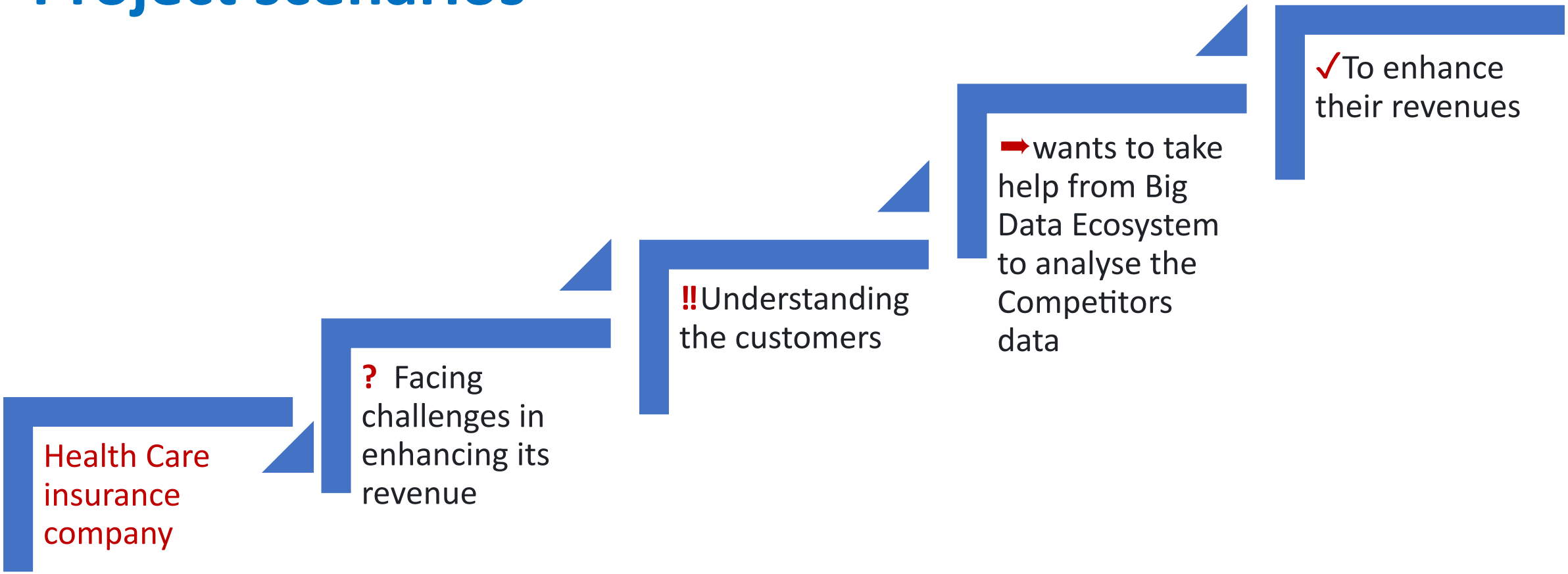
Capstone Project

Pabitra Aryal

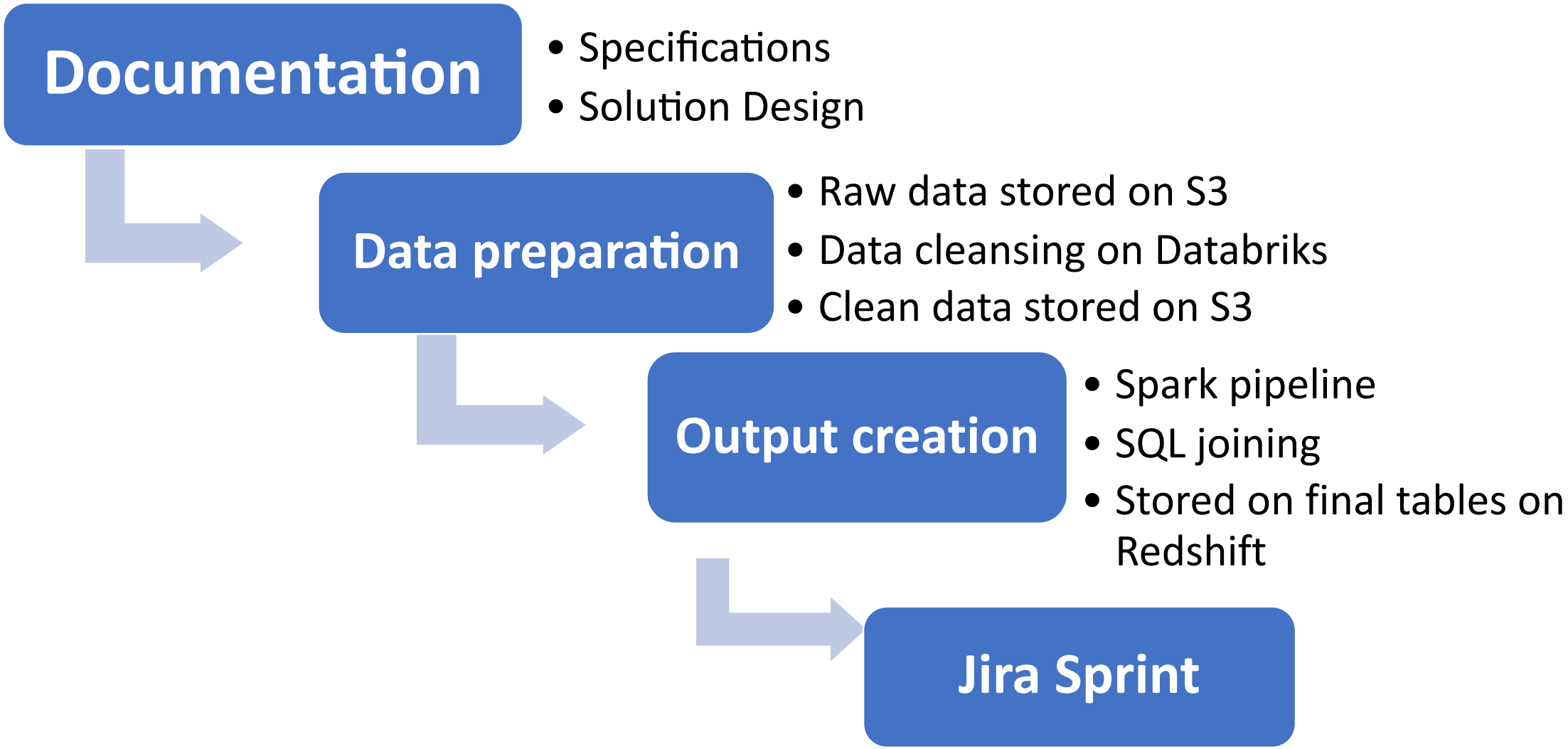
Takeo Bootcamp, BDE37

Date: 20 Dec 2023

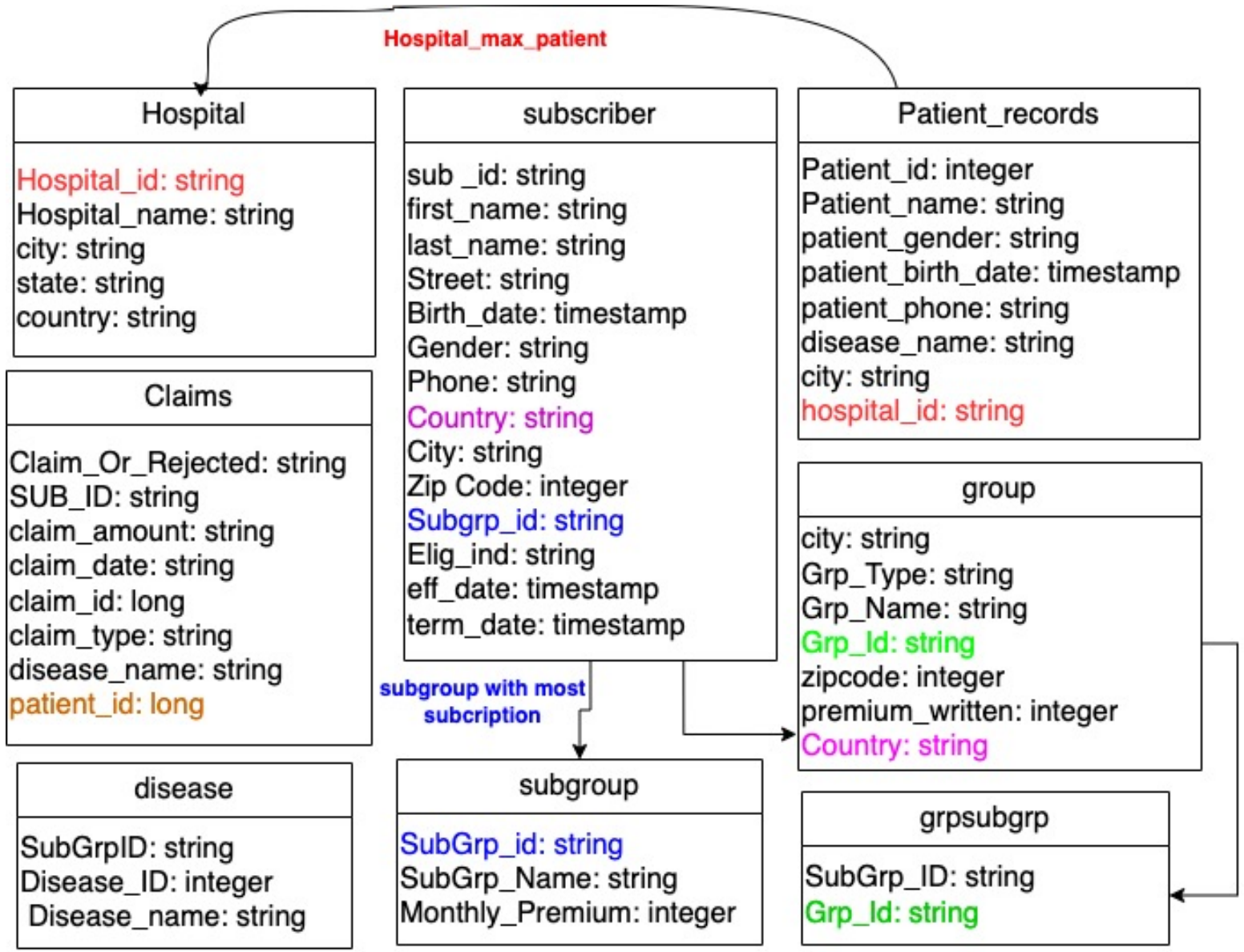
Project scenarios



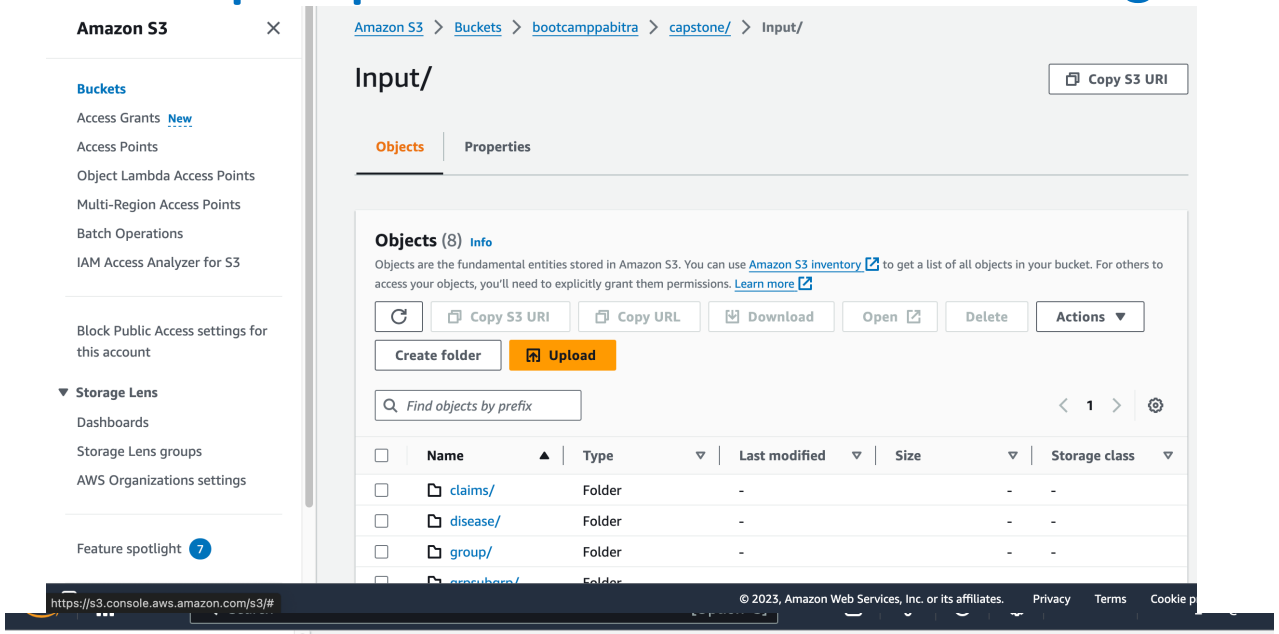
Steps follwed for the final results

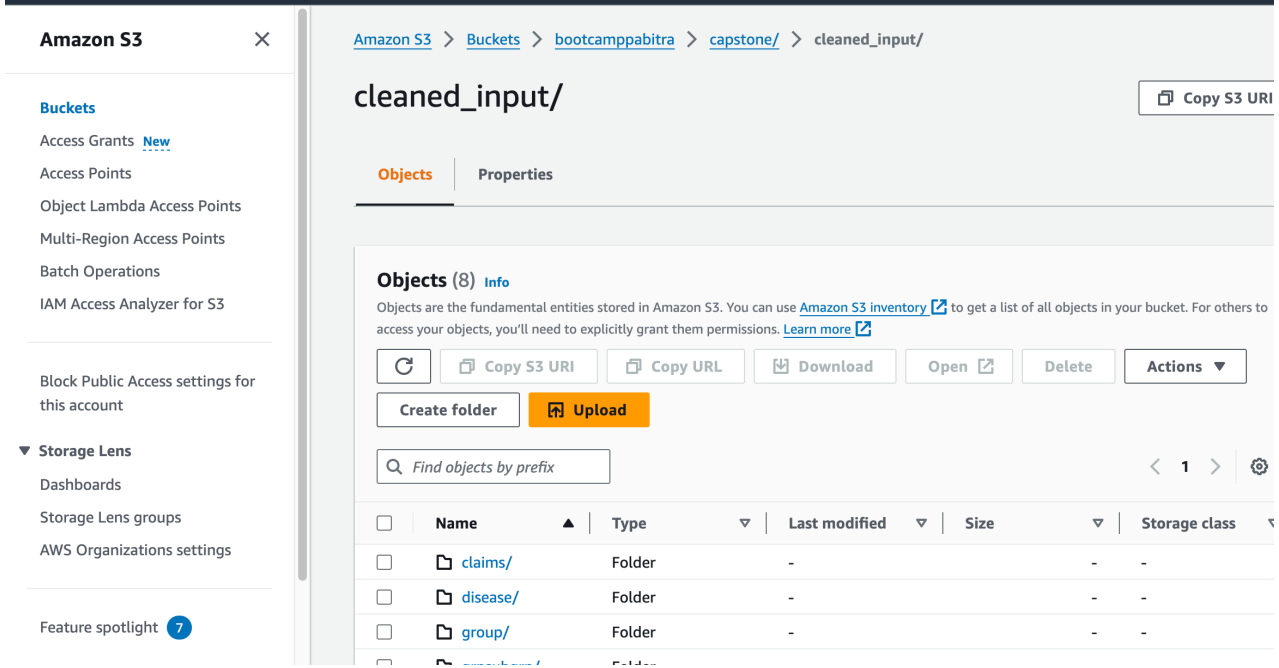


ER diagram


Hospital_max_patient
Hospital
Hospital_id: string
Hospital_name: string
city: string
state: string
country: string
subscriber
sub _id: string
first_name: string
last_name: string
Street: string
Birth_date: timestamp
Gender: string
Phone: string
Country: string
City: string
Zip Code: integer
Subgrp_id: string
Elig_ind: string
eff_date: timestamp
term_date: timestamp
Patient_records
Patient_id: integer
Patient_name: string
patient_gender: string
patient_birth_date: timestamp
patient_phone: string
disease_name: string
city: string
hospital_id: string
Claims
Claim_Or_Rejected: string
SUB_ID: string
claim_amount: string
claim_date: string
claim_id: long
claim_type: string
disease_name: string
patient_id: long
subgroup with most subcription
group
city: string
Grp_Type: string
Grp_Name: string
Grp_Id: string
zipcode: integer
premium_written: integer
Country: string
disease
SubGrpID: string
Disease_ID: integer
Disease_name: string
subgroup
SubGrp_id: string
SubGrp_Name: string
Monthly_Premium: integer
grpsubgrp
SubGrp_ID: string
Grp_Id: string

Data preparation : *data cleansing*





databricks

capstone_data_cleaning (Python)

```
#### Count string, timestamp, and integer null values
from pyspark.sql import functions as F
df.select(*[
    (
        F.count(F.when((F.isnan(c) | F.col(c).isNull()), c)) if t not in ("timestamp", "date")
        else F.count(F.when(F.col(c).isNull(), c))
    ).alias(c)
    for c, t in df.dtypes if c in df.columns
]).show()
```

```
+----------+------------+--------------+------------------+-------------+------------+----+-----------+
|Patient_id|Patient_name|patient_gender|patient_birth_date|patient_phone|disease_name|city|hospital_id|
+----------+------------+--------------+------------------+-------------+------------+----+-----------+
|         0|          17|             0|                 0|            2|           0|   0|          0|
+----------+------------+--------------+------------------+-------------+------------+----+-----------+
```

```
df = df.na.fill('NaN', subset=['Patient_name']) \
    .na.fill('NaN', subset=['patient_phone'])
df.show()
```

```
+----------+------------+--------------+------------------+-------------+------------+--------------------+-----------+
|Patient_id|Patient_name|patient_gender|patient_birth_date| patient_phone|  disease_name|                city|hospital_id|
+----------+------------+--------------+------------------+-------------+------------+--------------------+-----------+
|    187158|      Harbir|        Female|        1924-06-30|+91 0112009318|  Galactosemia|            Rourkela|      H1001|
|    112766|    Brahmdev|        Female|        1948-12-20|+91 1727749552|Bladder cancer|         Tiruvottiyur|      H1016|
```

Use case solution

databricks

Use_Case_solutions (Python)

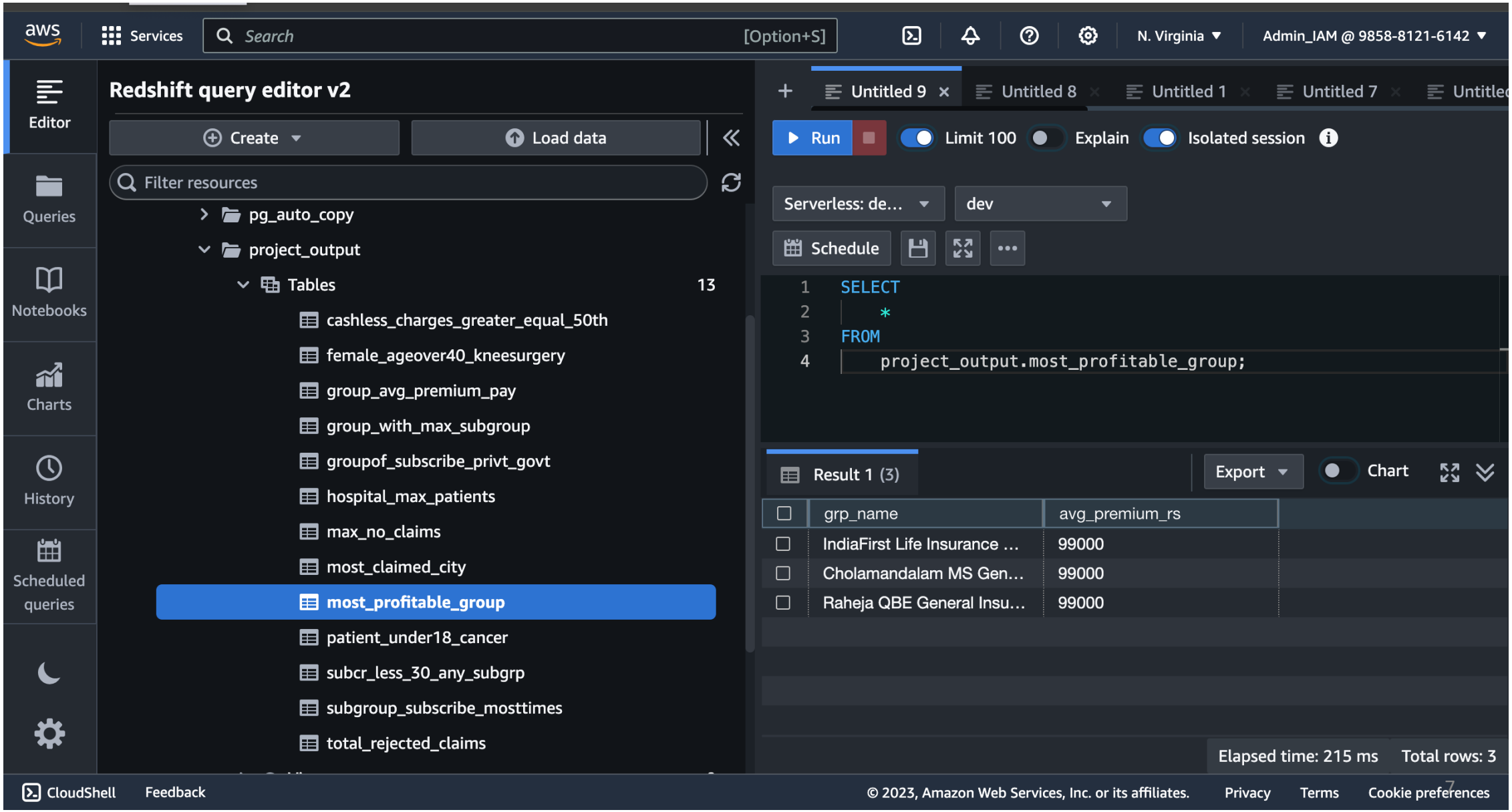
Import Notebook

```
df.createOrReplaceTempView("patient_records")
df3=spark.sql("with T as (select count(c.claim_id) cnt, p.city from claims as c inner join patient_records as p on c.patient_id=p.Patient_id group by p.
city) select city, cnt as total_count from T where cnt=(select max(cnt) from T)")
df3.show()
```

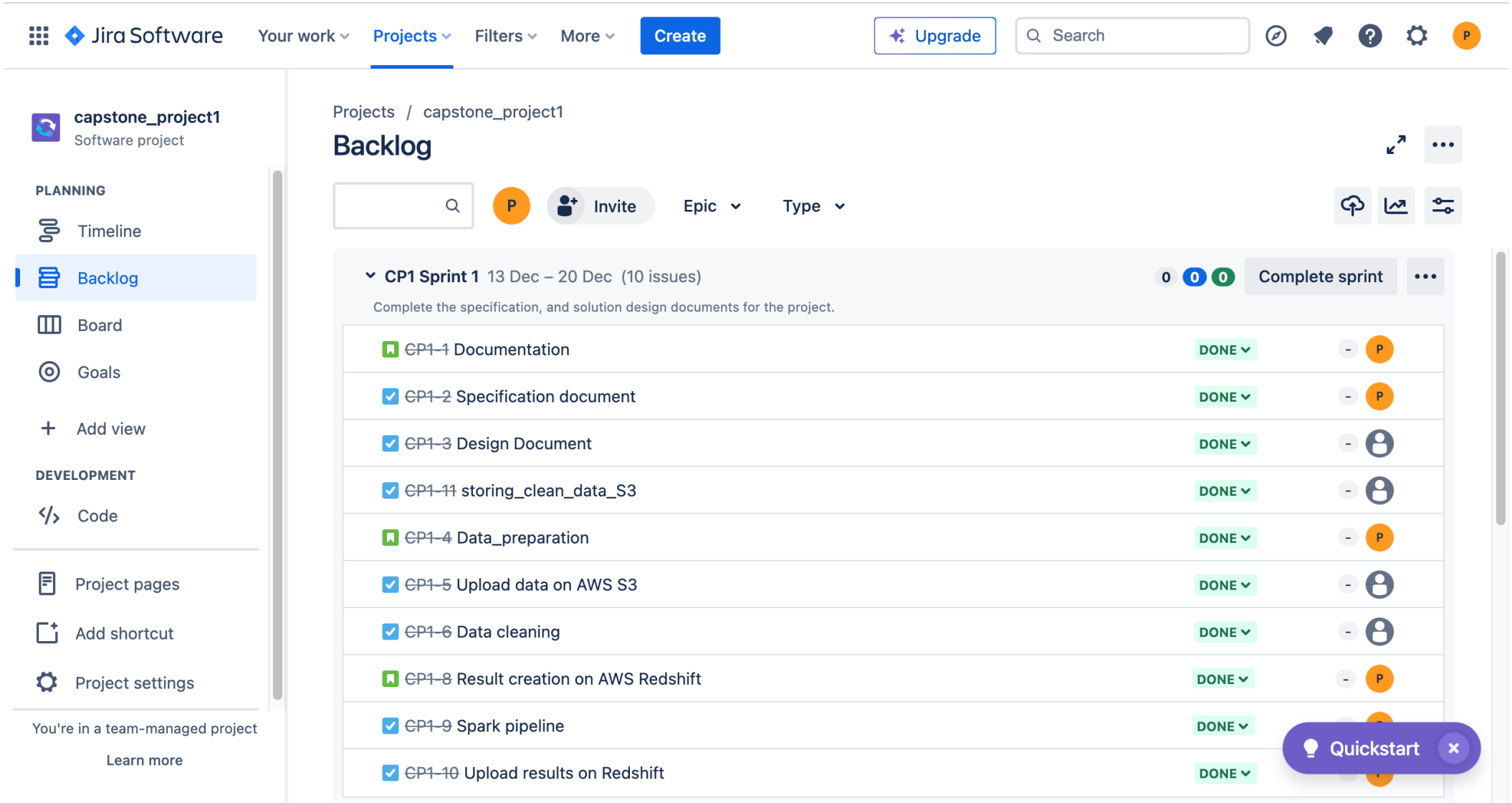
```
+------------+-----------+
|        city|total_count|
+------------+-----------+
|      Mysore|          2|
|    Amravati|          2|
|   Kamarhati|          2|
|    Jabalpur|          2|
|Bihar Sharif|          2|
|   Ghaziabad|          2|
|       Morbi|          2|
|  Karimnagar|          2|
+------------+-----------+
```

```
######################## Use case 7 ########################## uploading to Redshift #############
df3.write.format("redshift").option("url","jdbc:redshift://default-workgroup.985881216142.us-east-1.redshift-serverless.amazonaws.com:5439/dev").\
    option("dbtable", "project_output.most_claimed_city").\
    option("aws_iam_role", "*****************").\
    option("driver","com.amazon.redshift.jdbc42.Driver").\
    option("tempdir", "s3a://pabitrabucket/tmpdir/").\
    option("user", "*****************").\
    option("password", "*****************").save()
```

Output creation: *final tables on Redshift*



Jira Sprint


Jira Software
Your work
Projects
Filters
More
Create
Upgrade
Search
capstone_project1
Software project
PLANNING
Timeline
Backlog
Board
Goals
Add view
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Projects / capstone_project1
Backlog
Invite
Epic
Type
CP1 Sprint 1 13 Dec – 20 Dec (10 issues)
Complete sprint
Complete the specification, and solution design documents for the project.
CP1-1 Documentation DONE
CP1-2 Specification document DONE
CP1-3 Design Document DONE
CP1-11 storing_clean_data_S3 DONE
CP1-4 Data_preparation DONE
CP1-5 Upload data on AWS S3 DONE
CP1-6 Data cleaning DONE
CP1-8 Result creation on AWS Redshift DONE
CP1-9 Spark pipeline DONE
CP1-10 Upload results on Redshift DONE
Quickstart

Thank you for listening!