PROJECT UPDATE FORM

**Consultant Name**: Frodo Baggins UnderHill

**Technology**: (Click check box)

Data Science  Exchange Hadoop SharePoint Android iOS

**MARKETING**

**Physical Location During Marketing**: Hobbiton

**Interview Availability During Marketing**:

9am-5pm EST Mon-Fri and 10am-4pm EST Saturday by Appointment

**PROJECT**

**Company Name**: The Hobbit Hole Company

**Dates**: August 18 to Present

**Location**: Mount Doom, Mordor

**Job Title**: Hadoop Big Data Engineer

**Mobile App**: N/A

**Environment**: Apple Private Cloud with on-prem servers

**Technologies Used**: Spark, Spark Streaming, Scala, Kafka

**Project Summary**: (Write a short description of the project: purpose, situation, problem, solution)

I was involved in the modernization of the corporate ETL process. I was responsible for analysis, design, and development of a new enterprise ETL solution. The new Data ETL solution integrates Kafka, Spark, Scala, Cassandra, Avro, Protobuf, and Oracle to deliver a high-throughput of data and transformation from a wide variety of applications like Apple Care, Apple Music, iTunes, iCloud, retailers, etc. I developed event driven solutions with Spark Streaming from Mac OS applications and other data sources in various file formats.

**Project Points**: (Please provide approx. 25 bullet points with keywords): (See HELP below if you need ideas)

* Understanding of business rules, business logic, and use cases to be implemented.
* The solution is intended to be highly customizable by the final user, enabling the end user to create their own data pipelines, transformations and final loading destinations.
* The sources of data can be as disparate as CSV, Protobuf, Avro, Kafka, Teradata, Oracle, etc.
* Users can configure data format, data schema, transformations, joins, filters, and SQL queries. Users can load transformed data to their own platforms, using default Cassandra or Netezza, HDFS, Teradata, Oracle, etc.
* The project initially a POC, to demonstrate parallel data processing using big data ecosystem tools like Spark implemented with Scala.
* Unit testing and Integration testing was performed using ScalaTest.
* Maven used for the managing the project lifecycle, Splunk for log reporting, Hubble for metrics management, and Jenkins for continuous integration.
* Apple proprietary tools were also used.
* Responsible for the development of the Kafka Consumer (Spark-Karka) source code.
* The Kafka Consumer reads from a hard-coded Kafka Topic, creates a dataset and delivers parsed outputs to the console.
* Responsible for development of a Scala application that sends a REST request, with GMT timestamp and gets a response from a web server.
* Responsible for the development of a sample Spark application to read protobuf input from an encoded proto file, using the result classes from protoc compilation to create an RDD and a DataFrame.
* Note that in one case I used Spark Context, which is deprecated and other case with Spark Session, which is the new standard.