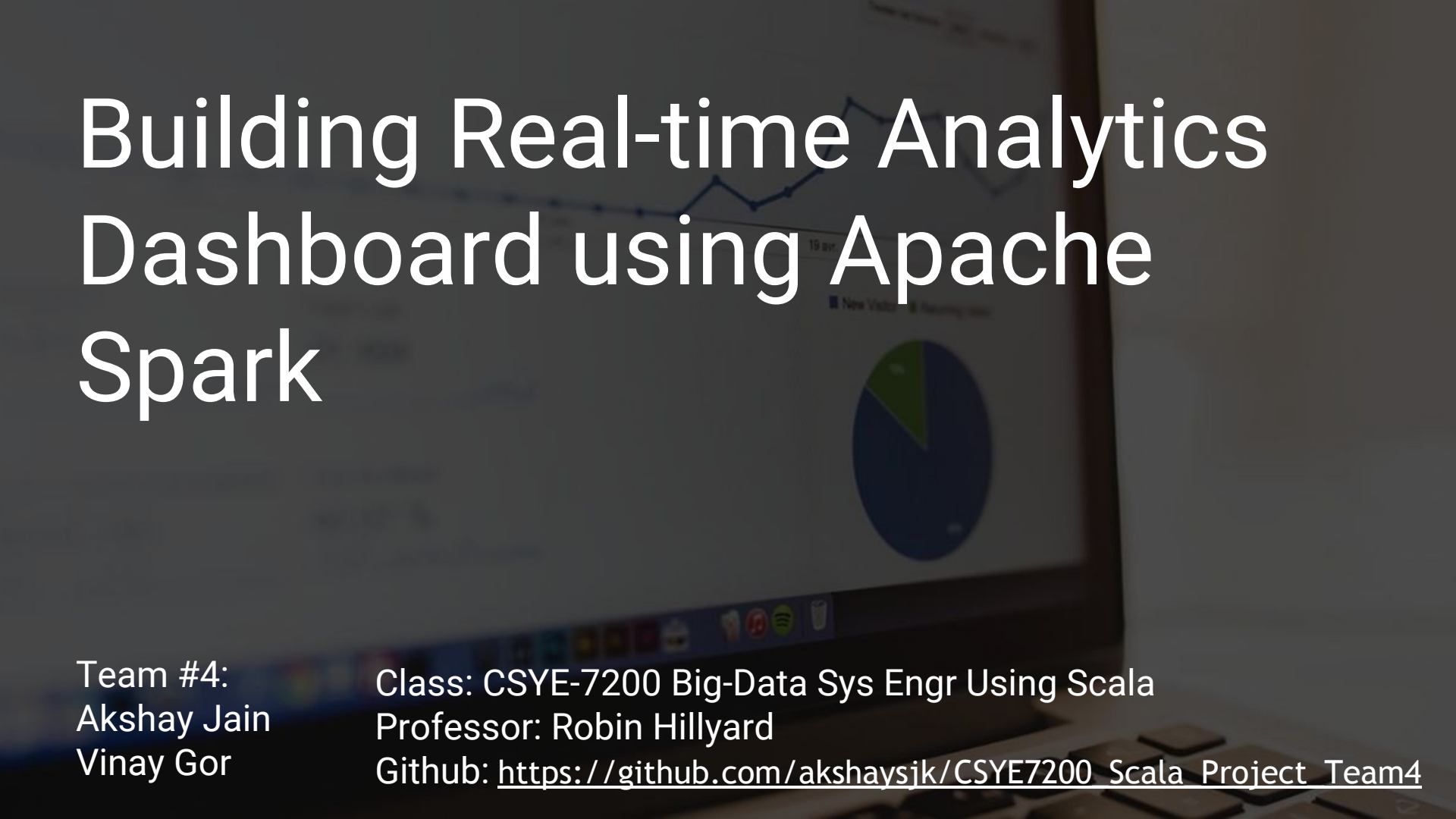


Building Real-time Analytics Dashboard using Apache Spark

The background image shows a laptop screen with a dashboard. The dashboard features a line chart at the top with a blue line and a pie chart below it. The pie chart is divided into two segments, one dark blue and one green. The laptop keyboard is visible at the bottom of the frame.

Team #4:
Akshay Jain
Vinay Gor

Class: CSYE-7200 Big-Data Sys Engr Using Scala
Professor: Robin Hillyard
Github: https://github.com/akshaysjk/CSYE7200_Scala_Project_Team4

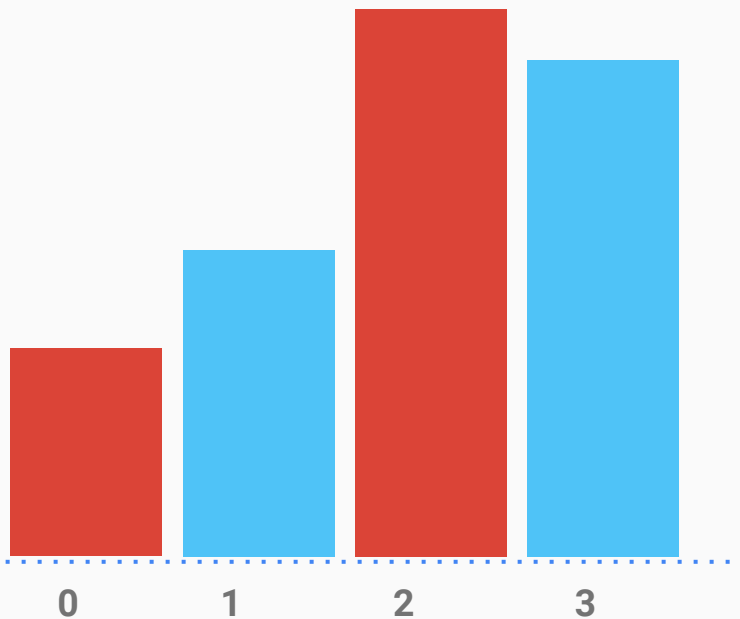
Goals of the Project

- To build a real-time analytics model using Stream Analytics.(Apache Spark)
- Reading of Data will be done in batches, to simulate real-time scenario.(Apache Kafka)
- Build a Real-time dashboard to display how the sales go on a particular day across different locations.
- Warehouse and inventory management at peak locations can be handled gracefully based on real-time analysis.

The problem

Batch Processing

The time delay between the collection of data and getting the result after the batch process.





The solution

Real-time Processing

Get the analytics in
real-time on Dashboard

Data Source

Data is taken from an ongoing competition on Analytics Vidhya website :
Practice Problem: Black Friday

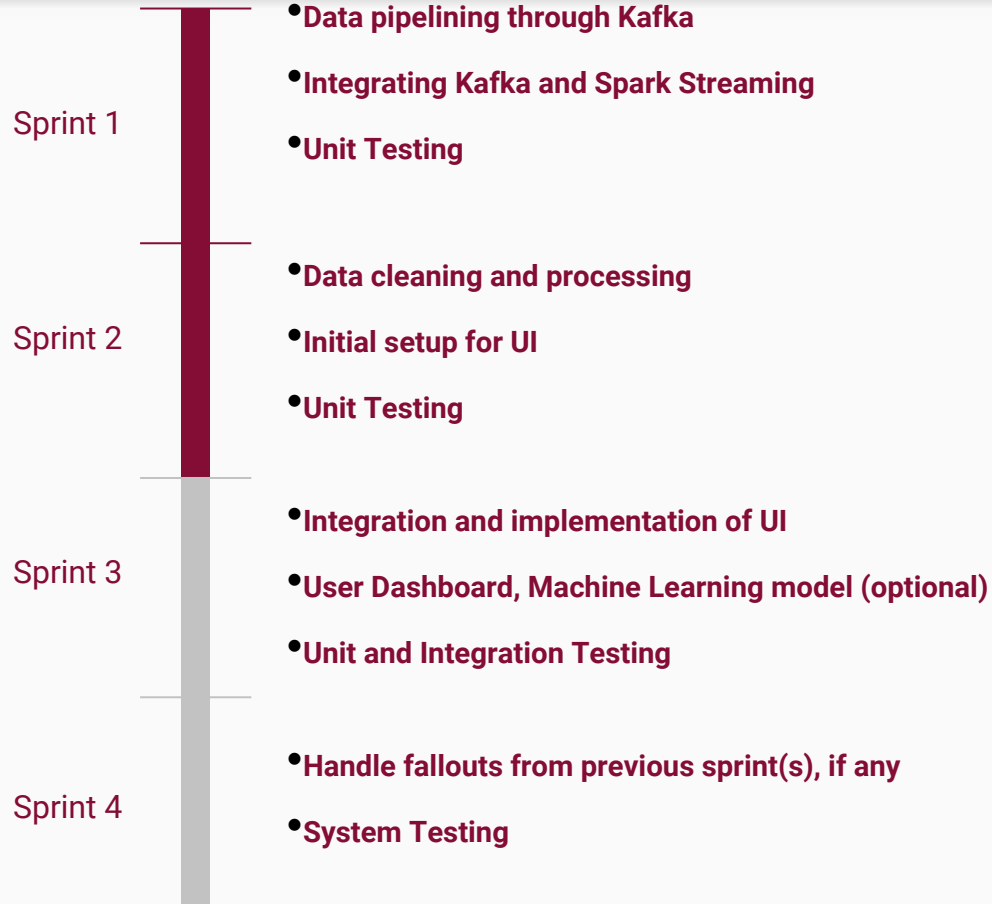
URL:

https://datahack.analyticsvidhya.com/contest/black-friday/#data_dictionary

Data :

train.csv : - consists of products and user details 0.5 million rows and 12 columns

Milestones / Sprints



Blockers faced

*Version compatibility with
Scala/Play/Spark*

Play Framework

Web Sockets

Methodology

Step 1

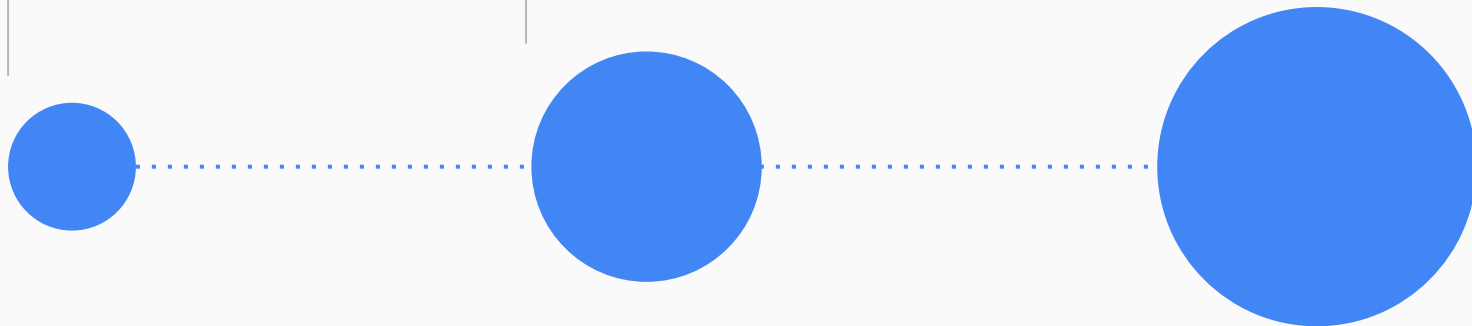
Read Data from CSV
through Apache Kafka

Step 2

Using Spark Streaming
to read data

Step 3

Clean the Data, run
analysis and pass data
to Dashboard through
Websockets



Technology Stack

Technologies:

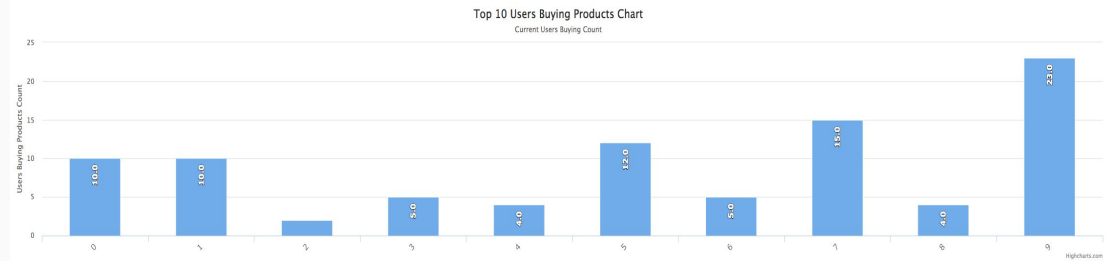
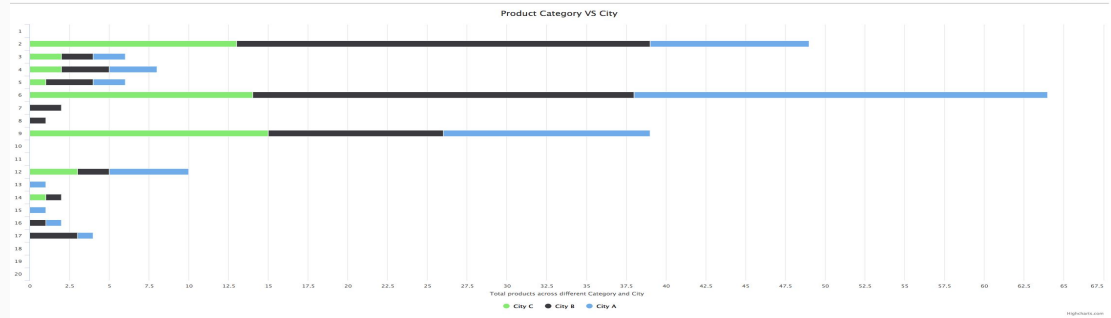
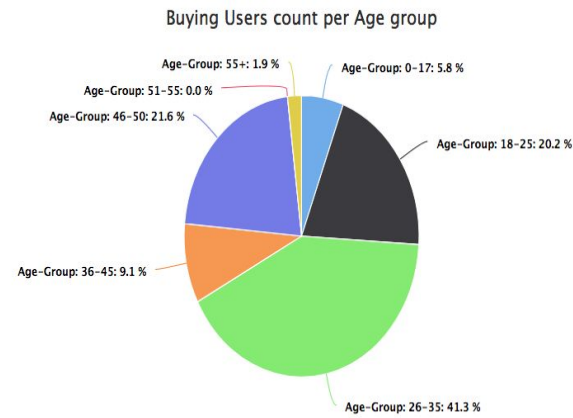
- Apache Kafka
- Spark Streaming
- Play Framework
- Web Socket Communication to pass data from Controller to Dashboard
- Highcharts.js for displaying charts
- Akka Actors for communication with Web Socket

Languages Used:

- Scala (66%)
- JavaScript (26%)
- HTML (6%)

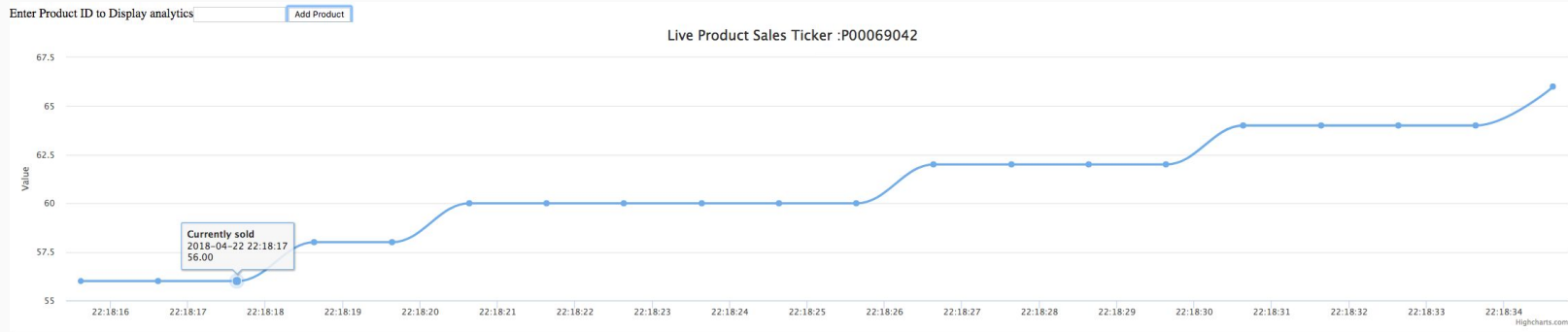
Use Case 1

Employee can see overview of the sales of products(eg: highest selling Product) on the dashboard homepage



Use case 2

Employee inputs query (such as product number) and gets the query specific real-time values.



Acceptance criteria

- 85% of the time, Spark Streaming will clean the data received, process it and generate/update the dashboard within 10 sec

$1524356551266 - 1524356550685 = 605 \text{ Milliseconds}$
~ 0.6 seconds

$1524356552174 - 1524356550728 = 1446$
~ 1.5 seconds

Time range ~ 0.6 to 5.3

Criteria Met!

```
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
Streaming Data Started!
[info] application - Received a message
[info] application - Logging input : ProductID P00102542 at time :1524356550661
[info] application - Logging input : ProductID P00273442 at time :1524356550685
[info] application - Logging input : ProductID P00281542 at time :1524356550686
[info] application - Logging input : ProductID P00367542 at time :1524356550687
[info] application - Logging input : ProductID P00253042 at time :1524356550688
```

Developer Tools - http://localhost:9000/startStreaming

Elements Console Sources Network Performance Memory Application

top Filter Default levels ▾ Gr

I am here

Logging result :=> P00273442 at time: 1524356551266

Logging result :=> P00367542 at time: 1524356551363

Logging result :=> P00273442 at time: 1524356551518

```
[info] application - Logging input : ProductID P00000142 at time :1524356550723
[info] application - Logging input : ProductID P00284642 at time :1524356550727
[info] application - Logging input : ProductID P00313342 at time :1524356550728
[info] application - Logging input : ProductID P00288342 at time :1524356550732
```

Logging result :=> P00273442 at time: 1524356552055

Logging result :=> P00313342 at time: 1524356552174

Logging result :=> P00313342 at time: 1524356552254

Future Scope

- Recommendation of product to Users with the help of Machine learning Algorithm
- User's Dashboard to display Analytics of the products purchased

Thank You!