00:01:51.526Z",  "PropertyCode": "1"  } |
| Sample Response | {  "Rating": {  "Session": {  "locationId": "1",  "gameId": "35",  "gameDate": "0001-01-01T00:00:00",  "shiftId": null,  "deptId": null,  "areaId": null,  "startDatetime": "0001-01-01T00:00:00"  },  "Action": {  "parimutuelWon": 0,  "ratingType": 0,  "ratingFinal": false,  "endDateTime": "0001-01-01T00:00:00",  "timePlayed": null,  "ratedBy": "sourceReference",  "calculationId": null,  "calcTheoPct": null,  "games": null,  "aveBet": 0,  "turnOver": 0,  "actualWin": 0,  "theoWin": 0,  "markerBuyIn": 0,  "cpvBuyIn": 10,  "watIn": 5,  "otherBuyIn": 0,  "cashBuyIn": 0,  "totalBuyIn": 15,  "progressiveWon": 0  },  "ratingHostId": "41248951875000",  "ratingSourceId": "sourceId",  "ratingStatus": 1,  "patronId": "5109926",  "pointsEarned": 100,  "Other": null  },  "transAction": 0,  "transDateTime": "2016-04-26T00:01:51.526Z",  "transStation": null,  "postedBy": null,  "approvedBy": null,  "Source": null,  "PrioritySpecified": false,  "PropertyCode": "1",  "Timestamp": "0001-01-01T00:00:00",  "Notes": null,  "IsTestMessage": false,  "TestMessageExpirationDateTime": "0001-01-01T00:00:00"  } |

# STNConnect API Security

## API Access

Access to the STNConnect API is controlled through an API Manager. This API manager utilizes token based security to access any Station Casinos API.

Therefore, you will be required to subscribe to the available APIs (listed above) via our API store. From within our API store, you can obtain the proper credentials required to successfully transact with your specific API. Each request to an API will require this token authorization to be included in the header of the requests. Below is an example (using SOAPUI as a testing tool) of the information needed for processing a request (this example show three windows: Patron, Inhouse Ratings and Pari-mutuel Ratings examples):



You will notice the authorization token used for access to the API. You will obtain this authorization token from our API token store. Use of the token will be incorporated within the header as follows:

**GET http://scdev-wso2.westus.cloudapp.azure.com:8280/stadium/v1/Patrons/013042121111041868 HTTP/1.1**

**Accept-Encoding: gzip,deflate**

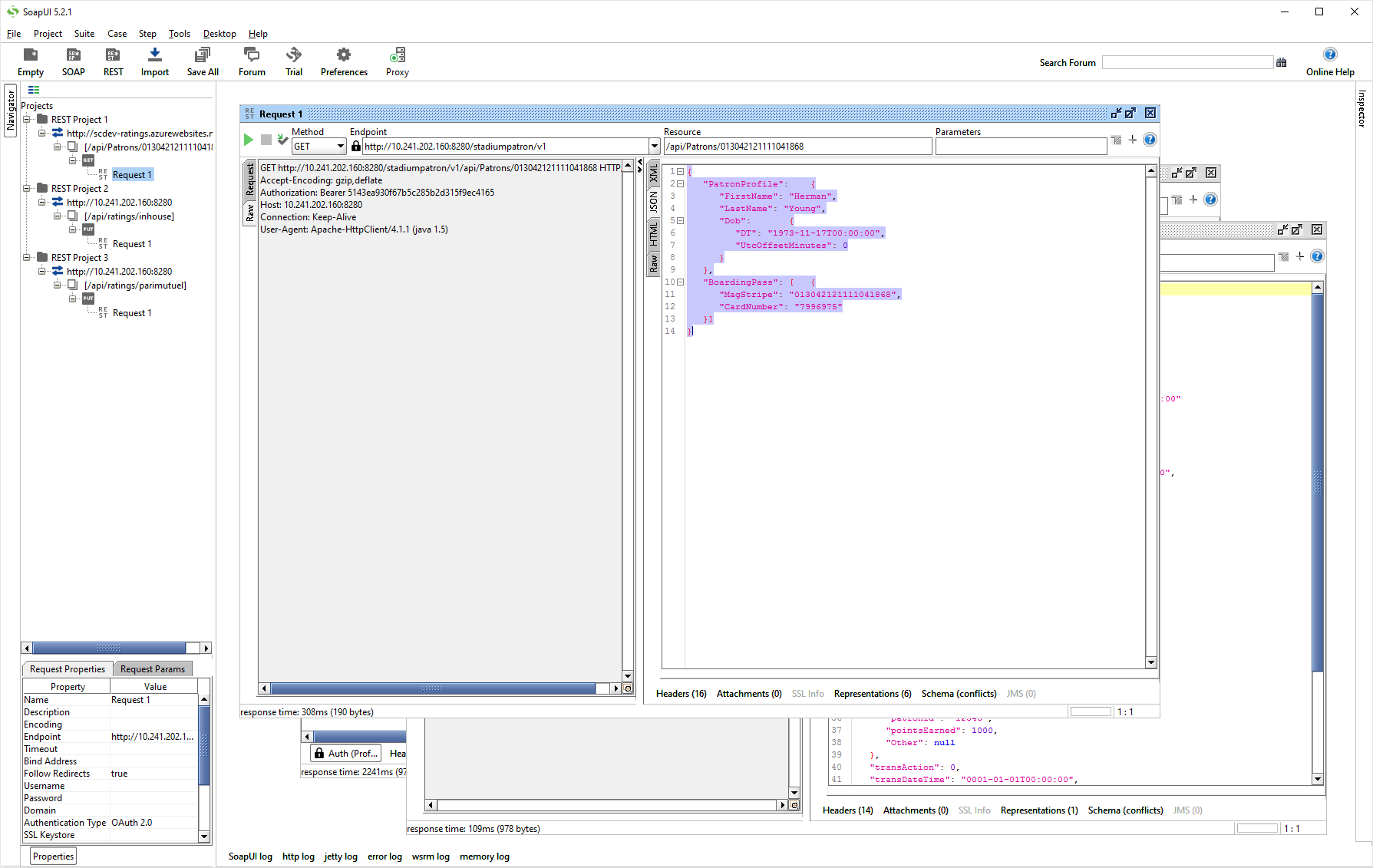
**Authorization: Bearer 5143ea930f67b5c285b2d315f9ec4165 – THIS WILL BE YOUR CURRENT AUTHORIZATION TOKEN**

**Host: scdev-wso2.westus.cloudapp.azure.com:8280**

**Connection: Keep-Alive**

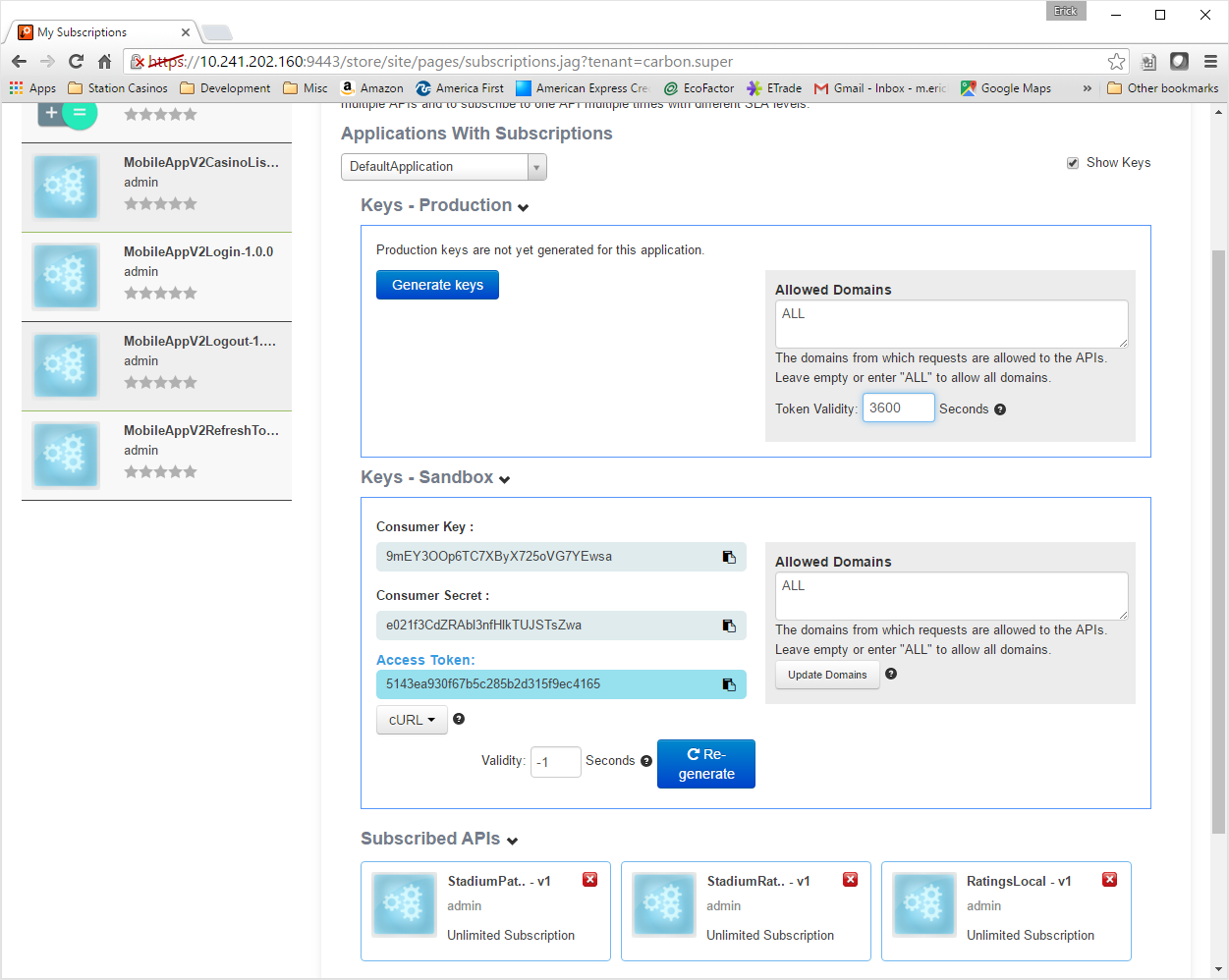
**User-Agent: Apache-HttpClient/4.1.1 (java 1.5)**

Here is a sample within SOAPUI:



## API Store

By accessing the API store, you will be provided the information needed to transact requests in the Station Casinos API manager. An example of this API store is shown below:



You will notice the key elements to the security: Consumer Key, Consumer Secret and the Access Token. The access token (as shown in the SOAPUI example above) is the key used within the header of each request. Without a valid Access Token, you will not be granted rights to execute any Station Casinos API call.

## API Credentials

You specific API credentials, which will grant you access to the Station Casinos API Store, will be provided to you outside this document. These will be used to log into the API store, from which you will be able to control your subscriptions and access credentials.

# Appendix A: Workflow Diagram

The following is a high level workflow diagram illustrating the process of communicating to the Station Casinos API Manger to properly interact with the CMS backend systems.



# Appendix B: Swagger API Documentation

The following are URLs to the current Swagger API documentation for both Patron and Ratings API usage:

<http://scdev-ratings.azurewebsites.net/swagger/ui/index>

<http://scdev-patron.azurewebsites.net/swagger/ui/index>

The

# Appendix C: Property ID

The following are the list of Property Ids:

|  |  |
| --- | --- |
| **ID** | **Description** |
| 2 | Barleys |
| 3 | Boulder Station |
| 4 | Fiesta Rancho |
| 5 | Wildfire Sunset |
| 6 | Green Valley Ranch |
| 7 | Wildfire Lake Mead |
| 8 | Wildfire Boulder |
| 9 | Palace Station |
| 10 | Fiesta Henderson |
| 11 | Wildfire Lanes |
| 12 | Red Rock Station |
| 13 | Santa Fe Station |
| 14 | Sunset Station |
| 15 | The Greens |
| 16 | Texas Station |
| 17 | Wildfire Rancho |
| 18 | Wild Wild West |
| 23 | Wildfire Anthem |
| 24 | Wildfire Valley View |
| 25 | Baldini Reno |

# Appendix D: Location ID

The following are the list of Property Ids:

|  |  |
| --- | --- |
| **ID** | **Description** |
| 1 | Cage |
| 2 | Book |
| 3 | Accounting |
| 4 | Surveillance |
| 5 | PlayerTerminal |
| 6 | CallCenter |
| 7 | RewardsCenter |
| 8 | Kiosk |
| 9 | Web |
| 10 | Mobile |
| 11 | Interactive Voice Response |

# Appendix E: Game ID (Race)

The following is the list of game Ids for race wagers:

|  |  |
| --- | --- |
| **Game Id** | **Game Type** |
| 1 | win |
| 2 | place |
| 3 | show |
| 4 | win & place |
| 5 | win, place & show |
| 6 | win & show |
| 7 | place & show |
| 8 | exacta |
| 9 | exacta box |
| 10 | quinella |
| 11 | quinella box |
| 12 | trifecta |
| 13 | trifecta box |
| 14 | double |
| 15 | pick-3 / win3 |
| 16 | pick-4 / win-4 |
| 17 | pick-5 |
| 18 | pick-6 |
| 19 | pick-7 |
| 20 | pick-8 |
| 21 | pick-9 |
| 22 | pick-10 |
| 23 | twin quinella |
| 24 | twin quinella box |
| 25 | double quinella |
| 26 | double quinella box |
| 27 | double exacta |
| 28 | double exacta: box first race |
| 29 | double exacta: box second race |
| 30 | double exacta: box both races |
| 31 | twin trifecta |
| 32 | twin trifecta box |
| 33 | twin exacta - aka - twin perfecta |
| 34 | twin exacta box |
| 35 | superfecta |
| 36 | superfecta box |
| 37 | tri-super - first half straight |
| 38 | tri-super - first half Box |
| 39 | tri-super - second half straight |
| 40 | tri-super - second half box |
| 41 | trifecta Key box |
| 42 | superfecta Key box |
| 43 | exact5 Key box |
| 44 | twin trifecta Key box |
| 45 | twin superfecta |
| 46 | twin superfecta box |
| 47 | omni pool - show quinella |
| 48 | omni box - show quinella box |
| 49 | Grand Slam – special pick 4 |
| 50 | Trio |
| 51 | First Four |
| 52 | Double Trio |
| 53 | Triple Trio |
| 54 | Exact-5 |
| 55 | Exact-5 Box |
| 56 | Swedish V4 |
| 57 | Swedish V5 |
| 58 | Swedish V64 |
| 59 | Swedish V65 |
| 60 | Swedish V75 |
| 61 | Swedish V86 |
| 62 | racetrack roulette |
| 63 | twin super Key box |
| 64 | tri-super first half Key box |
| 65 | tri-super second half Key box |
| 66 | Perfecta leg in a perfecta-perfecta-tri (PPT) |
| 67 | Perfecta box leg in a PPT |
| 68 | Trifecta leg in a PPT |
| 69 | Trifecta box leg in a PPT |

# Appendix F: Game ID (Sports)

The following are the list of game Ids for sports wagers:

|  |  |
| --- | --- |
| **Game Id** | **Wager Type** |
| 1 | Straight |
| 2 | Parlay |
| 3 | KeyTo |
| 4 | Round Robin |
| 5 | Teaser |
| 6 | Parlay Card |
| 7 | Contest |