Travel System Solution Description

I. Explain the solution

1. Assumption
   - Any record with touch OFF status without touch ON before will be an unprocessable record
   - Any record that has data cell value null/empty will be unprocessable
   - A row of data of touch will be called a "touch record"
   - TouchOn and TouchOff in a trip will be same companyId and busId
   - Bus fee mean a trip cost
2. Pre-condition

- Define a Cost mapping, to fast lookup cost of TripX_To_TripY

Ex:

"StopA_TO_StopB": 4.50

"StopB_TO_StopC": 6.25

"StopA_TO_StopC": 8.45

"StopB_TO_StopA": 4.50

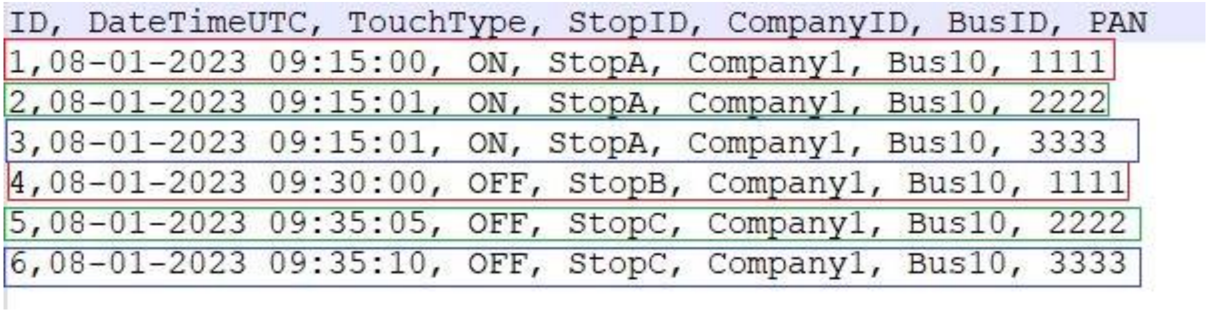
"StopC_TO_StopB": 6.25

"StopC_TO_StopA": 8.45

- Make a group of touching of each PAN (sorted by 'DateTimeUTC') from csv input file

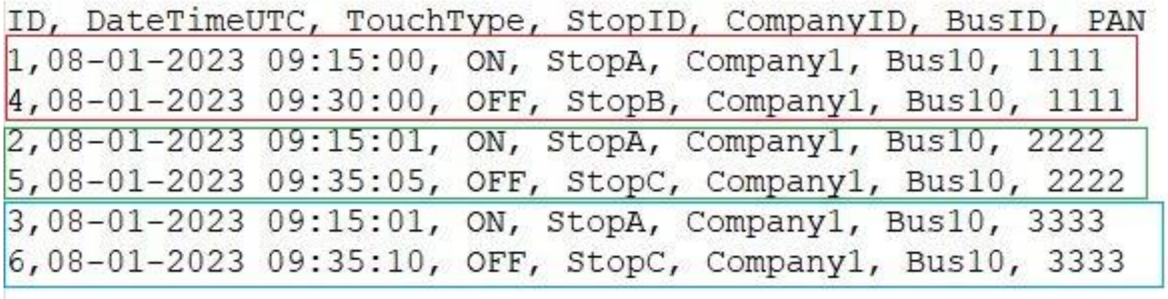
Ex:

Input:



```
ID, DateTimeUTC, TouchType, StopID, CompanyID, BusID, PAN
1,08-01-2023 09:15:00, ON, StopA, Company1, Bus10, 1111
2,08-01-2023 09:15:01, ON, StopA, Company1, Bus10, 2222
3,08-01-2023 09:15:01, ON, StopA, Company1, Bus10, 3333
4,08-01-2023 09:30:00, OFF, StopB, Company1, Bus10, 1111
5,08-01-2023 09:35:05, OFF, StopC, Company1, Bus10, 2222
6,08-01-2023 09:35:10, OFF, StopC, Company1, Bus10, 3333
```

After grouping input and Sort by 'DateTimeUTC'



```
ID, DateTimeUTC, TouchType, StopID, CompanyID, BusID, PAN
1,08-01-2023 09:15:00, ON, StopA, Company1, Bus10, 1111
4,08-01-2023 09:30:00, OFF, StopB, Company1, Bus10, 1111
2,08-01-2023 09:15:01, ON, StopA, Company1, Bus10, 2222
5,08-01-2023 09:35:05, OFF, StopC, Company1, Bus10, 2222
3,08-01-2023 09:15:01, ON, StopA, Company1, Bus10, 3333
6,08-01-2023 09:35:10, OFF, StopC, Company1, Bus10, 3333
```

Follow that order, we can assume that the “touch ON” record is always before the “touch OFF” record, and two “touch ON” – “touch OFF” records in a PAN group are adjoining together in a trip.

3. Bus fee calculation rule

- Prepare 3 lists to store information of “trips”, “unprocessableTouchData”, “summary”

- Touch record has any null/empty fields value will be added to “unprocessableTouchData”

- Calculate the bus fee for each PAN based on the list records touch ON/OFF

Rule are:

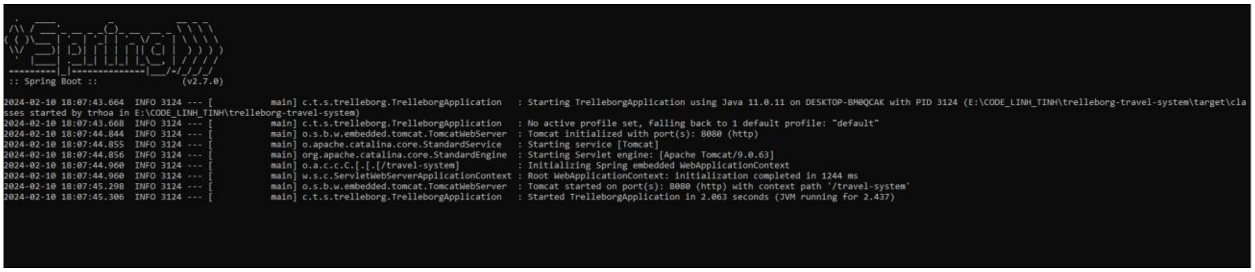
1. If A touchOnRecord has touchOffRecord (next to touchOnRecord)
   1.1 Build key to look up on “cost mapping” via pattern *<touchOnRecord.stopID_TO_ touchOffRecord.stopID>*
   Ex: “StopA_TO_StopB” = 4.50
   1.2 Build trip data base on touchOnRecord and touchOffRecord information.
   1.3 Add trip information to the “trips” list (COMPLETE status)
2. If next to touchOnRecord is not touchOffRecord.
   2.1 Lookup in “cost mapping” any key containt “touchOnRecord.stopID”, then find max cost of those map item.
   2.2 Build trip data base on touchOnRecord and maximum cost of “touchOnRecord.stopID”
   2.3 Add trip information to the “trips” list (INCOMPLETE status)
3. If touchOnRecord has the TouchType = OFF
   3.1 This touchOffRecord without touchOnRecord, shoule be an unprocessable trip, build a trip data base on touchOffRecord and set trip status “NO TOUCH ON DATA”
   3.2 Add this trip into “unprocessableTouchData” list
4. Base on “trips”, calculate that data to “summary”
5. Generate results of “trips”, “unprocessableTouchData”, “summary”

II. Run and testing

1. Test cases resource stored in “trelleborg-travel-system\src\main\resources\csv_test_data”
   - *case 1: "happycase_enought_data_on_off_samebus.csv"*
     Happy cases with 6 valid touch pairs (3 touch on/ 3 touch off)
   - *case 2: "one_incompleted_and_one_unprocessable.csv"*
     8 touch record with 6 valid touch on/off (complete), 1 touch on without touch off (incompleted), 1 touch off without touch on (unprocessable)
   - *case 3: "complext_cases.csv"*
     10 touch record with:
     3 empty/null fields, 1 no touch on (4 unprocessable)
     3 completed, 2 incomplete, 1 canceled (6 trips)
   - *case 4: "10_valid_touch_3_days.csv"*
     18 valid touch records in 3 days (9 complete)
2. Run application:
   Go to: /trelleborg-travel-system

run: mvn clean package

run: mvn spring-boot:run

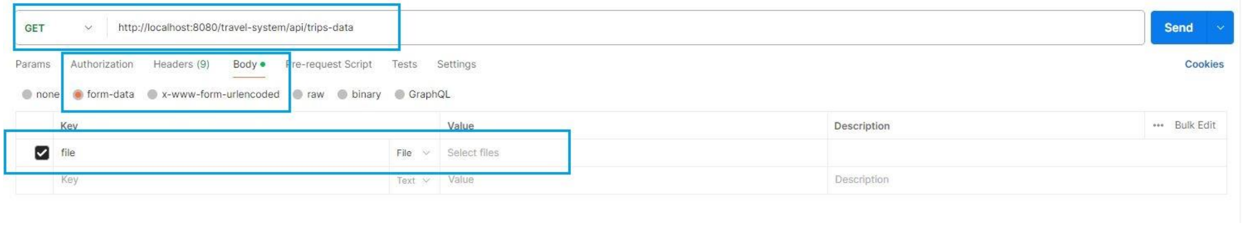


```
  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::                (v2.7.0)

2024-02-10 18:07:43.664  INFO 3124 --- [           main] c.t.s.trelleborg.TrelleborgApplication   : Starting TrelleborgApplication using Java 11.0.11 on DESKTOP-BM0QCAK with PID 3124 (E:\CODE_LINH_TINH\trelleborg-travel-system\target\cla
sses started by trhoa in E:\CODE_LINH_TINH\trelleborg-travel-system)
2024-02-10 18:07:43.668  INFO 3124 --- [           main] c.t.s.trelleborg.TrelleborgApplication   : No active profile set, falling back to 1 default profile: "default"
2024-02-10 18:07:44.844  INFO 3124 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port(s): 8080 (http)
2024-02-10 18:07:44.855  INFO 3124 --- [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2024-02-10 18:07:44.856  INFO 3124 --- [           main] org.apache.catalina.core.StandardEngine  : Starting Servlet engine: [Apache Tomcat/9.0.63]
2024-02-10 18:07:44.960  INFO 3124 --- [           main] o.a.c.c.C.[.[.[/travel-system]           : Initializing Spring embedded WebApplicationContext
2024-02-10 18:07:44.960  INFO 3124 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1244 ms
2024-02-10 18:07:45.298  INFO 3124 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8080 (http) with context path '/travel-system'
2024-02-10 18:07:45.306  INFO 3124 --- [           main] c.t.s.trelleborg.TrelleborgApplication   : Started TrelleborgApplication in 2.063 seconds (JVM running for 2.437)
```

3. Import postman Curl:

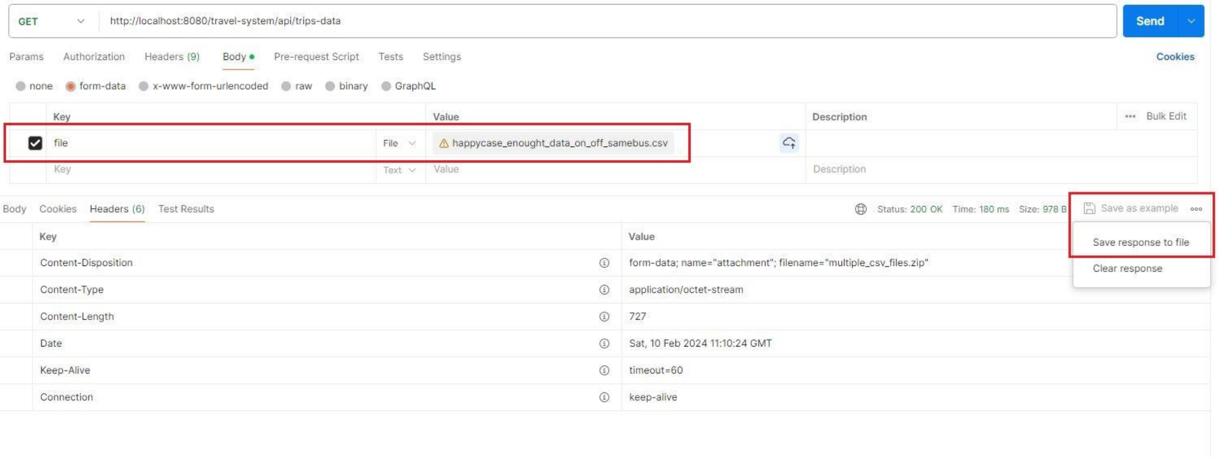
```
curl --location --request GET 'http://localhost:8080/travel-system/api/trips-data' \
--form 'file=@"/path/to/file"'
```



GET http://localhost:8080/travel-system/api/trips-data

	Key		Value	Description
☑	file	File	Select files	
	Key	Text	Value	Description

4. Select an input file then run



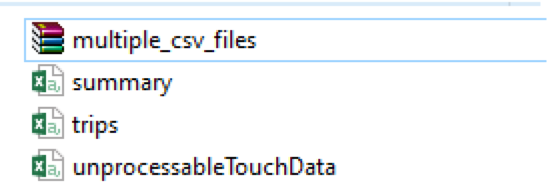
GET http://localhost:8080/travel-system/api/trips-data

	Key		Value	Description
☑	file	File	⚠ happycase_enought_data_on_off_samebus.csv	
	Key	Text	Value	Description

Status: 200 OK   Time: 180 ms   Size: 978 B

Key	Value
Content-Disposition	form-data; name="attachment"; filename="multiple_csv_files.zip"
Content-Type	application/octet-stream
Content-Length	727
Date	Sat, 10 Feb 2024 11:10:24 GMT
Keep-Alive	timeout=60
Connection	keep-alive

Select “Save response to file”, we will get the zip named “multiple_csv_files”, extract and see 3 expected output files “trips.csv”, “unprocessableTouchData.csv”,”summary.csv”



multiple_csv_files
summary
trips
unprocessableTouchData

Note:

Enhancements:

- Build a mapping for each Stop station with maximum cost, to improve performance process.

- Unit testing controller and services.

Thanks for your time to read that solution document!