# Introduction

1.1 Overview

This document is a guide to configuring the BET FEED SERVICE sample project in a production environment. The BET FEED SERVICE is a windows service with a target .Net Framework 4.5.2 and greater. The BET FEED SERVICE retrieves a Json Object in the form of a string messages securely from RabbitMQ.

1.2 The Connection End-point URLs

Login details and Queue name for RabbitMQ interface can be obtained from the EIS support team. New clients are created by EIS Project Manager. In order to start retrieving sample data, please supply the EIS support team the connecting teams IP address range for whitelisting on our User Acceptance Testing environments.

|  |  |
| --- | --- |
| Live | http://rabbitmq.hollywoodbets.net:15672 |
| UAT | http://demo-rabbitmq.betsolutions.net:15672 |

1.3 Table of content

[Introduction 1](#_Toc65496567)

[Horse Data Structure 2](#_Toc65496568)

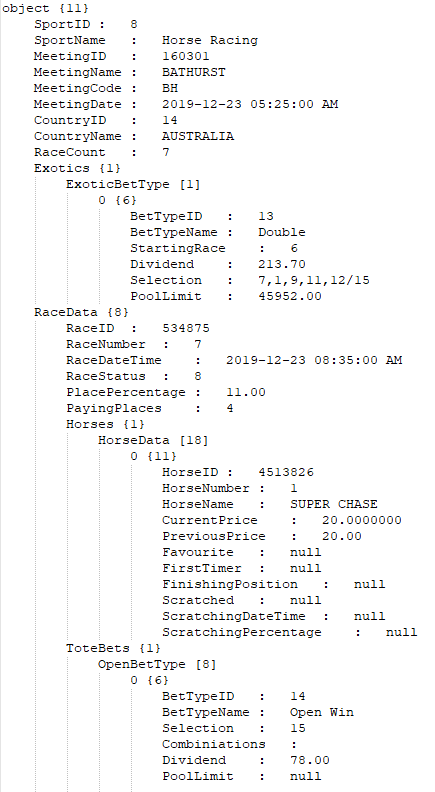
[Soccer Data Structure 6](#_Toc65496569)

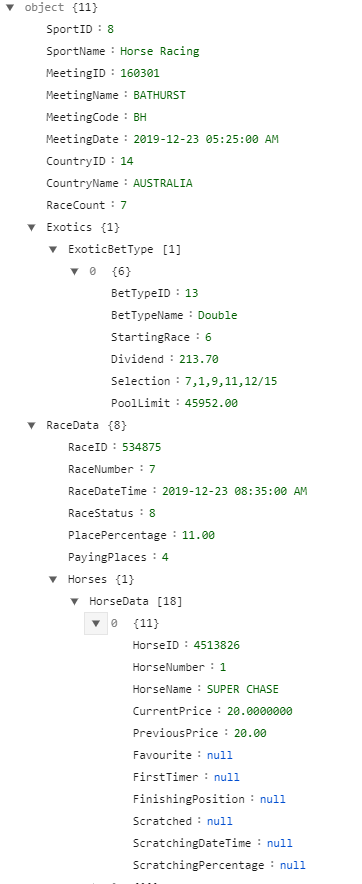
[Lucky numbers DATA Structure 7](#_Toc65496570)

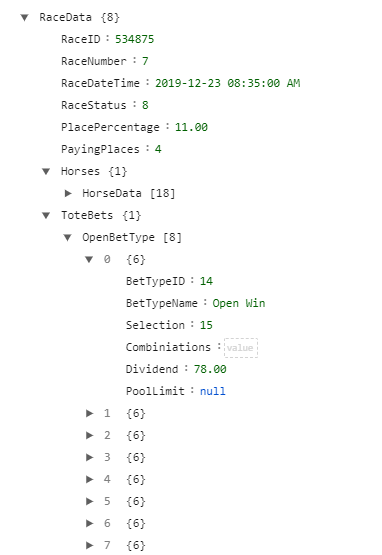
**Horse Integration**

# Horse Data Structure

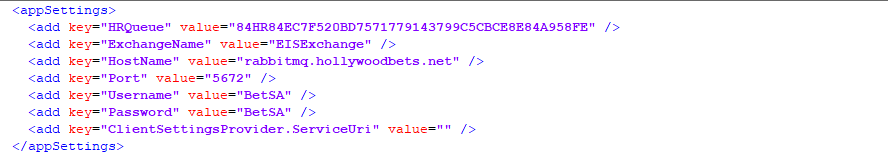
The BET FEED SERVICE receives a standard Json Object string from RabbitMQ ready to be deserialized and saved to database.







Web Config



FeedUpdateRepository.cs

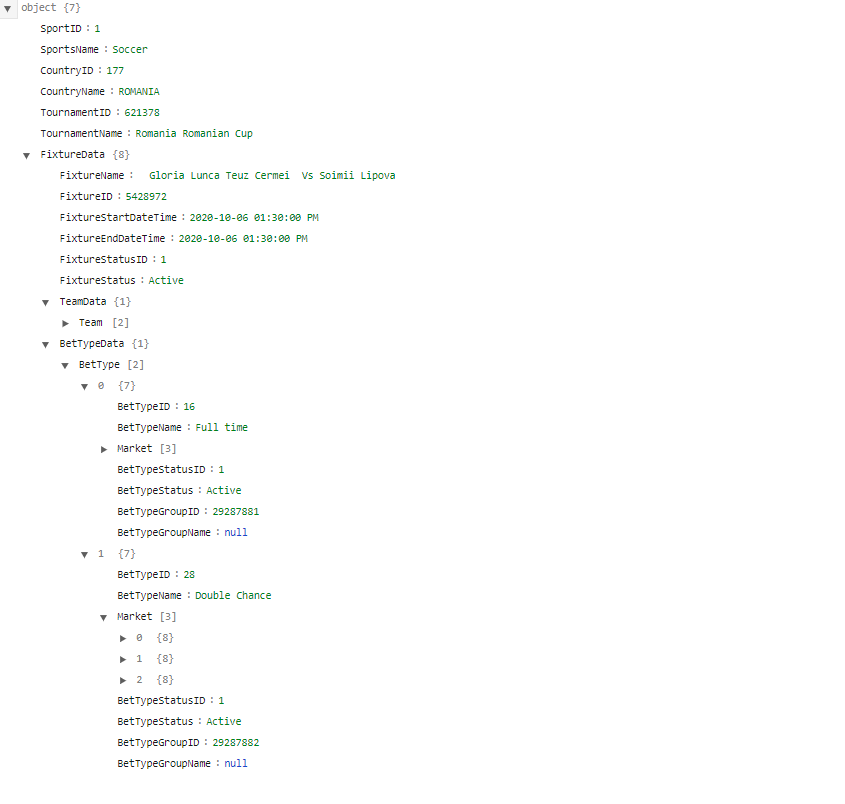
This class accepts the Json and sportID = 8 which is Horse Racing. Once here the data is then ready to be processed.



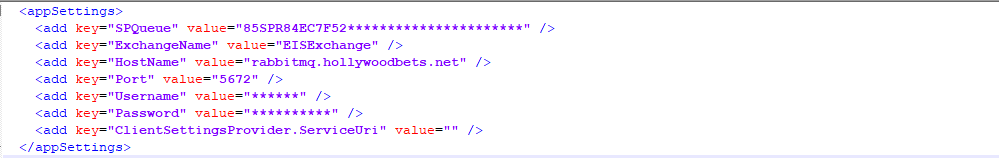
**Soccer Integration**

# Soccer Data Structure

The BET FEED SERVICE receives a standard Json Object string from RabbitMQ ready to be deserialized and saved to database.



Web.Config



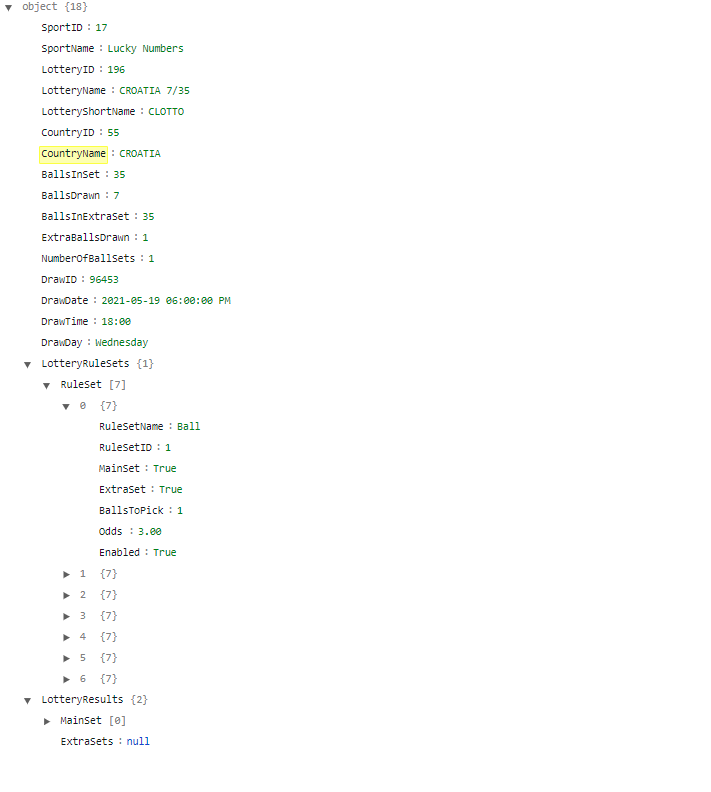
FeedUpdateRepository.cs

This class accepts the Json and sportID = 1 which is soccer. Once here the data is then ready to be processed.



# Lucky numbers DATA Structure

The BET FEED SERVICE receives a standard Json Object string from RabbitMQ ready to be deserialized and saved to database.



Web.Config



FeedUpdateRepository.cs

This class accepts the Json and sportID = 17 which is soccer. Once here the data is then ready to be processed.

