Internet Movies Database IMDb

Project Plan

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Purpose of this document 4

1.2 Intended Audience 4

1.3 Scope 4

1.4 Definitions and acronyms 4

1.4.1 Definitions 4

1.4.2 Acronyms and abbreviations 5

1.5 References 5

2. Background and Objectives 5

3. Organization 5

*3.1* Project group 5

3.2 Customer 6

4. Development process 6

*5.* Deliverables 6

6. Project risks 6

7. Communication 7

7.1 Canvas 7

7.2 Git 7

8. Project plan 7

8.1 Time schedule 7

8.1.1 Remarks 7

8.2 Test plan 9

8.2.1 Testing Remarks 10

9. References 10

# Introduction

## Purpose of this document

The purpose of this document is to provide a detailed project description of the application. The Internet Movie Database (IMDb) is a website that acts as a global film database.This website provides a wealth of public information about films, including the title, the year of release, the genre, the audience, critics' ratings, the length of the film, the synopsis of the film, actors, directors, and much more.Given the vast amount of information available on this site, we thought it would be interesting to examine the data on movies on the IMDb website from 2000 to 2017.

Our project focuses on such IMDb datasets, with one dataset focusing on 45,000 movies and containing 26 million ratings from 270,000 users on a scale of 1-5, and the other dataset focusing on 45,000 movies and containing 26 million ratings from 270,000 users.This dataset includes films that were released on or before July 1, 2017.This dataset has been normalized and added to the Ohio S3 bucket.We also took another dataset with 6 million records, which we put in the Oregon S3 bucket, and created a single cluster in Ohio where the tables for all the datasets were created.

## Intended Audience

* team members : Preeti, Yogita, Hrushikesh, Saroj

## Scope

Our project aims to analyse the movies data for two different data sources , one with datalength of 6 million IMDB data and second data source of 45,000 movies with various graphical representations to give an interpretation of these movie datasets.

## Definitions and acronyms

### Definitions

|  |  |
| --- | --- |
| **Keyword** | **Definitions** |
| Project Name | Internet Movie Database : Movie Talk |
| Project Supervisor | Preeti Khatri |
| Project Leader | Yogita Suryavanshi |
| Team Member | Preeti, Yogita, Hrushikesh, Saroj |
| Milestone | Feb 2021 - May 2021 |
| Git | https://github.com/preetikhatrisjsu/Data228\_Project |
| Scrum | An iterative and incremental agile software development method for managing software projects and product or application development |
| Kunagi | Web-based tool for integrated agile project management and collaboration based on Scrum |
| Scrum sprint | Weekly |
| Scrum master | Hrushikesh Pakola |
| Product owner | Saroj Saran |

### Acronyms and abbreviations

|  |  |
| --- | --- |
| **Acronym or**  **abbreviation** | **Definitions** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## References for data source

1. <https://www.imdb.com/interfaces/>
2. <https://www.kaggle.com/rounakbanik/the-movies-dataset>

# Background and Objectives

**Abstract:**

The Internet Movie Database (IMDb) is a website that acts as a global film database. This website provides a wealth of public information about films, including the title, the year of release, the genre, the audience, critics' ratings, the length of the film, the synopsis of the film, actors, directors, and much more. Given the vast amount of information available on this site, we thought it would be interesting to examine the data on movies on the IMDb website from 2000 to 2017.

Our project focuses on such IMDb datasets, with one dataset focusing on 45,000 movies and containing 26 million ratings from 270,000 users on a scale of 1-5, and the other dataset focusing on 45,000 movies and containing 26 million ratings from 270,000 users. This dataset includes films that were released on or before July 1, 2017. This dataset has been normalized and added to the Ohio S3 bucket. We also took another dataset with 6 million records, which we put in the Oregon S3 bucket, and created a single cluster in Ohio where the tables for all the datasets were created.

We have done cross-region replication i.e. when any file is changed or loaded with additional information, be it the Ohio dataset or the Oregon dataset it automatically updates the data in the Ohio cluster, which is used for further processing and analysis.

**Objective**

Our project aims to analyse the movies data for two different data sources , one with datalength of 6 million IMDB data and second data source of 45,000 movies with various graphical representations to give an interpretation of these movie datasets.

# Architecture & High Level Design

1. Using the Talend ETL tool we have cleaned the datasets.As we have two different data sources, data source 1 was of 45000 movie records which were cleaned and transformed using ETL and then the generated CSV was loaded to S3 bucket to create tables in Clusters. Whereas data source 2 consists of millions of IMDb records that were cleaned, transformed, and loaded to the S3 bucket directly.
2. We have loaded the data files from different sources into two different s3 buckets in different regions via Talend Etl Tool. This is done by enabling cross region replication which allows you to replicate data between distant regions to satisfy requirements and minimize latency.
3. By creating a Redshift Cluster we have created the tables and have loaded the files into the tables using the Redshift Query Editor
4. Python stack: Numpy, Matplotlib, Pandas and Seaborn. For visualisation we have used tableau and python
5. Elastic Beanstalk: With Elastic Beanstalk, we can quickly deploy and manage applications in the AWS Cloud without having to learn about the infrastructure that runs those applications. Elastic Beanstalk reduces management complexity without restricting choice or control. You simply upload your application, and Elastic Beanstalk automatically handles the details of capacity provisioning, load balancing, scaling, and application health monitoring.

# Organization

## Project group

|  |  |  |
| --- | --- | --- |
| **Name** | **Initials** | **Responsibility (roles)** |
| Preeti Khatri | PK | Analysis and Development |
| Yogita Suryavanshi | YS | Data sourcing/modeling and Development |
| Hrushikesh Pakola | HP | analysis and visualization |
| Saroj Saran | SS | Development, documentation |

## Customer

The target customers are listed below:

* Customers
* Business Analyst

# Development process

1. Using the Talend ETL tool we have cleaned the datasets.As we have two different data sources, data source 1 was of 45000 movie records which were cleaned and transformed using ETL and then thegenerated CSV was loaded to S3 bucket to create tables in Clusters. Whereas data source 2 consists of millions of IMDb records that were cleaned, transformed, and loaded to the S3 bucket directly.
2. We have loaded the data files from different sources into two different s3 buckets in different regions via Talend Etl Tool. This is done by enabling cross region replication which allows you to replicate data between distant regions to satisfy requirements and minimize latency.
3. By creating a Redshift Cluster we have created the tables and have loaded the files into the tables using Redshift Query Editor
4. Python stack: Numpy, Matplotlib, Pandas and Seaborn. For visualisation we have used tableau and python pandas.
5. Elastic Beanstalk:

With Elastic Beanstalk, we can quickly deploy and manage applications in the AWS Cloud without having to learn about the infrastructure that runs those applications. Elastic Beanstalk reduces management complexity without restricting choice or control. You simply upload your application, and Elastic Beanstalk automatically handles the details of capacity provisioning, load balancing, scaling, and application health monitoring.

# Deliverables

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **To** | **Output** | **Planned week** | **Promised week** | **Late +/-** | **Delivered week** | **Notes** |
| **Data Extraction** | **Data was downloaded from Kaggle and IMDB and made it ready for loading and cleaning** | **Feb 3rd week** | **Feb 3rd week** | **No** | **Feb 3rd week** |  |
| **Data Normalization and cleaning** | **Data was cleaned and normalized using Talend** | **Mar 1 week** | **Mar 1 week** | **No** | **Mar 1 week** |  |
| **Data Loading to S3** | **Data was loaded to s3 using talend** | **Mar 2 week** | **Mar 2 week** | **No** | **Mar 2 week** |  |
| **Data loading to S3 using AWS** | **Data was loaded to another S3 bucket using AWS** | **Mar 3 week** | **Mar 3 week** | **No** | **Mar 3 week** |  |
| **Data loading to Redshift Table** | **Data was loaded to Redshift table using AWS** | **Mar 4 Week** | **Mar 4 Week** | **no** | **Mar 4 Week** |  |
| **Visualization and analytics** | **Data was analysed using python and tableau** | **April 4 week** | **April 4 week** | **NO** | **April 4 week** |  |

# Project risks

|  |  |  |
| --- | --- | --- |
| **Possibility** | **Risk** | **Preventive action** |
| Cost risk for using S3 and Cluster on AWS | it could increased if used in access | we have deleted our cluster when it was not used and also prevented from querying the data using select \* |
| consistency in both the dataset and merge those with common key | it could lead to duplicate data | we have created movie id key in both the dataset to uniquely identified the record |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Communication

all the 4 members were connected through Zoom call weekly

## Collaboration

## Git

All source code and finished documentation will be uploaded to Github repository. ..

Repository URL: <https://github.com/preetikhatrisjsu/Data228_Project>

# Project plan

## Time schedule

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Id** | **Milestone**  **Description** | **Responsible Dept./Initials** | **Finished week** |  |  |  | **Metr.** | **Rem.** |  |
|  |  |  | **Plan** | **Forecast** |  | **Actual** |  |  |  |
|  |  |  |  | **Week** | **+/-** |  |  |  |  |
| 1 | **Data Extraction** | **Preeti** | **Feb 3rd week** | **Feb 3rd week** | **No** | **Feb 3rd week** |  |  |  |
| 2 | **Data Normalization and cleaning** | **Hrushikesh/Saroj** | **Mar 1 week** | **Mar 1 week** | **No** | **Mar 1 week** |  |  |  |
| 3 | **Data Loading to S3** | **Preeti** | **Mar 2 week** | **Mar 2 week** | **No** | **Mar 2 week** |  |  |  |
| 4 | **Data loading to S3 using AWS** | **Saroj** | **Mar 3 week** | **Mar 3 week** | **No** | **Mar 3 week** |  |  |  |
| 5 | **Data loading to Redshift Table** | **yogita** | **Mar 4 Week** | **Mar 4 Week** | **no** | **Mar 4 Week** |  |  |  |
| 6 | **Visualization and analytics** | **Hrushikesh/Yogita** | **April 4 week** | **April 4 week** | **NO** | **April 4 week** |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

### Remarks

|  |  |
| --- | --- |
| **Remark Id** | **Description** |
|  |  |
|  |  |
|  |  |
|  |  |

## Test plan

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test No.** | 001 | **Phase:** | 1 | **Author:** | Preeti | Date: Mar 2021 |
| **Test Category:** | | **Extract data and load into S3** | | |  |  |
| **Software Product:** | | Talend | | | |  |
| **Test Title:** | | Source and destination data count | | | | |
| **Test Purpose:** | | Source and destination data count should match | | | | |
| **Test Setup:** | | configured the talend job in such a way that will show the source and destination data count and if doesn't match job will throw an exception | | | | |
| **Prerequisites:** | | Talend job and source data | | | | |
| **Procedure:** | | configured the talend job in such a way that that will show the source and destination data count and if doesn't match job will throw an exception | | | | |
| **Checks:** | | data count | | | | |
| **Expected Results:** | | Source and destination data count should match | | | | |
| **Result:** | | Source and destination data count matched | | | | |
| **Reason for Failure:** | | No failure | | | | |
| **Remarks:** | |  | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test No.** | 002 | **Phase:** | 1 | **Author:** | Hrushikesh | Date: April 2021 |
| **Test Category:** | | correct data display on graph | | |  |  |
| **Software Product:** | | Tableau | | | |  |
| **Test Title:** | | correct data display on graph | | | | |
| **Test Purpose:** | | correct data should display on graph | | | | |
| **Test Setup:** | | rechecks were done to confirm that displayed data is matching with actual data source | | | | |
| **Prerequisites:** | | data should be loaded in s3 | | | | |
| **Procedure:** | | rechecks were done to confirm that displayed data is matching with actual data source | | | | |
| **Checks:** | | rechecks were done to confirm that displayed data is matching with actual data source | | | | |
| **Expected Results:** | | graph data should match s3 loaded data | | | | |
| **Result:** | | graph data is matching s3 loaded data | | | | |
| **Reason for Failure:** | | No failure | | | | |
| **Remarks:** | |  | | | | |

### Testing Remarks

|  |  |
| --- | --- |
| **Remark Id** | **Description** |
|  |  |
|  |  |
|  |  |
|  |  |

# References

* <https://movietalkwebapp.s3.us-east-2.amazonaws.com/index.html>
* <https://github.com/preetikhatrisjsu/Data228_Project>
* https://www.imdb.com/interfaces/
* <https://www.kaggle.com/rounakbanik/the-movies-dataset>
* [https://images.app.goo.gl/TKejwnoBruxZ7DZ](https://images.app.goo.gl/TKejwnoBruxZ7DZM8)
* <https://www.pinterest.com/pin/243335186087373891/>
* <https://www.lamag.com/culturefiles/covid-19-movie-theaters/>
* <https://www.pinterest.com/pin/547398529686485966/>
* <https://www.pinterest.com/pin/316448311291200389/>