Hanoi University Of Science And Technology
Vol. No.1, Issue No. 01, December 2020 https://www.hust.edu.vn/

Bài báo cáo: Lưu trữ, xử lý và phân tích dữ liệu lớn user log activity

Huy Mac Quang [1], Khanh Ho Quoc [2]

,Ly Tran Thi Cam [3], Hang Vu Thi[4]

[1]*Students of Hanoi University of Science and Technology (Student Code: 20173169)*

[2]*Students of Hanoi University of Science and Technology (Student Code: 20173191)*

[3]*Students of Hanoi University of Science and Technology (Student Code: 20173250)*

[4]*Students of Hanoi University of Science and Technology (Student Code: 20173096)*

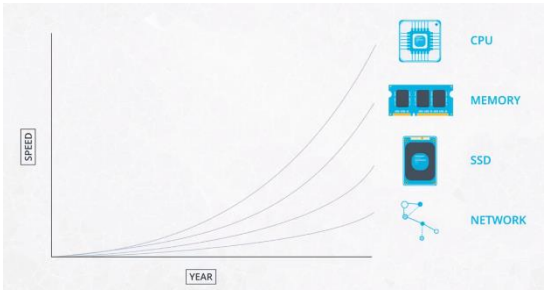
Lưu trữ và xử lý dữ liệu lớn – IT4931 (Class code: 118628)

ABSTRACT

https://github.com/huymq1710/HDFS-MultiNode

https://github.com/huymq1710/Spark-cluster-yarn-mode

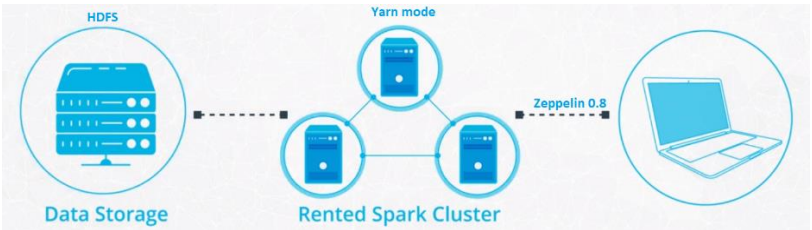
Với 400 users 8GB dữ liệu, có thể lưu và xử lý trong 1 máy nhưng khi lớn hơn. Có thể dùng 12 co-workers laptops, split data cho các máy qua mail, xử lý và mail lại kết quả nhưng "network is suck". Bài báo cáo môn học sẽ lưa trữ dữ liệu trên cụm HDFS, quản lý tài nguyên bằng Yarn và dùng Spark Yarn mode và Zeppelin cho việc interpreter vào dữ liệu trên cụm để xử lý. Tất cả đều được cài đặt trên cụm kết nối với nhau qua mạng Lan.



Keywords: Hadoop, Yarn, Spark, Zeppelin, Bigdata

I. INTRODUCTION

Project cuối kì sẽ kết hợp các bài thực hành phía trước để phục vụ cho việc phân tích và xử lý dữ liệu lần này. Sơ đồ của toàn bộ project:



File log lưu lại user activity của một ứng dụng nghe nhạc, từ tháng 1/2018 đến 10/2018 mỗi tháng ~2,3GB như sau:

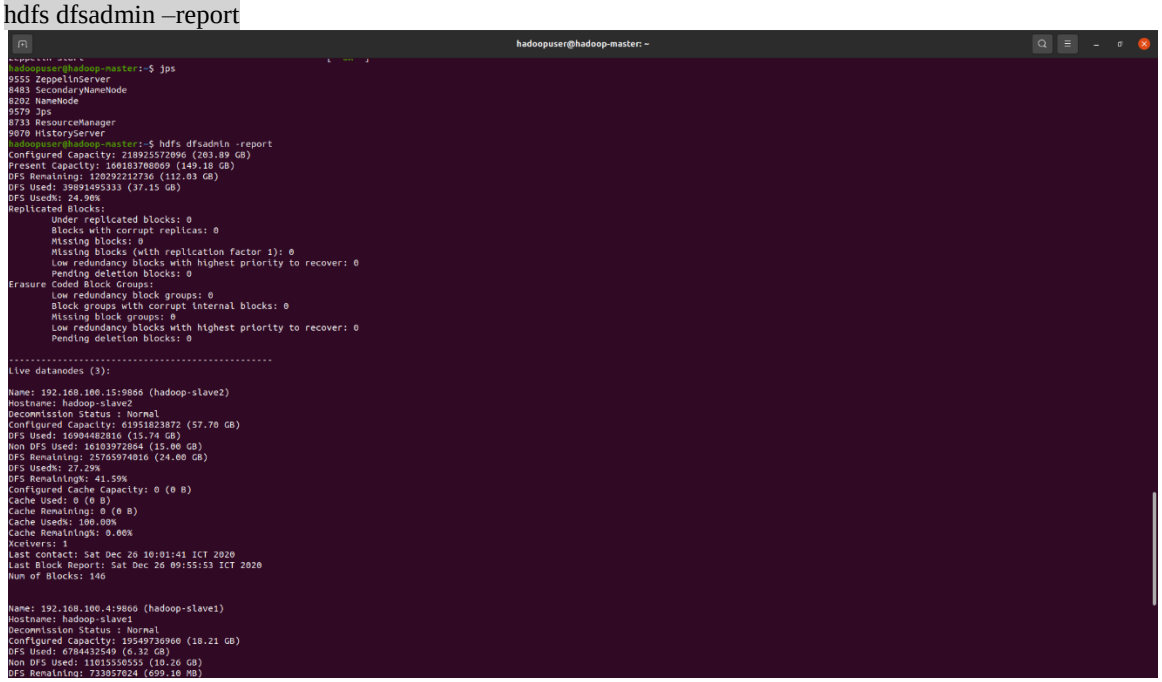
https://github.com/huymq1710/Spark-cluster-yarn-mode/tree/main/Pyspark%20in%20JupyterNotebook

II. Các phần mềm đã cài đặt trên toàn cụm

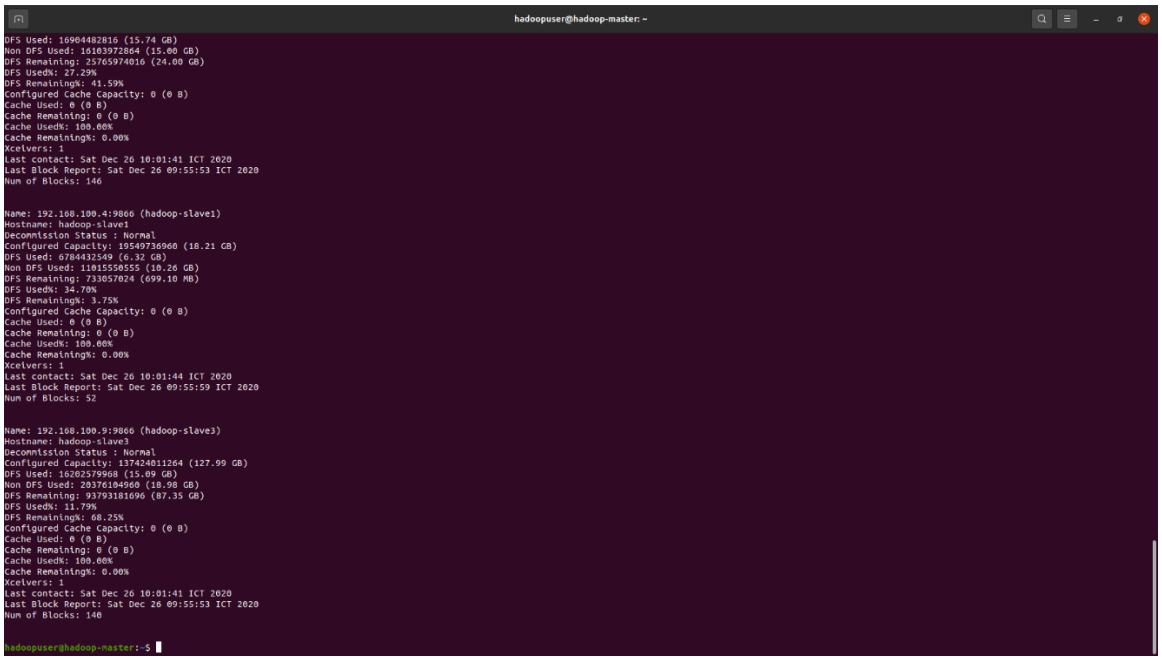
Install Hadoop and Yarn Cluster: https://github.com/huymq1710/HDFS-MultiNode

2.1 Hadoop Distributed File System (HDFS)

hdfs dfsadmin –report

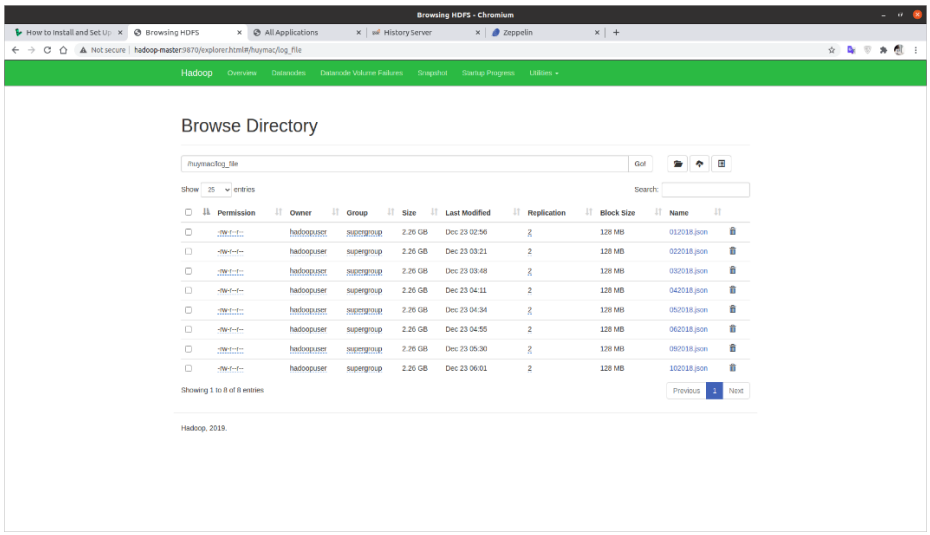


```
hadoopuser@hadoop-master:~$ jps
9555 ZeppelinServer
8483 SecondaryNameNode
8202 NameNode
9579 Jps
8733 ResourceManager
9070 HistoryServer
hadoopuser@hadoop-master:~$ hdfs dfsadmin -report
Configured Capacity: 218925572096 (203.89 GB)
Present Capacity: 160183708069 (149.18 GB)
DFS Remaining: 120292212736 (112.03 GB)
DFS Used: 39891495333 (37.15 GB)
DFS Used%: 24.90%
Replicated Blocks:
        Under replicated blocks: 0
        Blocks with corrupt replicas: 0
        Missing blocks: 0
        Missing blocks (with replication factor 1): 0
        Low redundancy blocks with highest priority to recover: 0
        Pending deletion blocks: 0
Erasure Coded Block Groups:
        Low redundancy block groups: 0
        Block groups with corrupt internal blocks: 0
        Missing block groups: 0
        Low redundancy blocks with highest priority to recover: 0
        Pending deletion blocks: 0

-------------------------------------------------
Live datanodes (3):

Name: 192.168.100.15:9866 (hadoop-slave2)
Hostname: hadoop-slave2
Decommission Status : Normal
Configured Capacity: 61951823872 (57.70 GB)
DFS Used: 16904482816 (15.74 GB)
Non DFS Used: 16163972864 (15.06 GB)
DFS Remaining: 25765974016 (24.00 GB)
DFS Used%: 27.29%
DFS Remaining%: 41.59%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Sat Dec 26 10:01:41 ICT 2020
Last Block Report: Sat Dec 26 09:55:53 ICT 2020
Num of Blocks: 146


Name: 192.168.100.4:9866 (hadoop-slave1)
Hostname: hadoop-slave1
Decommission Status : Normal
Configured Capacity: 19549736960 (18.21 GB)
DFS Used: 6784432549 (6.32 GB)
Non DFS Used: 11015550555 (10.26 GB)
DFS Remaining: 733057024 (699.10 MB)
```

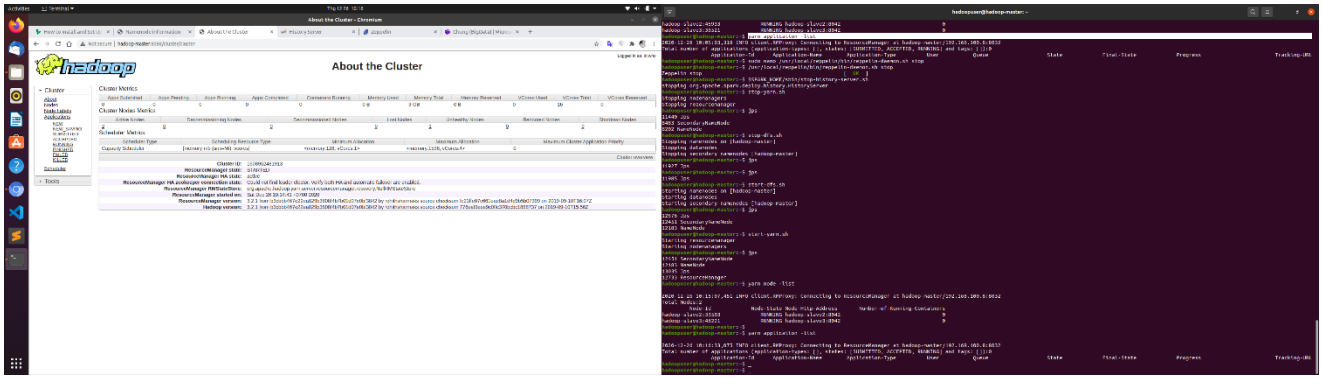


```
DFS Used: 16904482816 (15.74 GB)
Non DFS Used: 16163972864 (15.06 GB)
DFS Remaining: 25765974016 (24.00 GB)
DFS Used%: 27.29%
DFS Remaining%: 41.59%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Sat Dec 26 10:01:41 ICT 2020
Last Block Report: Sat Dec 26 09:55:53 ICT 2020
Num of Blocks: 146


Name: 192.168.100.4:9866 (hadoop-slave1)
Hostname: hadoop-slave1
Decommission Status : Normal
Configured Capacity: 19549736960 (18.21 GB)
DFS Used: 6784432549 (6.32 GB)
Non DFS Used: 11015550555 (10.26 GB)
DFS Remaining: 733057024 (699.10 MB)
DFS Used%: 34.70%
DFS Remaining%: 3.75%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Sat Dec 26 10:01:44 ICT 2020
Last Block Report: Sat Dec 26 09:55:59 ICT 2020
Num of Blocks: 52


Name: 192.168.100.9:9866 (hadoop-slave3)
Hostname: hadoop-slave3
Decommission Status : Normal
Configured Capacity: 137424011264 (127.99 GB)
DFS Used: 16202579968 (15.09 GB)
Non DFS Used: 20376104960 (18.98 GB)
DFS Remaining: 93793181696 (87.35 GB)
DFS Used%: 11.79%
DFS Remaining%: 68.25%
Configured Cache Capacity: 0 (0 B)
Cache Used: 0 (0 B)
Cache Remaining: 0 (0 B)
Cache Used%: 100.00%
Cache Remaining%: 0.00%
Xceivers: 1
Last contact: Sat Dec 26 10:01:41 ICT 2020
Last Block Report: Sat Dec 26 09:55:53 ICT 2020
Num of Blocks: 140

hadoopuser@hadoop-master:~$
```

- Data log file trong HDFS



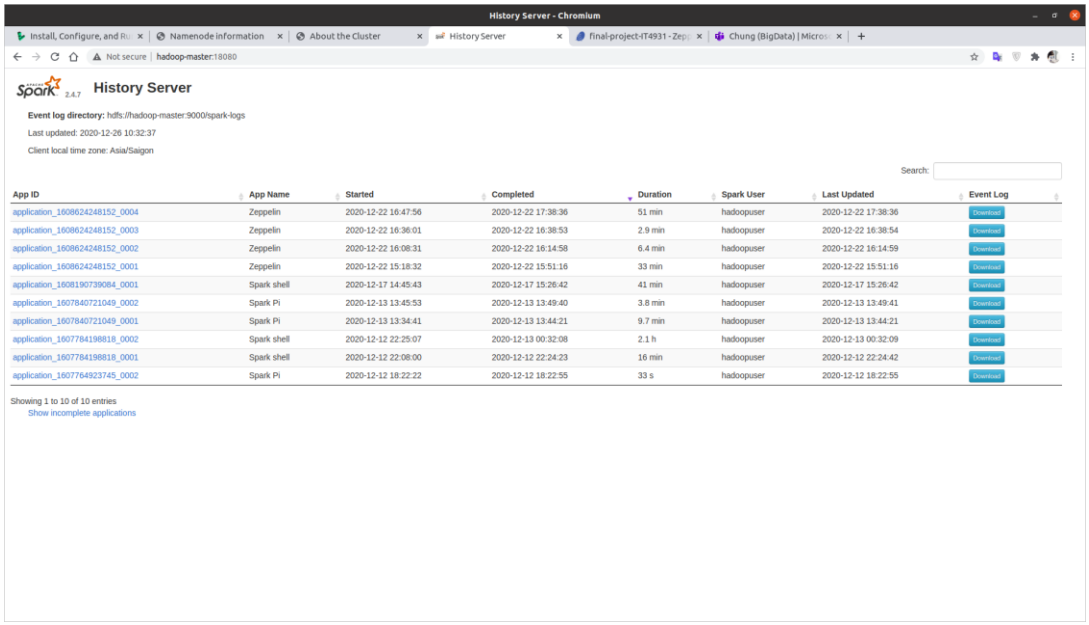
2.2 Yarn web UI



2.3 Spark cluster Yarn mode

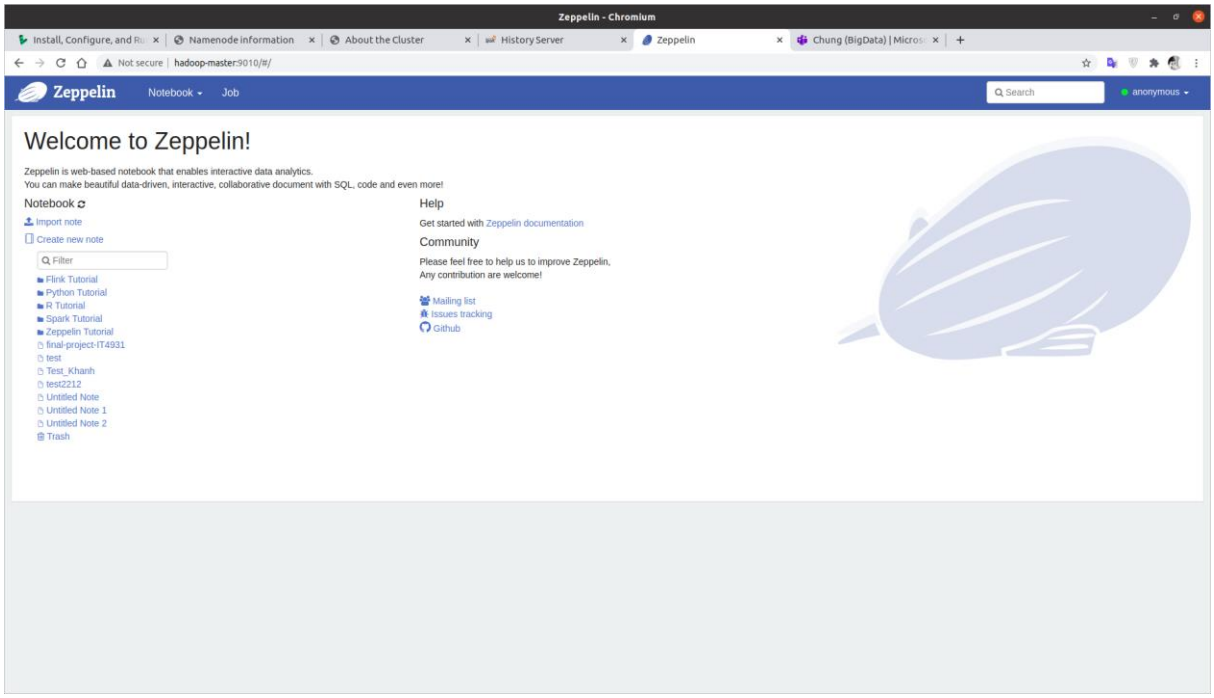
https://www.linode.com/docs/databases/hadoop/install-configure-run-spark-on-top-of-hadoop-yarn-cluster/

- Web UI ghi lại các bản log của các job



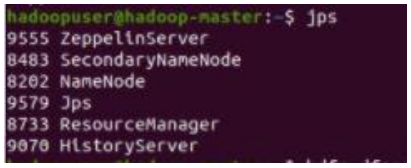
2.4 Zeppelin 0.8

https://medium.com/@shehanfernando/zeppelin-on-spark-cluster-6732d8eb1b59

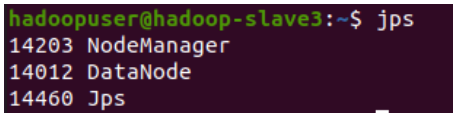


2.5 Jps trên các node

Master:

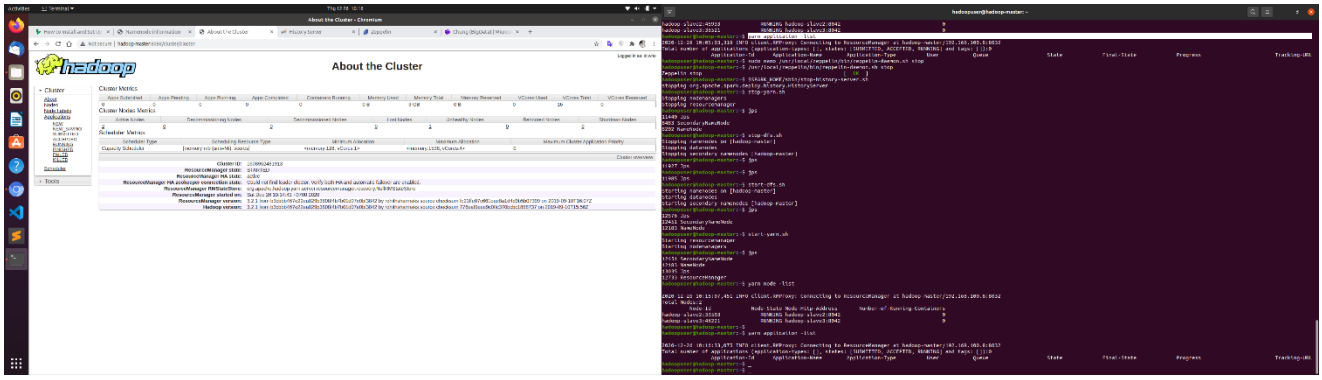


Slave:



2.6 Trường hợp khi 1 node đầy bộ nhớ

Node đó sẽ bị xếp thành unhealthy nodes



III. Thực thi job

3.1 Jps khi job đang được submit

- Master:

```
hadoopuser@hadoop-master:~$ jps
17058 HistoryServer
16515 SecondaryNameNode
18054 SparkSubmit
16246 NameNode
18215 Jps
17143 ZeppelinServer
16731 ResourceManager
```

- Slave 1:

```
hadoopuser@hadoop-slave1:~$ jps
3745 Jps
3349 NodeManager
3174 DataNode
3703 CoarseGrainedExecutorBackend
```

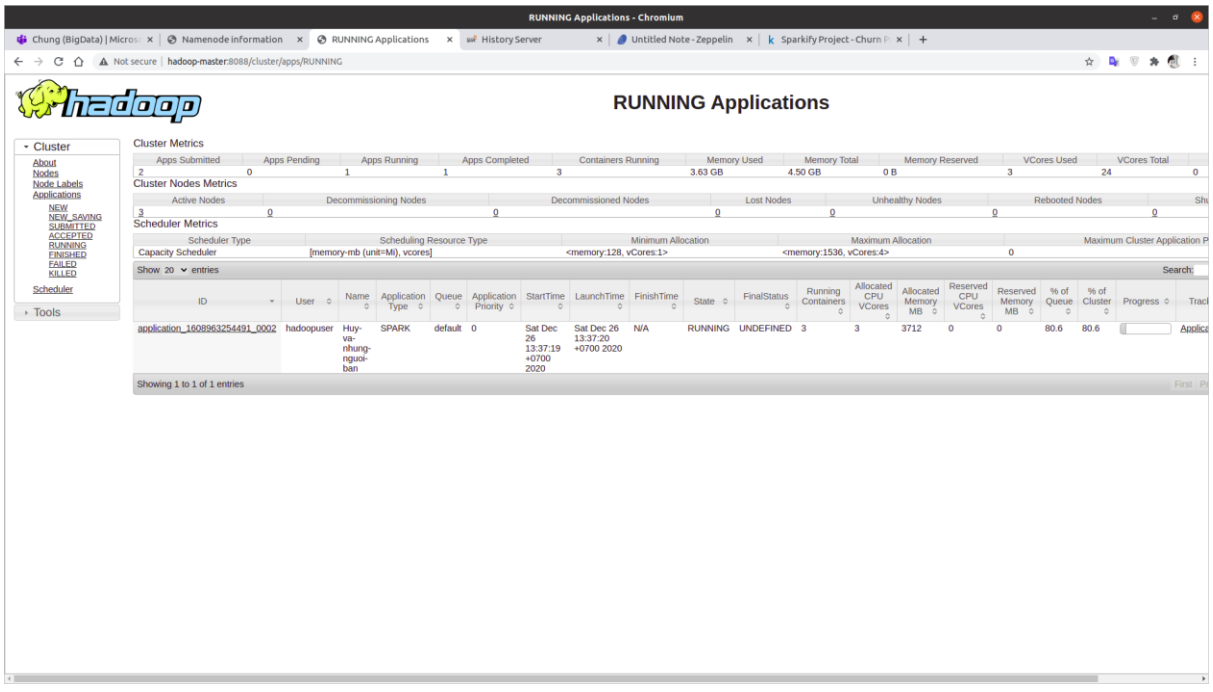
- Slave 2:

```
hadoopuser@hadoop-slave2:~$ jps
8082 CoarseGrainedExecutorBackend
8152 Jps
7657 DataNode
7822 NodeManager
```

- Slave 3:

```
hadoopuser@hadoop-slave3:~$ jps
2633 ExecutorLauncher
2378 NodeManager
2205 DataNode
3869 Jps
```

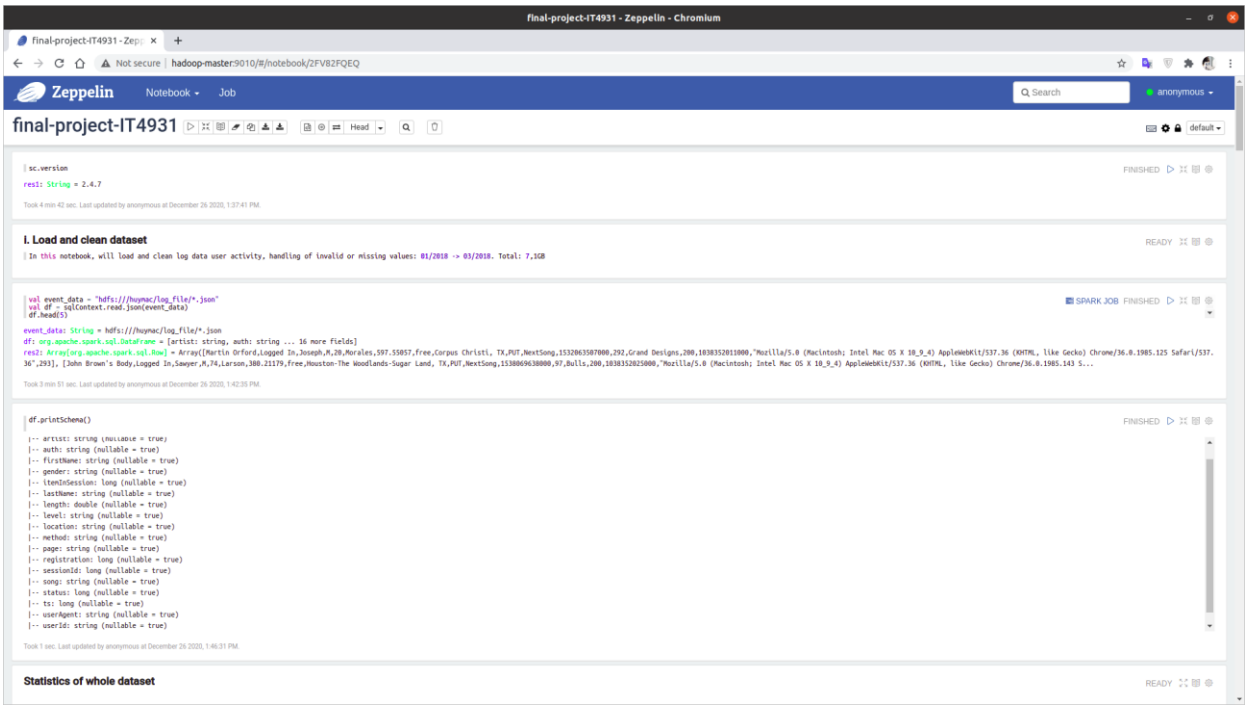
3.2 Check Yarn web UI khi job đang chạy

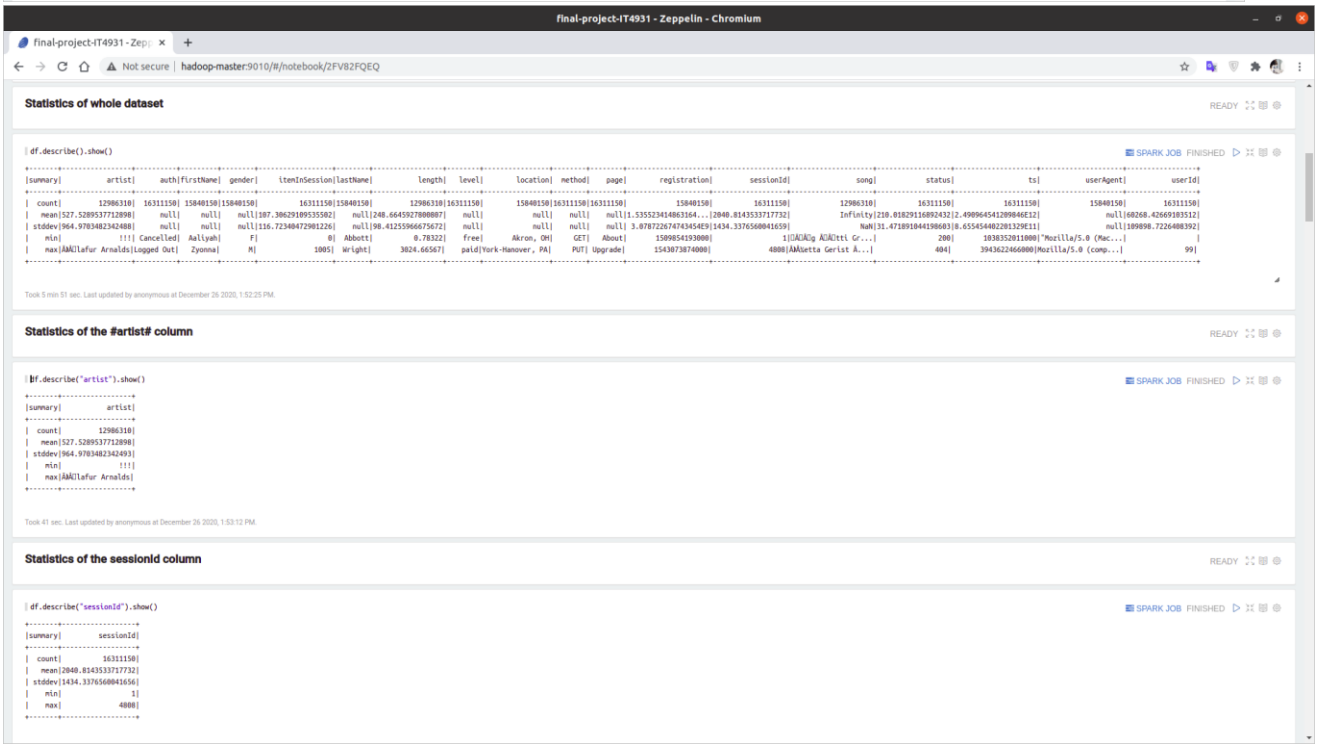


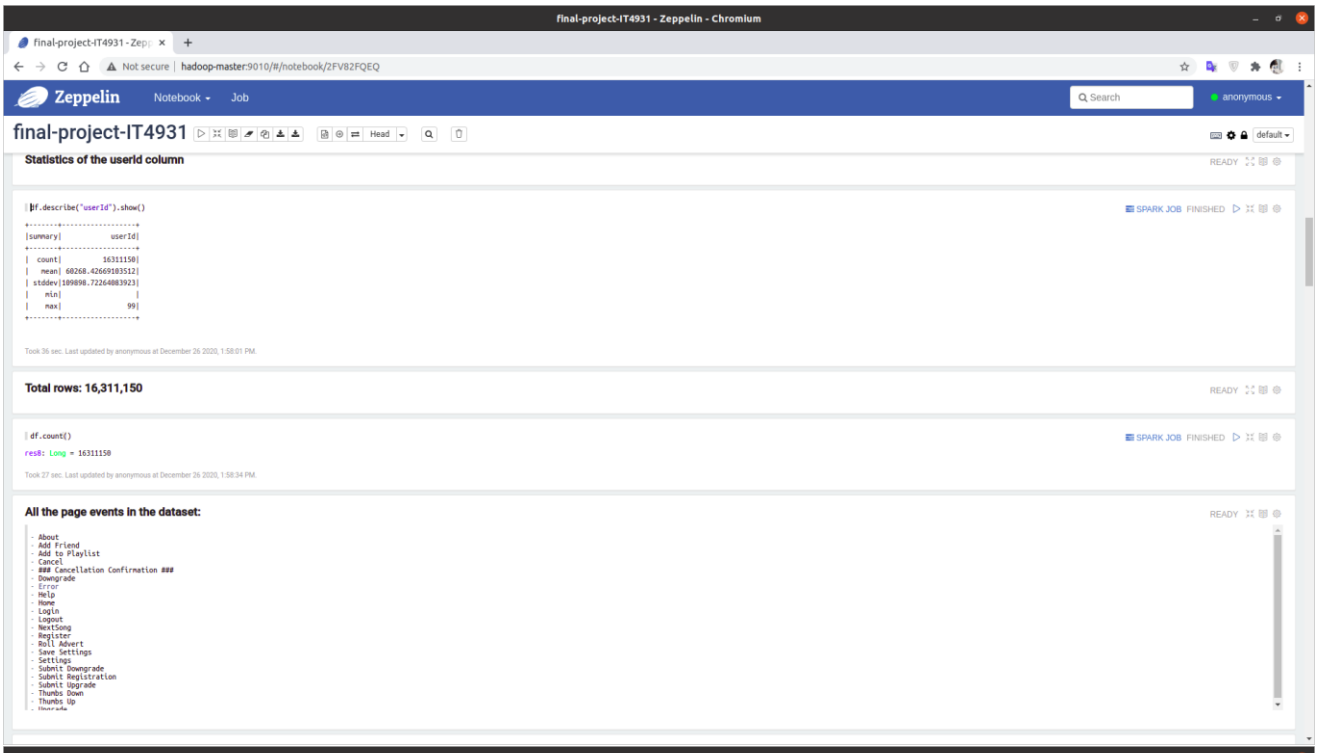
IV. Code trên Zeppelin

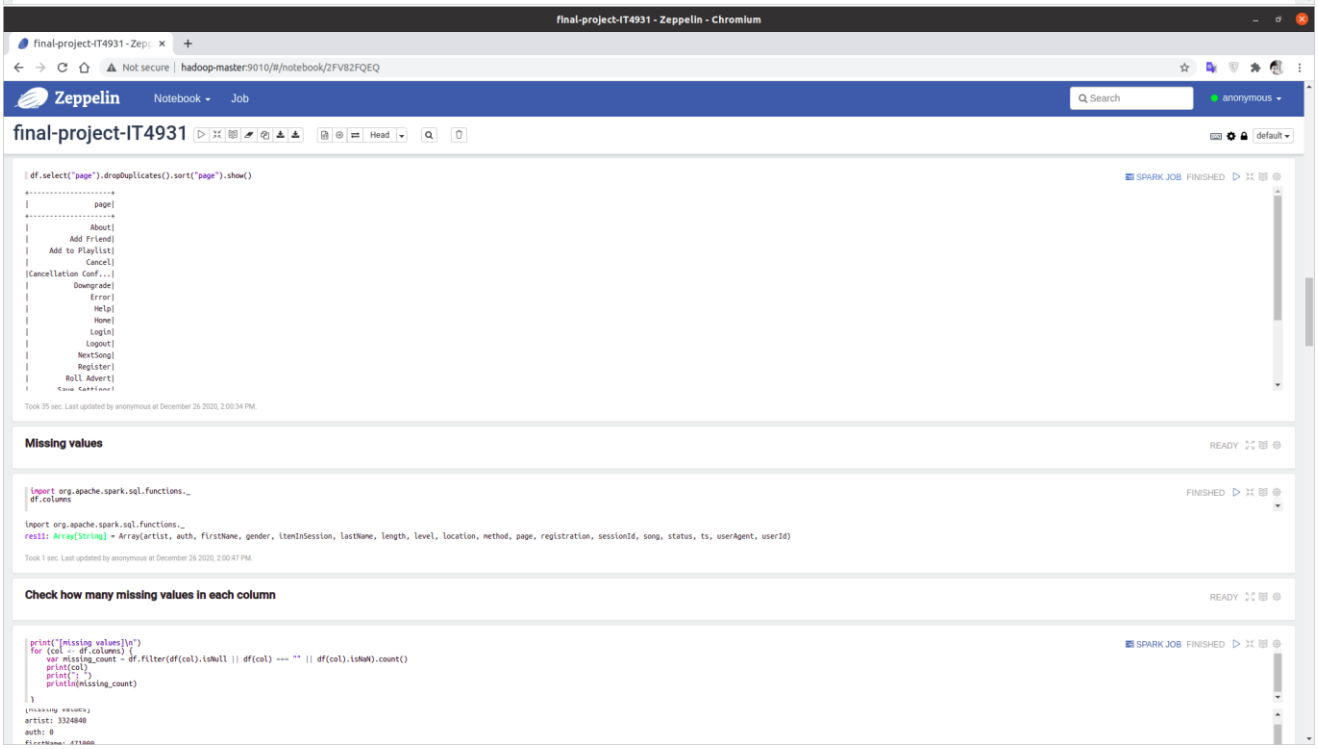
Ngôn ngữ: Scala 2.4.7

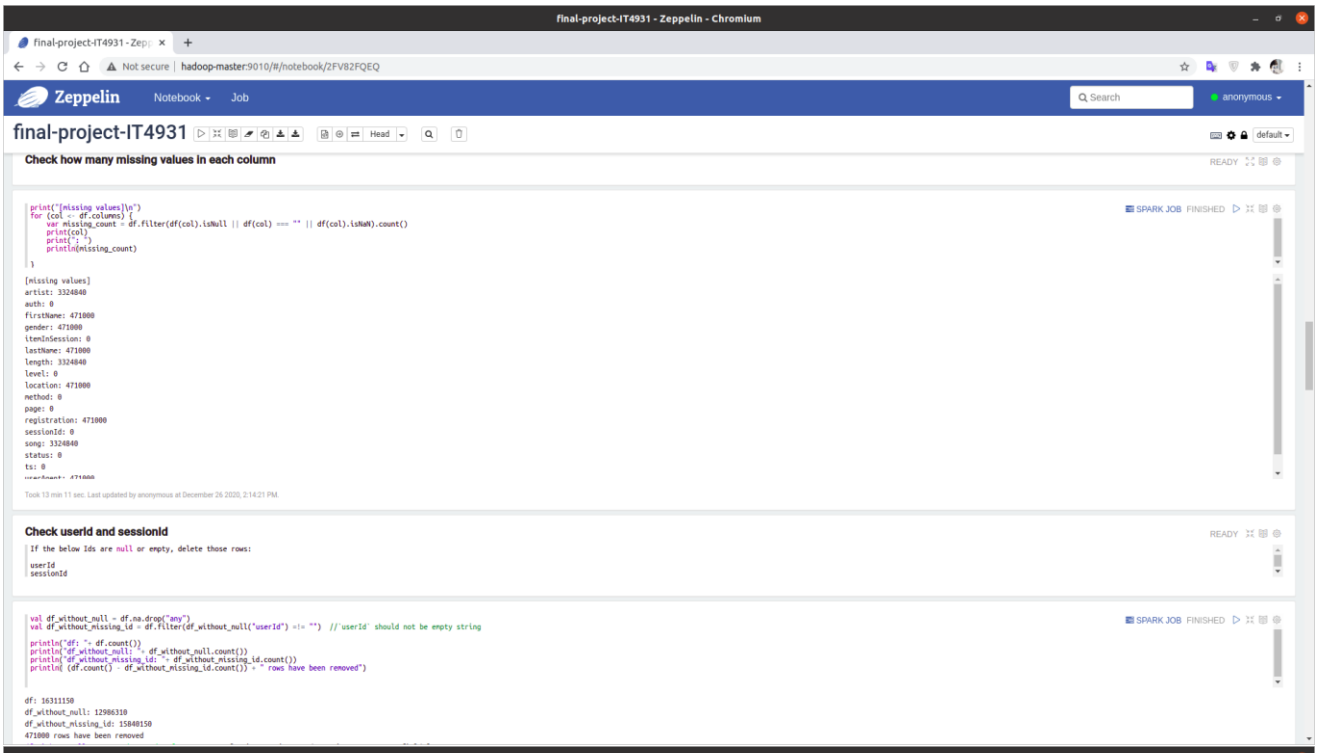
4.1 Load và clean dataset

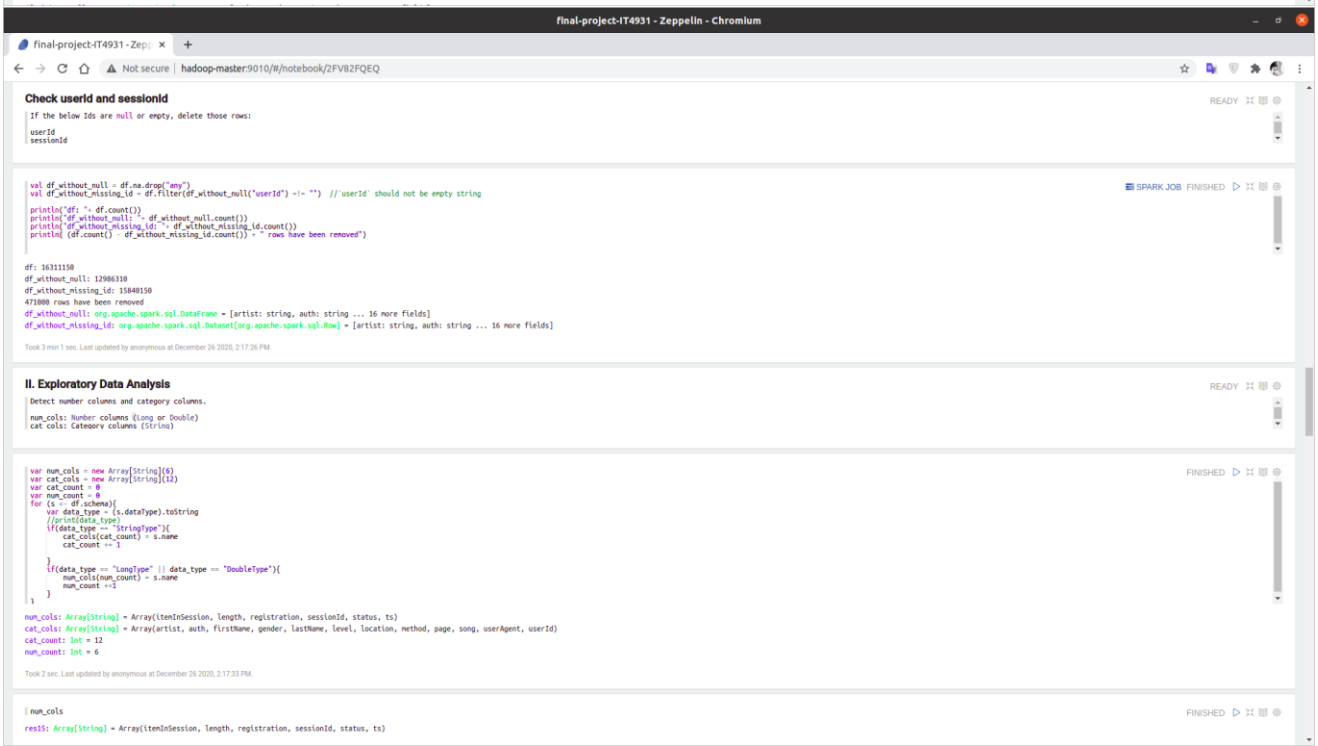




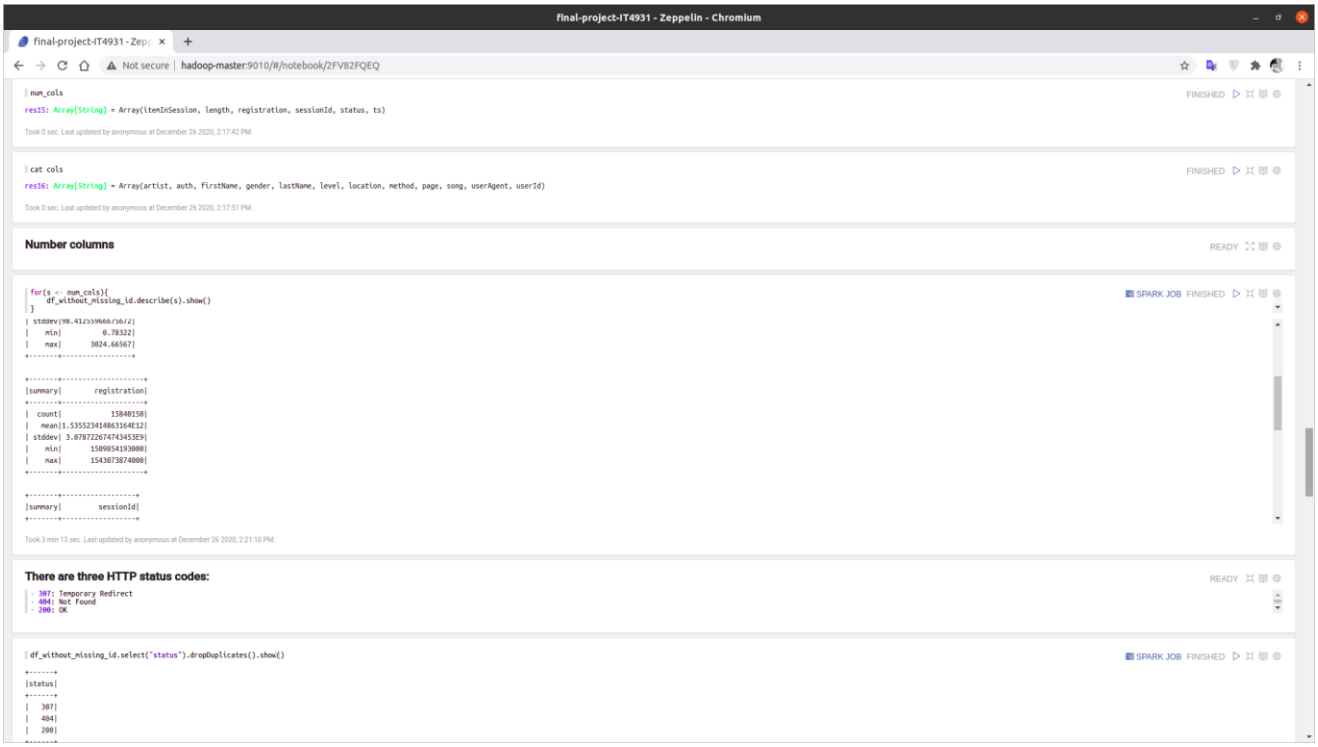

final-project-IT4931 - Zeppelin - Chromium
Zeppelin
Notebook
Job
final-project-IT4931
Statistics of the userId column
df.describe("userId").show()
Total rows: 16,311,150
df.count()
res8: Long = 16311150
All the page events in the dataset:
- About
- Add Friend
- Add to Playlist
- Cancel
- ### Cancellation Confirmation ###
- Downgrade
- Error
- Help
- Home
- Login
- Logout
- NextSong
- Register
- Roll Advert
- Save Settings
- Settings
- Submit Downgrade
- Submit Registration
- Submit Upgrade
- Thumbs Down
- Thumbs Up

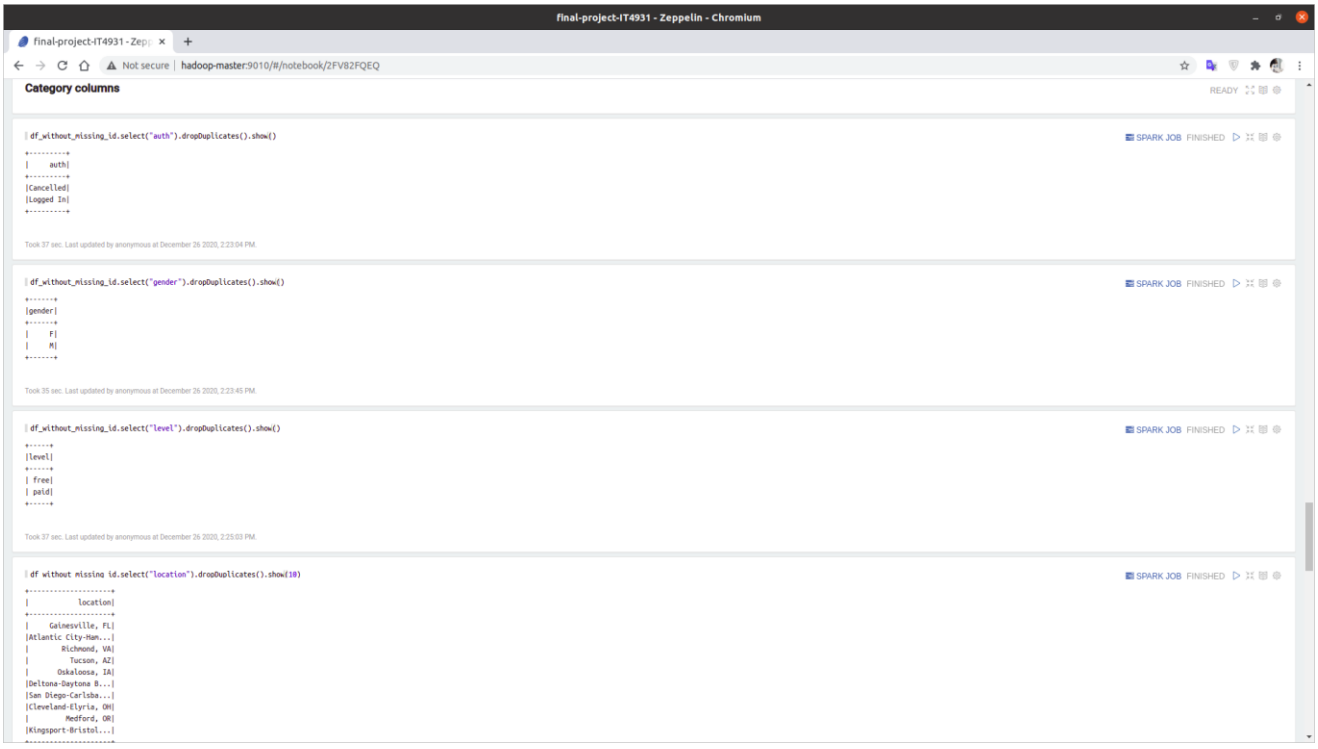

final-project-IT4931 - Zeppelin - Chromium
Zeppelin
Notebook
Job
final-project-IT4931
df.select("page").dropDuplicates().sort("page").show()
Missing values
import org.apache.spark.sql.functions._
df.columns
Check how many missing values in each column
print("[missing values]\n")
artist: 3324840
auth: 0

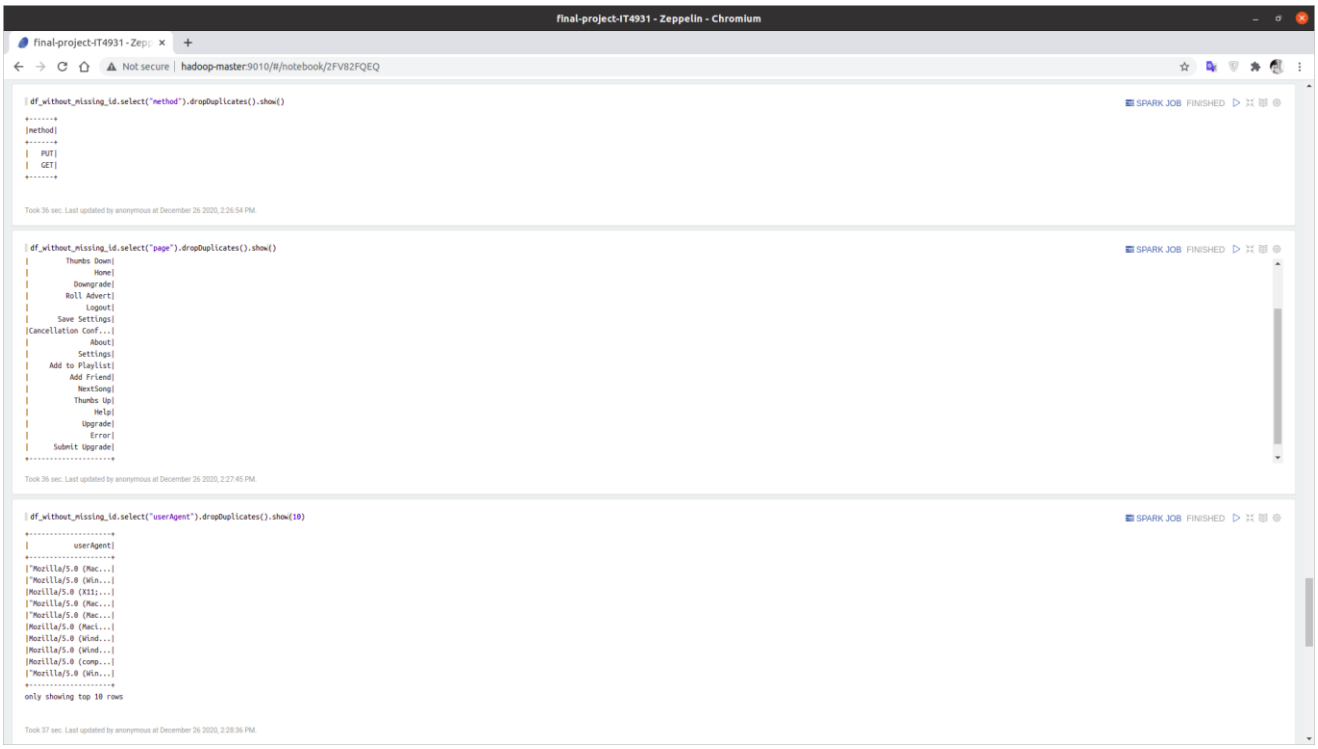

final-project-IT4931 - Zeppelin - Chromium
Zeppelin
Notebook
Job
final-project-IT4931
Check how many missing values in each column
[missing values]
artist: 3324840
auth: 0
firstName: 471000
gender: 471000
itemInSession: 0
lastName: 471000
length: 3324840
level: 0
location: 471000
method: 0
page: 0
registration: 471000
sessionId: 0
song: 3324840
status: 0
ts: 0
Check userId and sessionId
If the below Ids are null or empty, delete those rows:
userId
sessionId
df: 16311150
df_without_null: 12986310
df_without_missing_id: 15840150
471000 rows have been removed

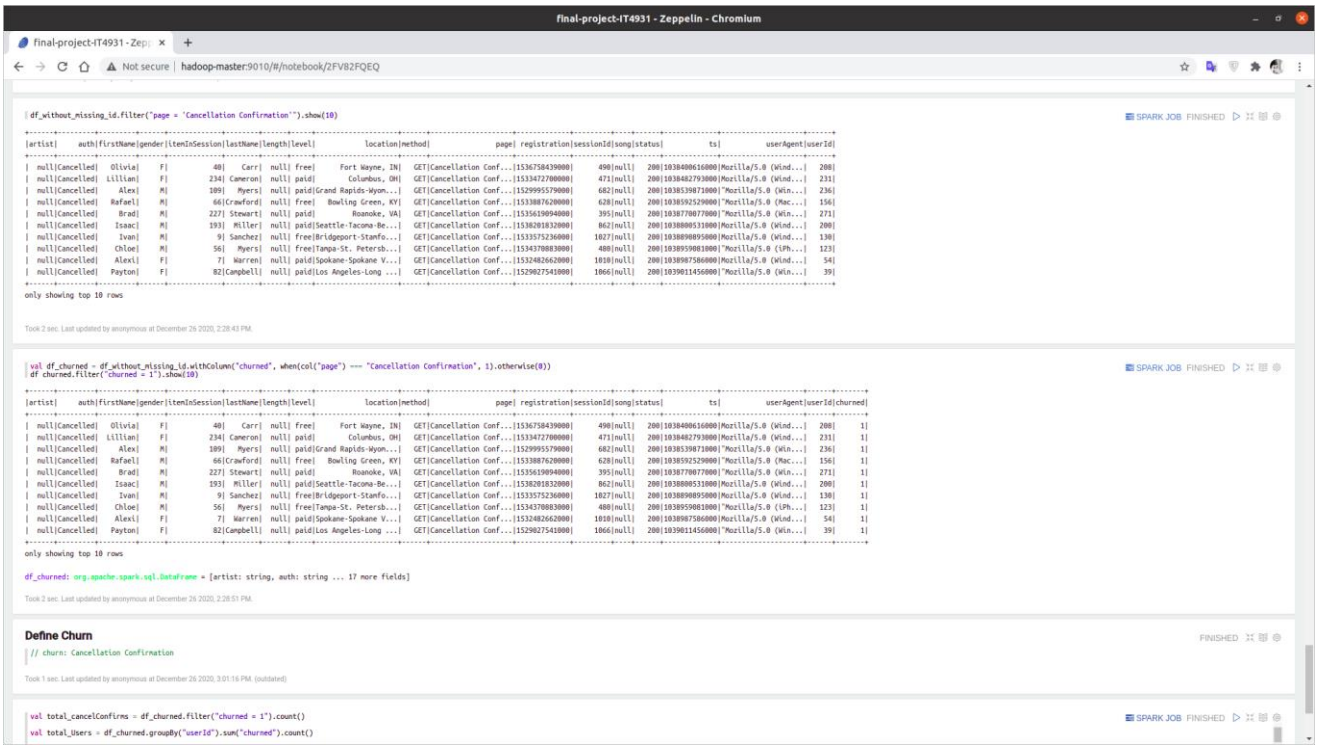

final-project-IT4931 - Zeppelin - Chromium
Check userId and sessionId
If the below Ids are null or empty, delete those rows:
userId
sessionId
df: 16311150
df_without_null: 12986310
df_without_missing_id: 15840150
471000 rows have been removed
II. Exploratory Data Analysis
Detect number columns and category columns.
num_cols: Number columns (Long or Double)
cat_cols: Category columns (String)
num_cols: Array[String] = Array(itemInSession, length, registration, sessionId, status, ts)
cat_cols: Array[String] = Array(artist, auth, firstName, gender, lastName, level, location, method, page, song, userAgent, userId)
cat_count: Int = 12
num_count: Int = 6

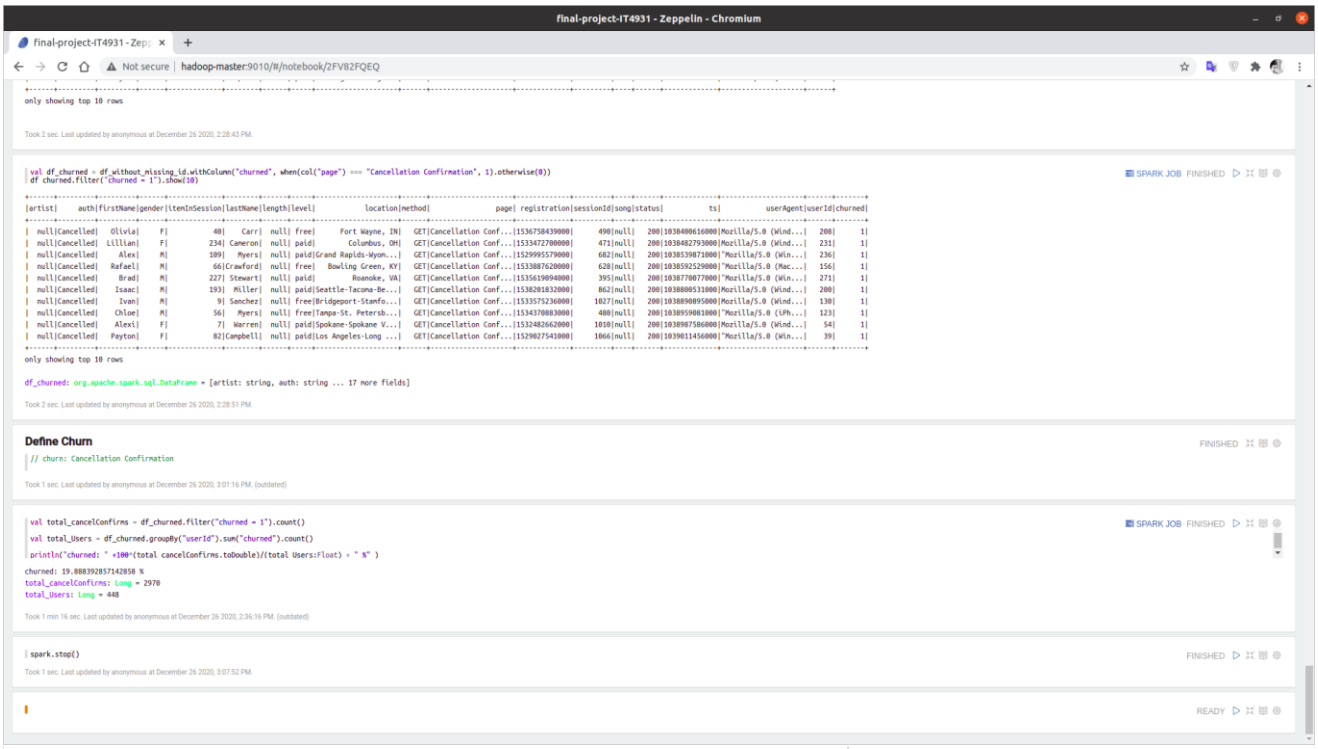
4.2 Exploratory data analysis



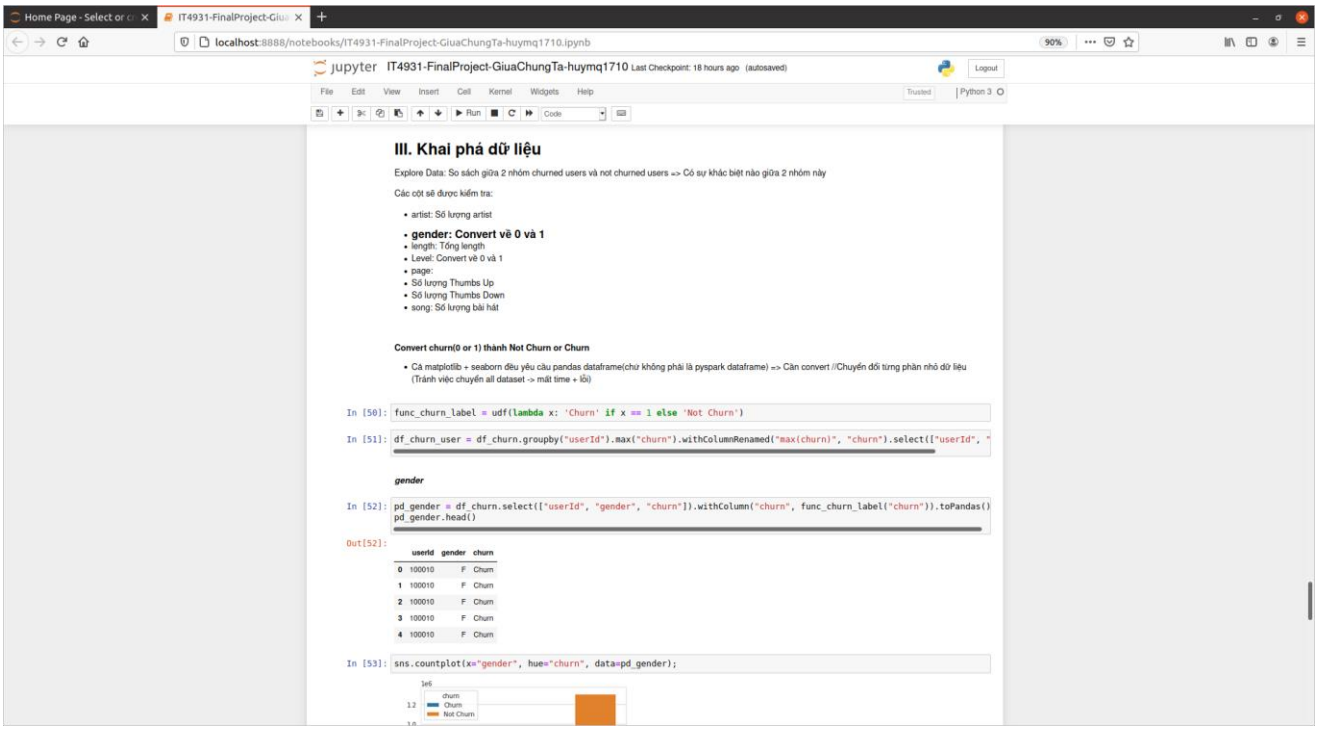


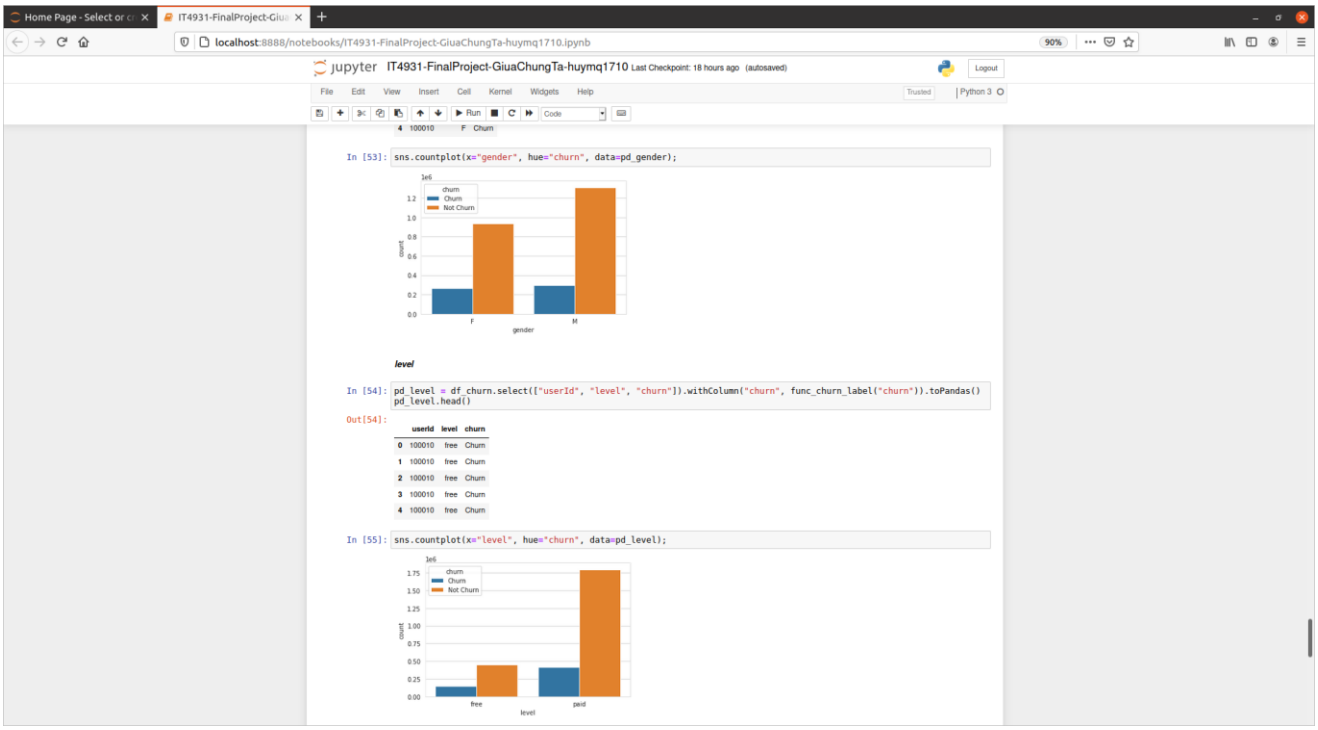

final-project-IT4931 - Zeppelin - Chromium
Not secure | hadoop-master:9010/#/notebook/2FVB2FQEQ
df_without_missing_id.select("method").dropDuplicates().show()
SPARK JOB FINISHED
|method|
| PUT|
| GET|
df_without_missing_id.select("page").dropDuplicates().show()
SPARK JOB FINISHED
| Thumbs Down|
| Home|
| Downgrade|
| Roll Advert|
| Logout|
| Save Settings|
|Cancellation Conf...|
| About|
| Settings|
| Add to Playlist|
| Add Friend|
| NextSong|
| Thumbs Up|
| Help|
| Upgrade|
| Error|
| Submit Upgrade|
df_without_missing_id.select("userAgent").dropDuplicates().show(10)
SPARK JOB FINISHED
only showing top 10 rows

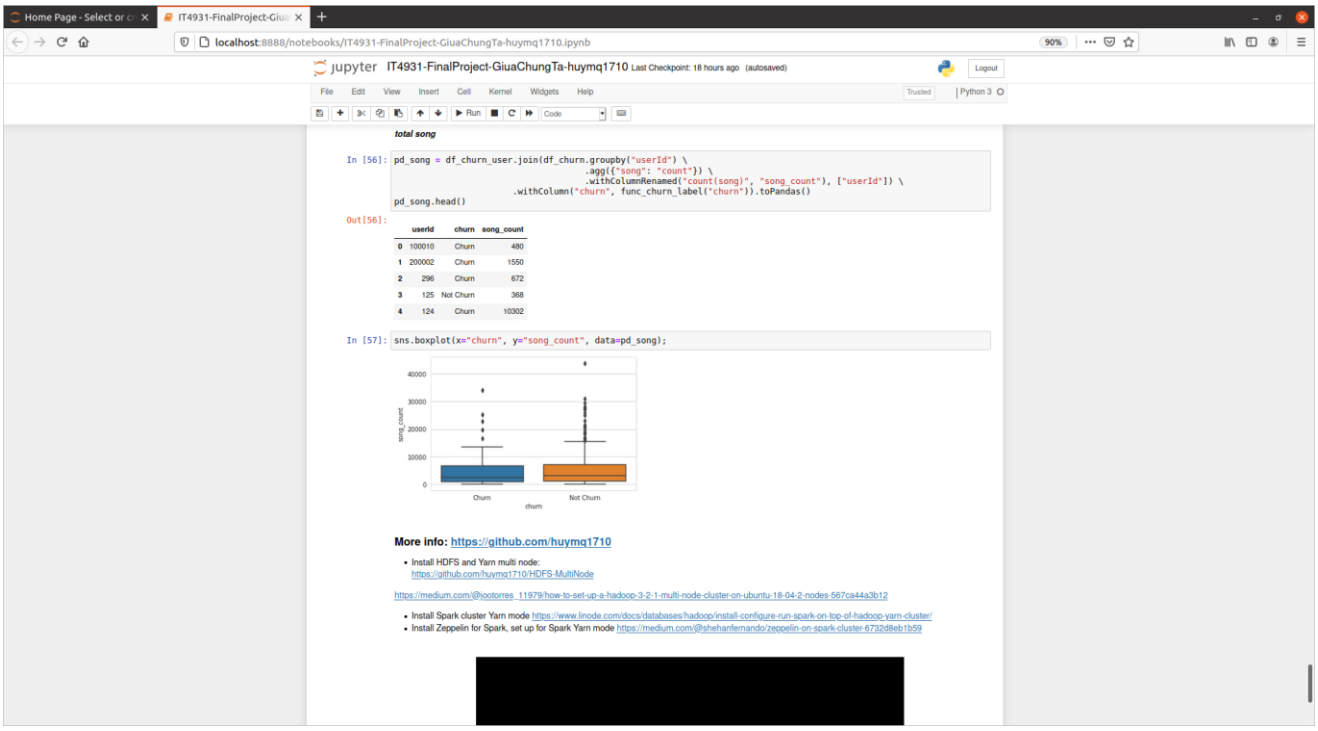

final-project-IT4931 - Zeppelin - Chromium
Not secure | hadoop-master:9010/#/notebook/2FVB2FQEQ
df_without_missing_id.filter("page = 'Cancellation Confirmation'").show(10)
SPARK JOB FINISHED
only showing top 10 rows
val df_churned = df_without_missing_id.withColumn("churned", when(col("page") === "Cancellation Confirmation", 1).otherwise(0))
df_churned.filter("churned = 1").show(10)
SPARK JOB FINISHED
only showing top 10 rows
df_churned: org.apache.spark.sql.DataFrame = [artist: string, auth: string ... 17 more fields]
Define Churn
// churn: Cancellation Confirmation
FINISHED
val total_cancelConfirms = df_churned.filter("churned = 1").count()
val total_Users = df_churned.groupBy("userId").sum("churned").count()
SPARK JOB FINISHED

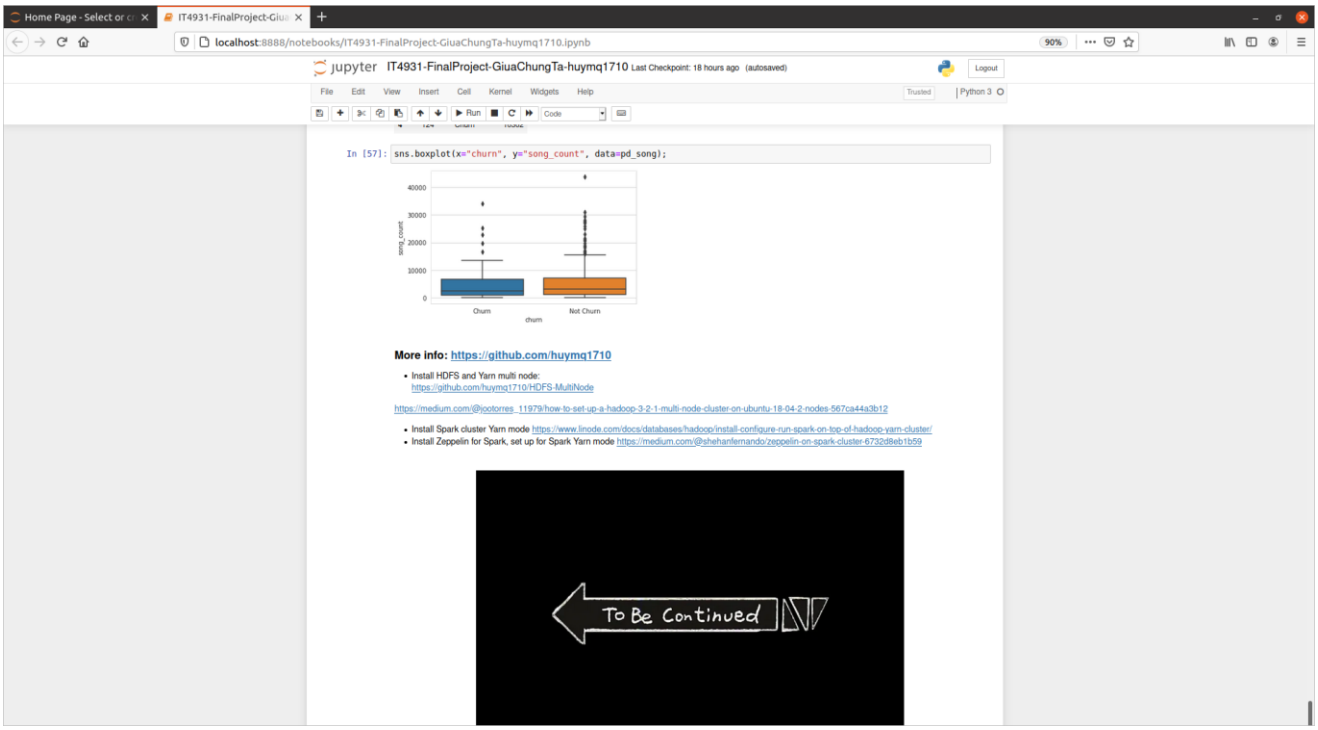


4.3 Khai phá dữ liệu sử dụng pyspark trên Jupyter notebook

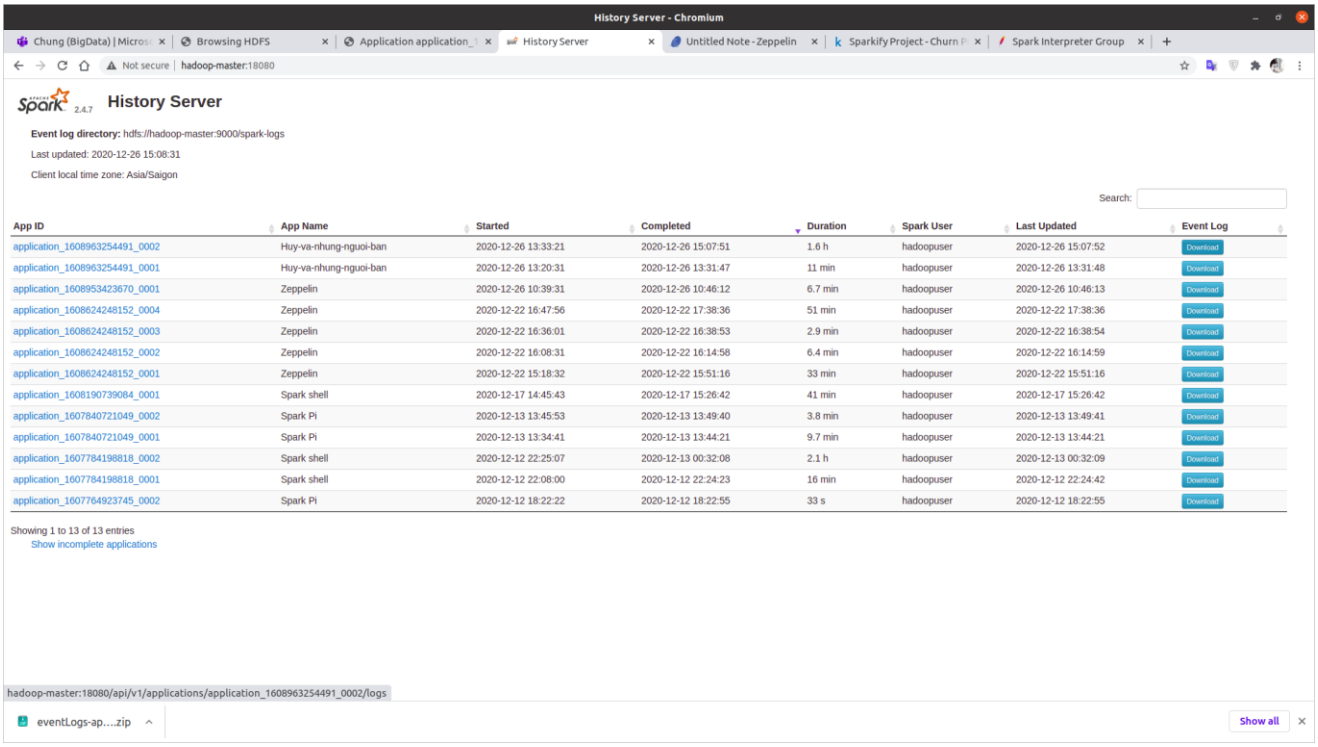

Home Page - Select or cr
IT4931-FinalProject-Giua
localhost:8888/notebooks/IT4931-FinalProject-GiuaChungTa-huymq1710.ipynb
jupyter IT4931-FinalProject-GiuaChungTa-huymq1710 Last Checkpoint: 18 hours ago (autosaved)
III. Khai phá dữ liệu
Explore Data: So sách giữa 2 nhóm churned users và not churned users => Có sự khác biệt nào giữa 2 nhóm này
Các cột sẽ được kiểm tra:
artist: Số lượng artist
gender: Convert về 0 và 1
length: Tổng length
Level: Convert về 0 và 1
page:
Số lượng Thumbs Up
Số lượng Thumbs Down
song: Số lượng bài hát
Convert churn(0 or 1) thành Not Churn or Churn
Cả matplotlib + seaborn đều yêu cầu pandas dataframe(chứ không phải là pyspark dataframe) => Cần convert //Chuyển đổi từng phần nhỏ dữ liệu (Tránh việc chuyển all dataset -> mất time + lỗi)
In [50]: func_churn_label = udf(lambda x: 'Churn' if x == 1 else 'Not Churn')
In [51]: df_churn_user = df_churn.groupby("userId").max("churn").withColumnRenamed("max(churn)", "churn").select(["userId",
gender
In [52]: pd_gender = df_churn.select(["userId", "gender", "churn"]).withColumn("churn", func_churn_label("churn")).toPandas()
pd_gender.head()
Out[52]:
userId gender churn
0 100010 F Chum
1 100010 F Chum
2 100010 F Chum
3 100010 F Chum
4 100010 F Chum
In [53]: sns.countplot(x="gender", hue="churn", data=pd_gender);
1e6
churn
Churn
Not Churn

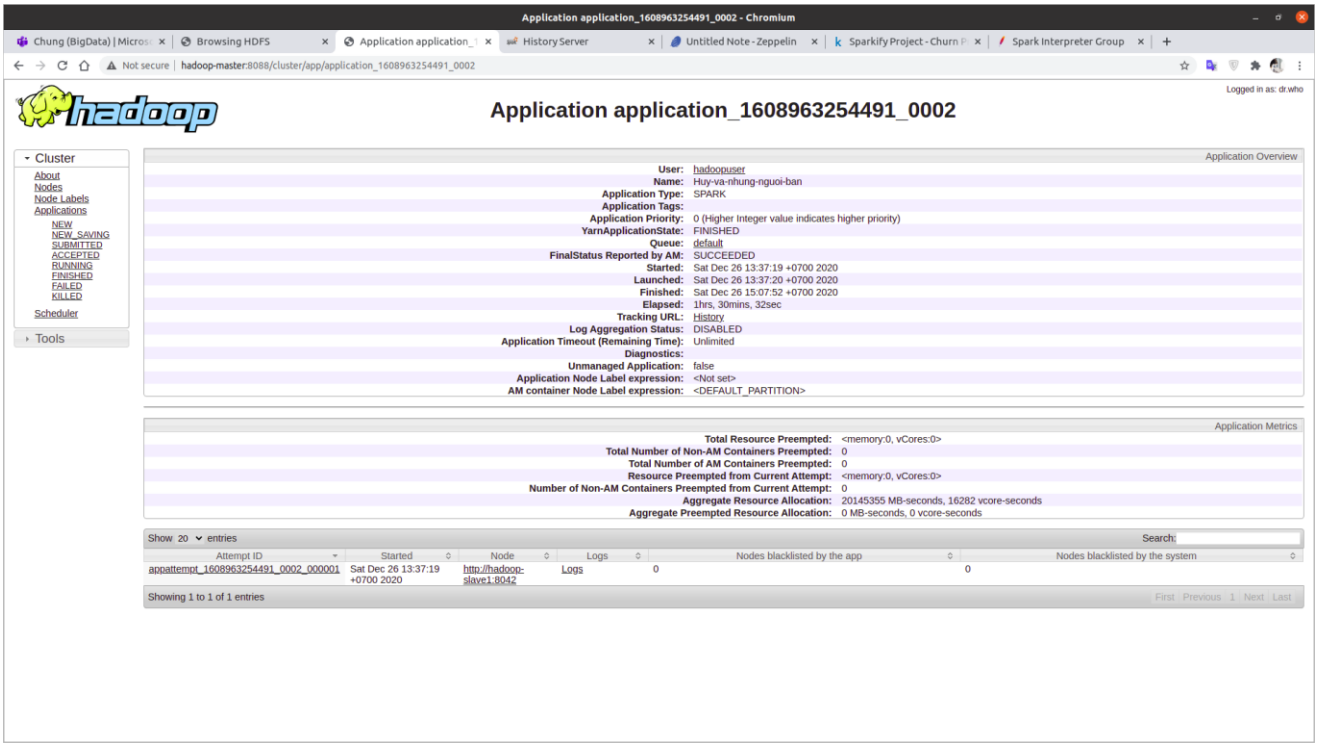

Home Page - Select or cr
IT4931-FinalProject-Giua
localhost:8888/notebooks/IT4931-FinalProject-GiuaChungTa-huymq1710.ipynb
jupyter IT4931-FinalProject-GiuaChungTa-huymq1710 Last Checkpoint: 18 hours ago (autosaved)
4 100010 F Chum
In [53]: sns.countplot(x="gender", hue="churn", data=pd_gender);
1e6
churn
Churn
Not Churn
count
gender
level
In [54]: pd_level = df_churn.select(["userId", "level", "churn"]).withColumn("churn", func_churn_label("churn")).toPandas()
pd_level.head()
Out[54]:
userId level churn
0 100010 free Chum
1 100010 free Chum
2 100010 free Chum
3 100010 free Chum
4 100010 free Chum
In [55]: sns.countplot(x="level", hue="churn", data=pd_level);
1e6
churn
Churn
Not Churn
count
free
paid
level


Home Page - Select or cr
IT4931-FinalProject-Giua
localhost:8888/notebooks/IT4931-FinalProject-GiuaChungTa-huymq1710.ipynb
jupyter IT4931-FinalProject-GiuaChungTa-huymq1710 Last Checkpoint: 18 hours ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Logout
Python 3
total song
In [56]:
pd_song = df_churn_user.join(df_churn.groupby("userId") \
.agg({"song": "count"}) \
.withColumnRenamed("count(song)", "song_count"), ["userId"]) \
.withColumn("churn", func_churn_label("churn")).toPandas()
pd_song.head()
Out[56]:
userId churn song_count
0 100010 Churn 480
1 200002 Churn 1550
2 296 Churn 672
3 125 Not Churn 368
4 124 Churn 10302
In [57]:
sns.boxplot(x="churn", y="song_count", data=pd_song);
Churn
Not Churn
More info: https://github.com/huymq1710
Install HDFS and Yarn multi node:
https://github.com/huymq1710/HDFS-MultiNode
https://medium.com/@jootorres_11979/how-to-set-up-a-hadoop-3-2-1-multi-node-cluster-on-ubuntu-18-04-2-nodes-567ca44a3b12
Install Spark cluster Yarn mode https://www.linode.com/docs/databases/hadoop/install-configure-run-spark-on-top-of-hadoop-yarn-cluster/
Install Zeppelin for Spark, set up for Spark Yarn mode https://medium.com/@shehanfernando/zeppelin-on-spark-cluster-6732d8eb1b59


Home Page - Select or cr
IT4931-FinalProject-Giua
localhost:8888/notebooks/IT4931-FinalProject-GiuaChungTa-huymq1710.ipynb
jupyter IT4931-FinalProject-GiuaChungTa-huymq1710 Last Checkpoint: 18 hours ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Logout
Python 3
In [57]:
sns.boxplot(x="churn", y="song_count", data=pd_song);
Churn
Not Churn
More info: https://github.com/huymq1710
Install HDFS and Yarn multi node:
https://github.com/huymq1710/HDFS-MultiNode
https://medium.com/@jootorres_11979/how-to-set-up-a-hadoop-3-2-1-multi-node-cluster-on-ubuntu-18-04-2-nodes-567ca44a3b12
Install Spark cluster Yarn mode https://www.linode.com/docs/databases/hadoop/install-configure-run-spark-on-top-of-hadoop-yarn-cluster/
Install Zeppelin for Spark, set up for Spark Yarn mode https://medium.com/@shehanfernando/zeppelin-on-spark-cluster-6732d8eb1b59
To Be Continued

4.4 Job done





V. Kết luận

Bằng việc sử dụng Hadoop, việc lưu trữ, quản lý được quản lý một cách dễ dàng đồng thời có tính chịu lỗi cao khi xày ra các trường hợp xấu như: node dead, lỗi mạng, out of storage,… Spark được xử lý và ghi trên Ram và được

Yarn quản lý tài nguyên, tăng tốc độ xử lý so với Hadoop . Zeppelin cung cấp một giao diện UI cho việc code và debug job phù hợp với việc báo cáo.

REFRENCES

[1] Install-Hadoop-and-Yarn:https://github.com/huymq1710/HDFS-MultiNode

[2] Install-Spark:https://www.linode.com/docs/guides/install-configure-run-spark-on-top-of-hadoop-yarn-cluster/

[3] Data and code: https://github.com/huymq1710/HDFS-MultiNode

[4] Hadoop: https://hadoop.apache.org/

[5] Spark: https://spark.apache.org/examples.html