MASTER DEGREE PROGRAM IN
DATA SCIENCE AND ADVANCED ANALYTICS

Game Day

The role of analytics in sport organizations to help increase revenues and create a better fan experience on matchdays

BY SIMÃO EDUARDO RAMOS COSTA PEREIRA

Supervisor: Miguel de Castro Simões Ferreira Neto
Co-Supervisor: João Bruno Morais de Sousa Jardim
BI4All Mentor: Rui Valente

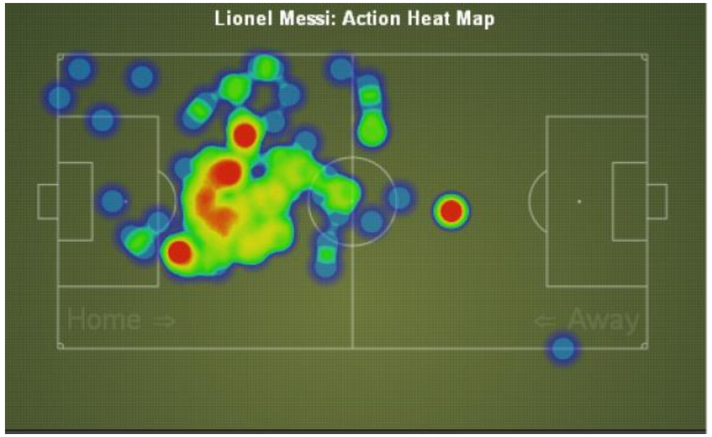
NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

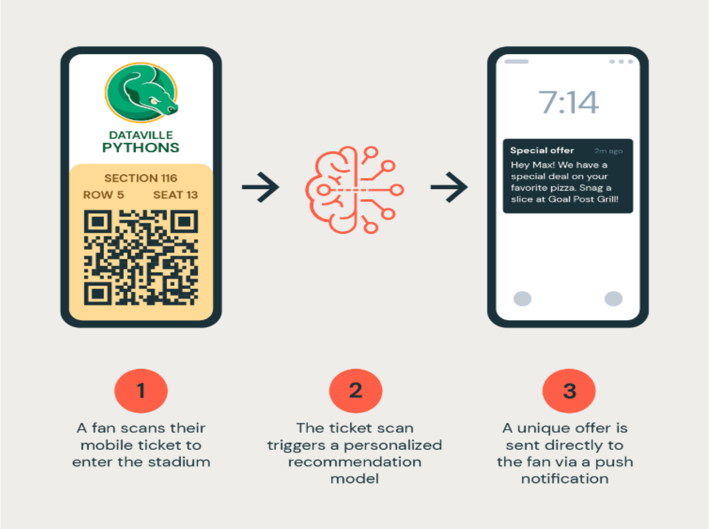
AGENDA

1. Introduction
2. Objectives
3. Literature Review
4. Methodology
5. Results
6. Conclusion
7. Future Work
8. Discussion

INTRODUCTION

- **Sports beyond performance**: Modern sports organizations are major economic and financial entities with necessity to remain competitive in and outside the field.
- **Role of analytics and big data in sports**: Essential in measuring and improving athlete performance but also crucial in harnessing data from fans and stakeholders.
- **Emergence of data departments**: Sports clubs forming analytics/data science teams with the goal of increasing revenues, reduce costs, enhance fan experiences.
- **Focus on fans**: Recognizing fans as the primary revenue source. Enhancing fan experience and value for sustained growth.





Special offer 2m ago
Hey Max! We have a special deal on your favorite pizza. Snag a slice at Goal Post Grill!

1. A fan scans their mobile ticket to enter the stadium
2. The ticket scan triggers a personalized recommendation model
3. A unique offer is sent directly to the fan via a push notification

OBJECTIVES

Exploration and innovation

Showcasing how analytics and big data can have a transformative role in the sports industry and how many functionalities one can deploy in this field.

Enhancement focus

Specifically aimed at improving fan experience and boosting revenue for sports organizations.

Real-world relevance

Build a BI solution in real world use cases capable of showcasing how data and analytics can be leveraged to benefit both organizations and fans

LITERATURE REVIEW

Analytical Applications in Sports

Emerging Trends: Shift towards more extensive use of analytics for business and fan engagement (Ratten, V., Dickson, G., 2020).

- Fan behaviour analysis - Social Media Analysis: Use of Hootsuite, Sprout Social, Brandwatch for fan sentiment analysis (Gong, X., Wang, Y., 2021).

- Food, Beverage, and Merchandising Analytics - Revenue Enhancement Strategies: Research by Sutton, W. A., et al. (2020) on the impact of analytics in these domains. Sutton, W. A., et al. (2020)

- Fan Engagement Analytics - Technological Influence: Studies by Caulfield, J., Jha, A. K. (2022) on technology's role in enhancing fan involvement and spending.

- Cloud Applications in Sports- Case Study: NBA's use of Azure for fan experience (Microsoft., 2021).

Technological tools for used for analytics in sports

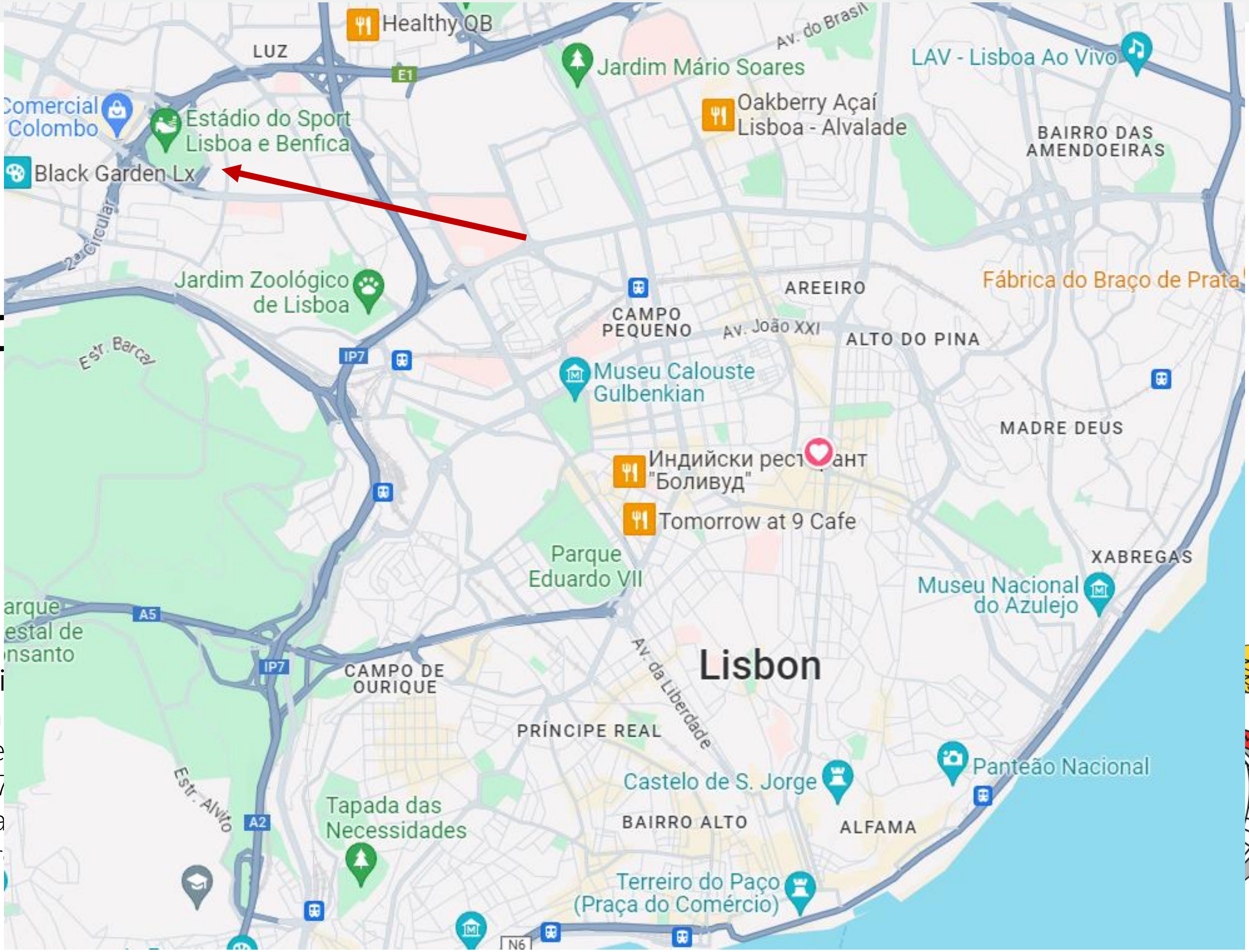
- CRM and BI Tools -Role of Salesforce, Microsoft Dynamics, SAP (Apostolou, K., Tjortjis, C. 2019), Tableau, and Power BI in data analysis (Venugopal, S., 2023).

- Social Media Analysis -Use of Hootsuite, Sprout Social, Brandwatch for fan sentiment analysis (Gong, X., Wang, Y., 2021

- Cloud capabilities -(Bouquet, P., Molinari, A., Bortolan, L., Dal Pra, A., Bezzi, G., & Vettorato, S., 2022).

METHODOLOGY

USE C



Considerati

- In both matche 38.7527
- The ma softwar

USE CASE 1: REAL TIME ACCESS TO A DATA WAREHOUSE AND TAKING LIVE ACTIONS

LIVE ACTIONS

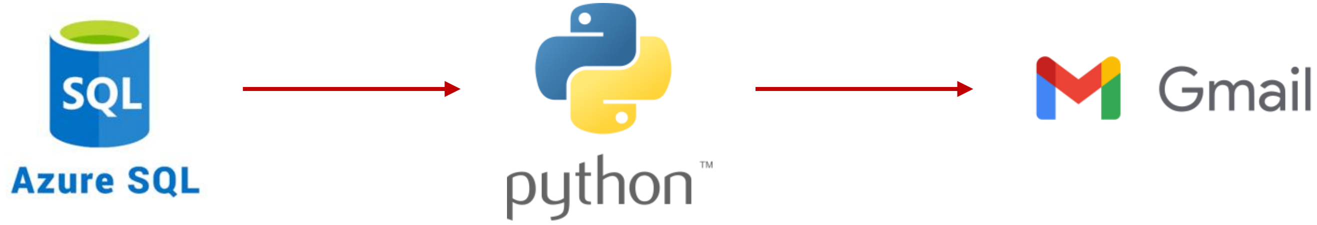
Before the match

if the club's database indicates that a specific fan does not possess a ticket five hours before the match, an email can be sent as a reminder to purchase tickets.

At halftime

If the club's database confirms that a specific fan holds a ticket for the game, a notification can be sent at halftime featuring promotional offers.

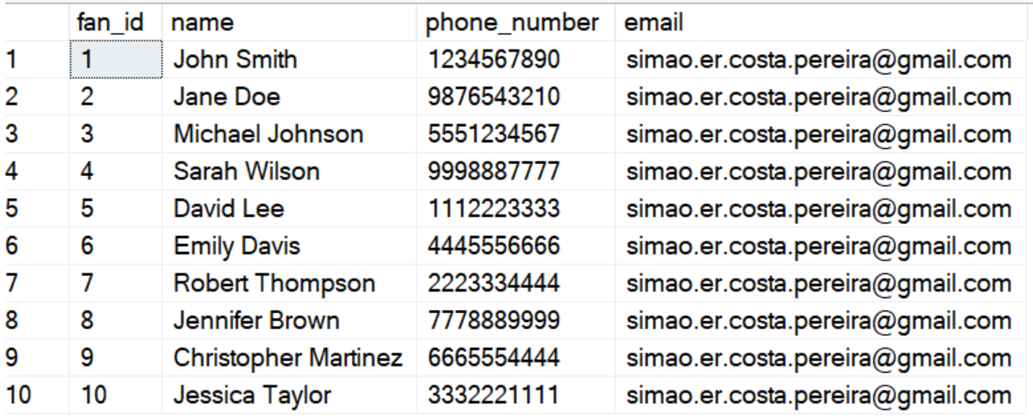
How:



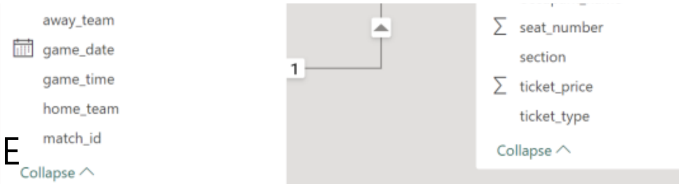
To deploy these actions, Python was used to connect to Azure SQL Database and leverage the data warehouse, enabling the sending of personalized emails.

USE CASE 1: REAL TIME ACCESS TO A DATA WAREHOUSE AND TAKING LIVE ACTIONS

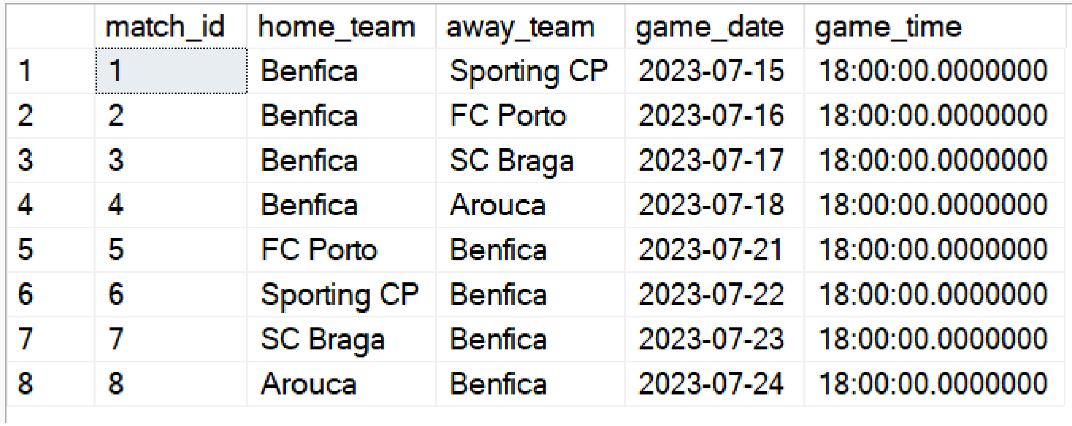
FANS SQL TABLE



	fan_id	name	phone_number	email
1	1	John Smith	1234567890	simao.er.costa.pereira@gmail.com
2	2	Jane Doe	9876543210	simao.er.costa.pereira@gmail.com
3	3	Michael Johnson	5551234567	simao.er.costa.pereira@gmail.com
4	4	Sarah Wilson	9998887777	simao.er.costa.pereira@gmail.com
5	5	David Lee	1112223333	simao.er.costa.pereira@gmail.com
6	6	Emily Davis	4445556666	simao.er.costa.pereira@gmail.com
7	7	Robert Thompson	2223334444	simao.er.costa.pereira@gmail.com
8	8	Jennifer Brown	7778889999	simao.er.costa.pereira@gmail.com
9	9	Christopher Martinez	6665554444	simao.er.costa.pereira@gmail.com
10	10	Jessica Taylor	3332221111	simao.er.costa.pereira@gmail.com

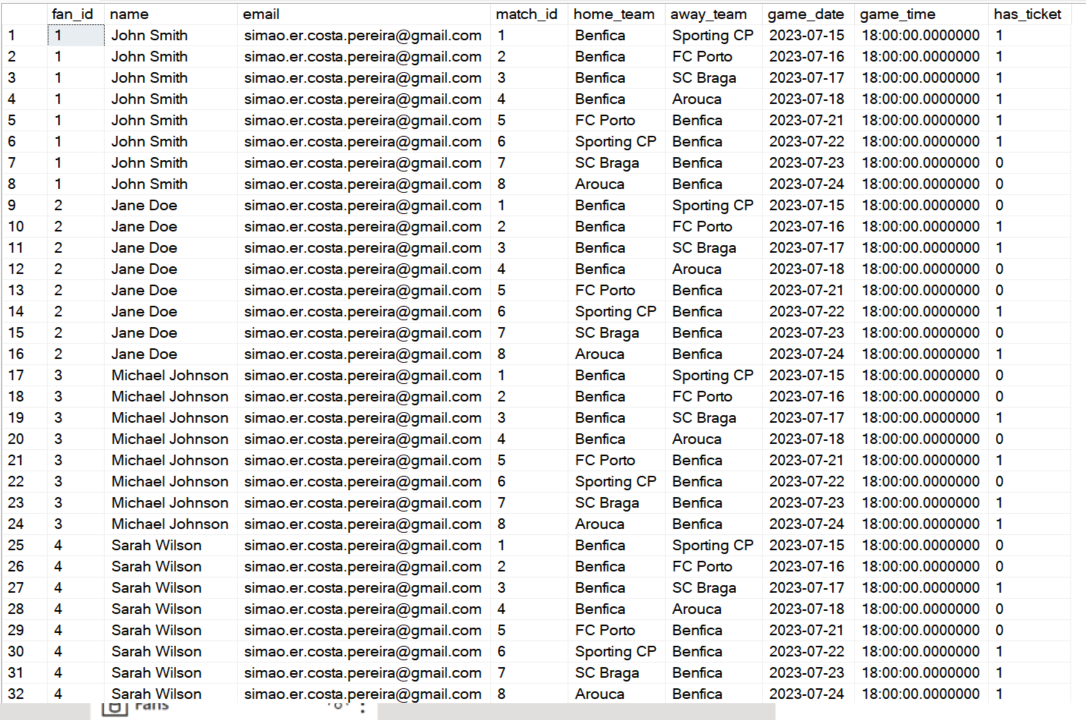


MATCHES SQL TABLE



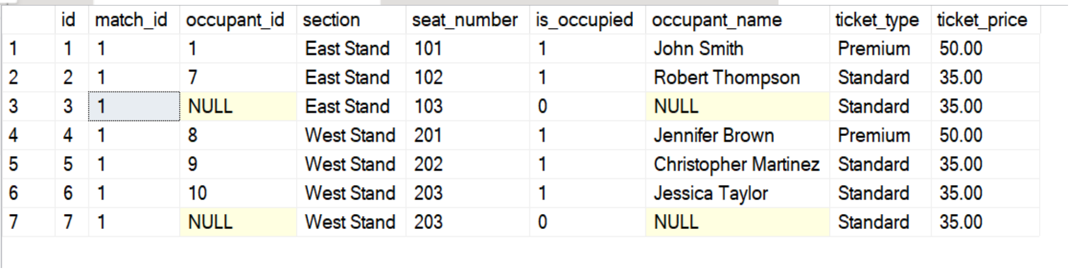
	match_id	home_team	away_team	game_date	game_time
1	1	Benfica	Sporting CP	2023-07-15	18:00:00.0000000
2	2	Benfica	FC Porto	2023-07-16	18:00:00.0000000
3	3	Benfica	SC Braga	2023-07-17	18:00:00.0000000
4	4	Benfica	Arouca	2023-07-18	18:00:00.0000000
5	5	FC Porto	Benfica	2023-07-21	18:00:00.0000000
6	6	Sporting CP	Benfica	2023-07-22	18:00:00.0000000
7	7	SC Braga	Benfica	2023-07-23	18:00:00.0000000
8	8	Arouca	Benfica	2023-07-24	18:00:00.0000000

FANS_DETAILS SQL TABLE



	fan_id	name	email	match_id	home_team	away_team	game_date	game_time	has_ticket
1	1	John Smith	simao.er.costa.pereira@gmail.com	1	Benfica	Sporting CP	2023-07-15	18:00:00.0000000	1
2	1	John Smith	simao.er.costa.pereira@gmail.com	2	Benfica	FC Porto	2023-07-16	18:00:00.0000000	1
3	1	John Smith	simao.er.costa.pereira@gmail.com	3	Benfica	SC Braga	2023-07-17	18:00:00.0000000	1
4	1	John Smith	simao.er.costa.pereira@gmail.com	4	Benfica	Arouca	2023-07-18	18:00:00.0000000	1
5	1	John Smith	simao.er.costa.pereira@gmail.com	5	FC Porto	Benfica	2023-07-21	18:00:00.0000000	1
6	1	John Smith	simao.er.costa.pereira@gmail.com	6	Sporting CP	Benfica	2023-07-22	18:00:00.0000000	1
7	1	John Smith	simao.er.costa.pereira@gmail.com	7	SC Braga	Benfica	2023-07-23	18:00:00.0000000	0
8	1	John Smith	simao.er.costa.pereira@gmail.com	8	Arouca	Benfica	2023-07-24	18:00:00.0000000	0
9	2	Jane Doe	simao.er.costa.pereira@gmail.com	1	Benfica	Sporting CP	2023-07-15	18:00:00.0000000	0
10	2	Jane Doe	simao.er.costa.pereira@gmail.com	2	Benfica	FC Porto	2023-07-16	18:00:00.0000000	1
11	2	Jane Doe	simao.er.costa.pereira@gmail.com	3	Benfica	SC Braga	2023-07-17	18:00:00.0000000	1
12	2	Jane Doe	simao.er.costa.pereira@gmail.com	4	Benfica	Arouca	2023-07-18	18:00:00.0000000	0
13	2	Jane Doe	simao.er.costa.pereira@gmail.com	5	FC Porto	Benfica	2023-07-21	18:00:00.0000000	0
14	2	Jane Doe	simao.er.costa.pereira@gmail.com	6	Sporting CP	Benfica	2023-07-22	18:00:00.0000000	1
15	2	Jane Doe	simao.er.costa.pereira@gmail.com	7	SC Braga	Benfica	2023-07-23	18:00:00.0000000	0
16	2	Jane Doe	simao.er.costa.pereira@gmail.com	8	Arouca	Benfica	2023-07-24	18:00:00.0000000	1
17	3	Michael Johnson	simao.er.costa.pereira@gmail.com	1	Benfica	Sporting CP	2023-07-15	18:00:00.0000000	0
18	3	Michael Johnson	simao.er.costa.pereira@gmail.com	2	Benfica	FC Porto	2023-07-16	18:00:00.0000000	0
19	3	Michael Johnson	simao.er.costa.pereira@gmail.com	3	Benfica	SC Braga	2023-07-17	18:00:00.0000000	1
20	3	Michael Johnson	simao.er.costa.pereira@gmail.com	4	Benfica	Arouca	2023-07-18	18:00:00.0000000	0
21	3	Michael Johnson	simao.er.costa.pereira@gmail.com	5	FC Porto	Benfica	2023-07-21	18:00:00.0000000	1
22	3	Michael Johnson	simao.er.costa.pereira@gmail.com	6	Sporting CP	Benfica	2023-07-22	18:00:00.0000000	0
23	3	Michael Johnson	simao.er.costa.pereira@gmail.com	7	SC Braga	Benfica	2023-07-23	18:00:00.0000000	1
24	3	Michael Johnson	simao.er.costa.pereira@gmail.com	8	Arouca	Benfica	2023-07-24	18:00:00.0000000	1
25	4	Sarah Wilson	simao.er.costa.pereira@gmail.com	1	Benfica	Sporting CP	2023-07-15	18:00:00.0000000	0
26	4	Sarah Wilson	simao.er.costa.pereira@gmail.com	2	Benfica	FC Porto	2023-07-16	18:00:00.0000000	0
27	4	Sarah Wilson	simao.er.costa.pereira@gmail.com	3	Benfica	SC Braga	2023-07-17	18:00:00.0000000	1
28	4	Sarah Wilson	simao.er.costa.pereira@gmail.com	4	Benfica	Arouca	2023-07-18	18:00:00.0000000	0
29	4	Sarah Wilson	simao.er.costa.pereira@gmail.com	5	FC Porto	Benfica	2023-07-21	18:00:00.0000000	0
30	4	Sarah Wilson	simao.er.costa.pereira@gmail.com	6	Sporting CP	Benfica	2023-07-22	18:00:00.0000000	1
31	4	Sarah Wilson	simao.er.costa.pereira@gmail.com	7	SC Braga	Benfica	2023-07-23	18:00:00.0000000	1
32	4	Sarah Wilson	simao.er.costa.pereira@gmail.com	8	Arouca	Benfica	2023-07-24	18:00:00.0000000	1

STADIUM OCCUPANCY (FOR MATCH_ID =1) SQL TABLE



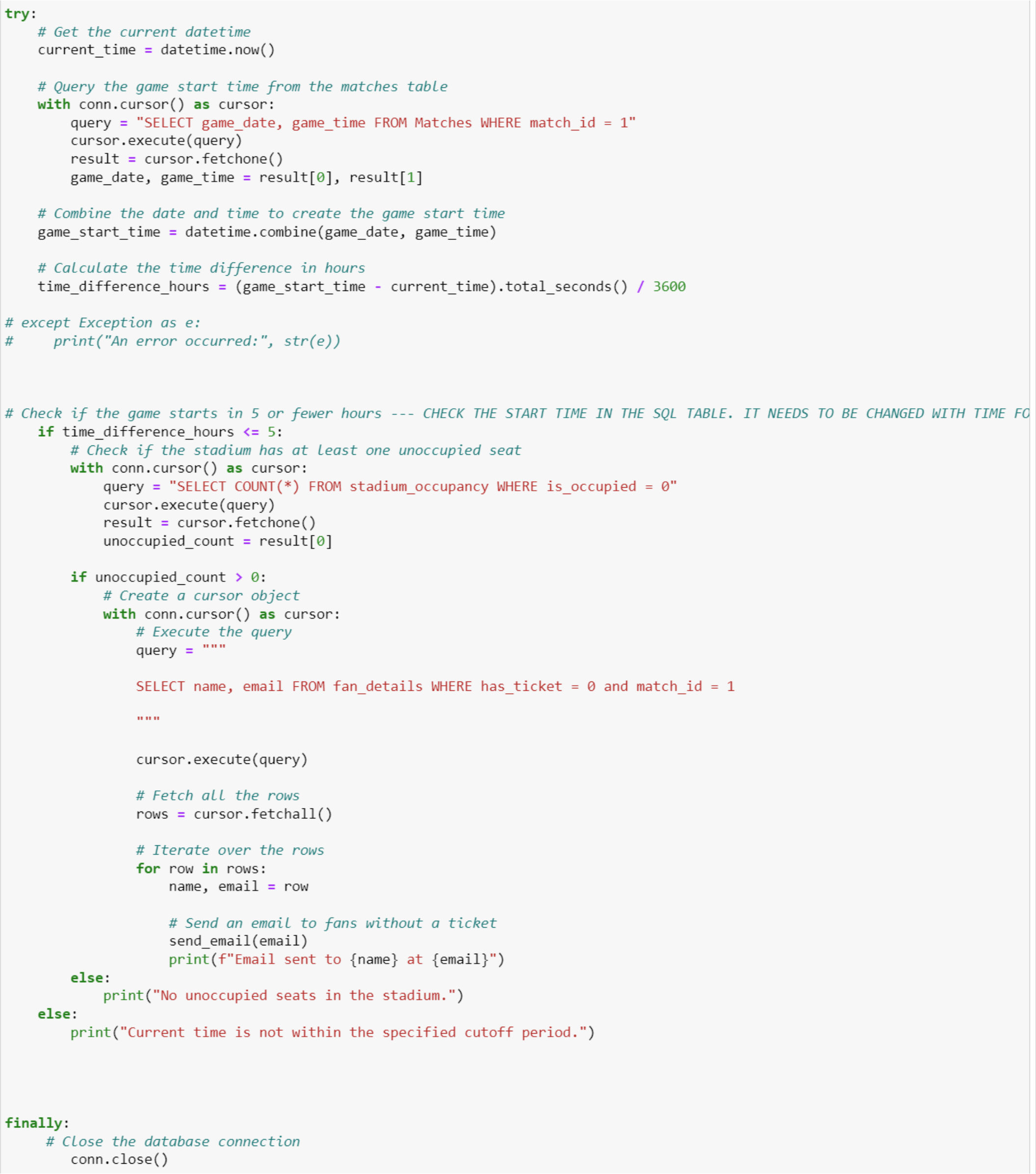
	id	match_id	occupant_id	section	seat_number	is_occupied	occupant_name	ticket_type	ticket_price
1	1	1	1	East Stand	101	1	John Smith	Premium	50.00
2	2	1	7	East Stand	102	1	Robert Thompson	Standard	35.00
3	3	1	NULL	East Stand	103	0	NULL	Standard	35.00
4	4	1	8	West Stand	201	1	Jennifer Brown	Premium	50.00
5	5	1	9	West Stand	202	1	Christopher Martinez	Standard	35.00
6	6	1	10	West Stand	203	1	Jessica Taylor	Standard	35.00
7	7	1	NULL	West Stand	203	0	NULL	Standard	35.00

METHODOLOGY

USE CASE 1: REAL TIME ACCESS TO A DATA WAREHOUSE AND TAKING LIVE ACTIONS

Action before the match:

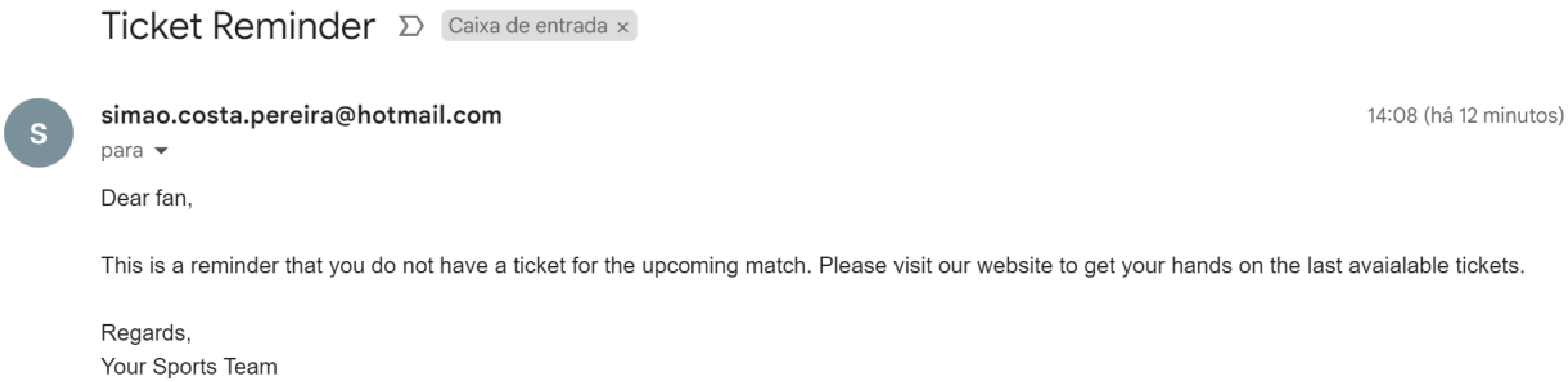
-Is it 5 hours before the match ?
-Fan has ticket ?



```
try:
    # Get the current datetime
    current_time = datetime.now()

    # Query the game start time from the matches table
    with conn.cursor() as cursor:
        query = "SELECT game_date, game_time FROM Matches WHERE match_id = 1"
        cursor.execute(query)
        result = cursor.fetchone()
        game_date, game_time = result[0], result[1]

    # Combine the date and time to create the game start time
    game_start_time = datetime.combine(game_date, game_time)

    # Calculate the time difference in hours
    time_difference_hours = (game_start_time - current_time).total_seconds() / 3600

# except Exception as e:
#     print("An error occurred:", str(e))



# Check if the game starts in 5 or fewer hours --- CHECK THE START TIME IN THE SQL TABLE. IT NEEDS TO BE CHANGED WITH TIME FO
    if time_difference_hours <= 5:
        # Check if the stadium has at least one unoccupied seat
        with conn.cursor() as cursor:
            query = "SELECT COUNT(*) FROM stadium_occupancy WHERE is_occupied = 0"
            cursor.execute(query)
            result = cursor.fetchone()
            unoccupied_count = result[0]

        if unoccupied_count > 0:
            # Create a cursor object
            with conn.cursor() as cursor:
                # Execute the query
                query = """

                SELECT name, email FROM fan_details WHERE has_ticket = 0 and match_id = 1

                """

                cursor.execute(query)

                # Fetch all the rows
                rows = cursor.fetchall()

                # Iterate over the rows
                for row in rows:
                    name, email = row

                    # Send an email to fans without a ticket
                    send_email(email)
                    print(f"Email sent to {name} at {email}")
        else:
            print("No unoccupied seats in the stadium.")
    else:
        print("Current time is not within the specified cutoff period.")



finally:
     # Close the database connection
        conn.close()
```

RESULTS

USE CASE 1: REAL TIME ACCESS TO A DATA WAREHOUSE AND TAKING LIVE ACTIONS



Ticket Reminder

simao.costa.pereira@hotmail.com

14:08 (há 12 minutos)

para

Dear fan,

This is a reminder that you do not have a ticket for the upcoming match. Please visit our website to get your hands on the last avaialable tickets.

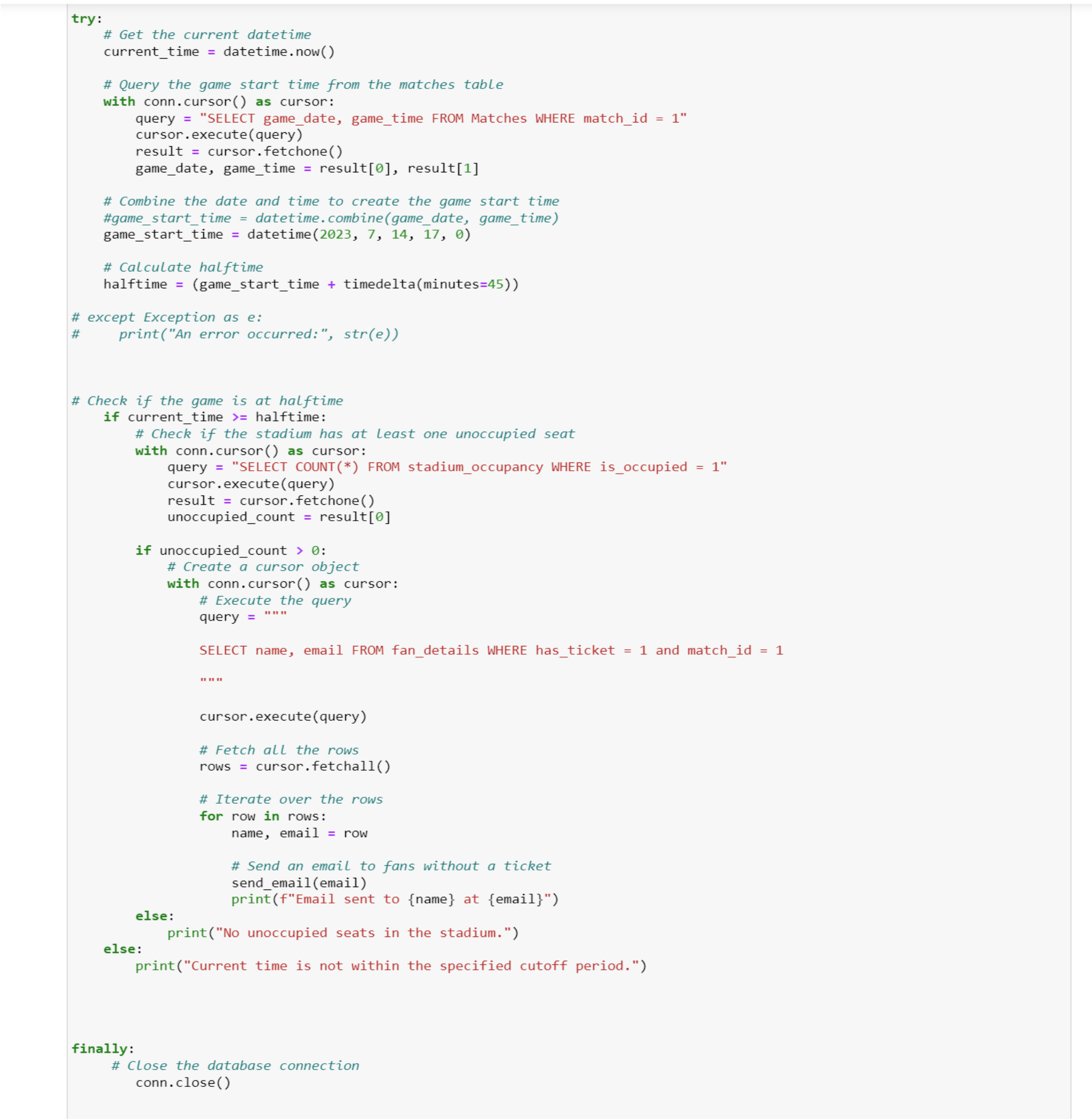
Regards,
Your Sports Team

METHODOLOGY

USE CASE 1: REAL TIME ACCESS TO A DATA WAREHOUSE AND TAKING LIVE ACTIONS

Action at halftime:

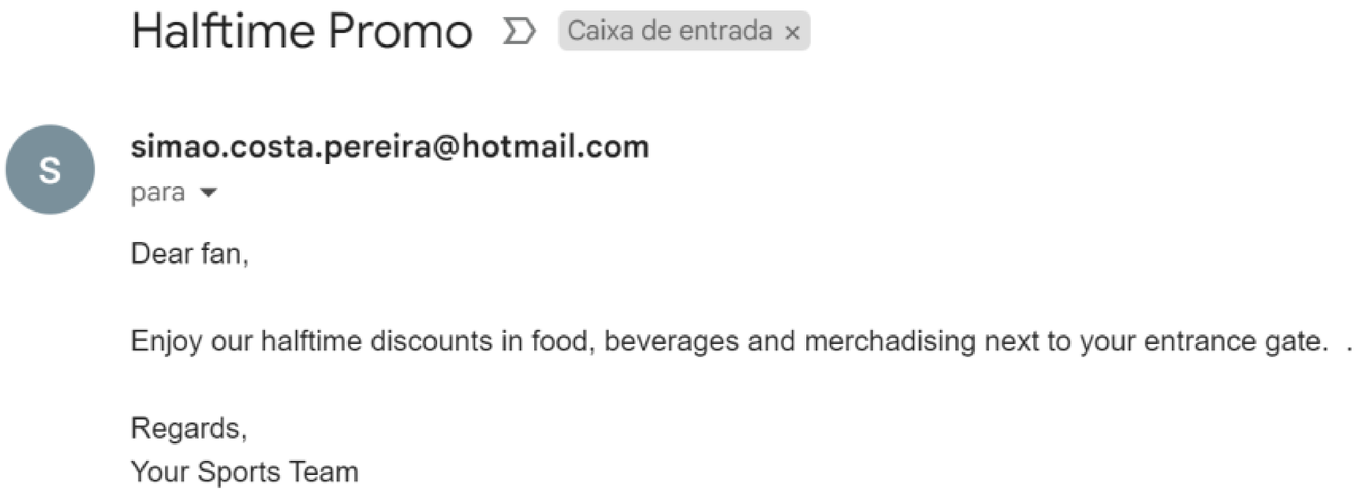
-Is it halftime ?
-Does the fan have a ticket for the match?



```python
try:
    # Get the current datetime
    current_time = datetime.now()

    # Query the game start time from the matches table
    with conn.cursor() as cursor:
        query = "SELECT game_date, game_time FROM Matches WHERE match_id = 1"
        cursor.execute(query)
        result = cursor.fetchone()
        game_date, game_time = result[0], result[1]

    # Combine the date and time to create the game start time
    #game_start_time = datetime.combine(game_date, game_time)
    game_start_time = datetime(2023, 7, 14, 17, 0)

    # Calculate halftime
    halftime = (game_start_time + timedelta(minutes=45))

# except Exception as e:
#     print("An error occurred:", str(e))



# Check if the game is at halftime
    if current_time >= halftime:
        # Check if the stadium has at least one unoccupied seat
        with conn.cursor() as cursor:
            query = "SELECT COUNT(*) FROM stadium_occupancy WHERE is_occupied = 1"
            cursor.execute(query)
            result = cursor.fetchone()
            unoccupied_count = result[0]

        if unoccupied_count > 0:
            # Create a cursor object
            with conn.cursor() as cursor:
                # Execute the query
                query = """

                SELECT name, email FROM fan_details WHERE has_ticket = 1 and match_id = 1

                """

                cursor.execute(query)

                # Fetch all the rows
                rows = cursor.fetchall()

                # Iterate over the rows
                for row in rows:
                    name, email = row

                    # Send an email to fans without a ticket
                    send_email(email)
                    print(f"Email sent to {name} at {email}")
        else:
            print("No unoccupied seats in the stadium.")
    else:
        print("Current time is not within the specified cutoff period.")



finally:
     # Close the database connection
        conn.close()
```

RESULTS

USE CASE 1: REAL TIME ACCESS TO A DATA WAREHOUSE AND TAKING LIVE ACTIONS



Halftime Promo

simao.costa.pereira@hotmail.com

Dear fan,

Enjoy our halftime discounts in food, beverages and merchadising next to your entrance gate. .

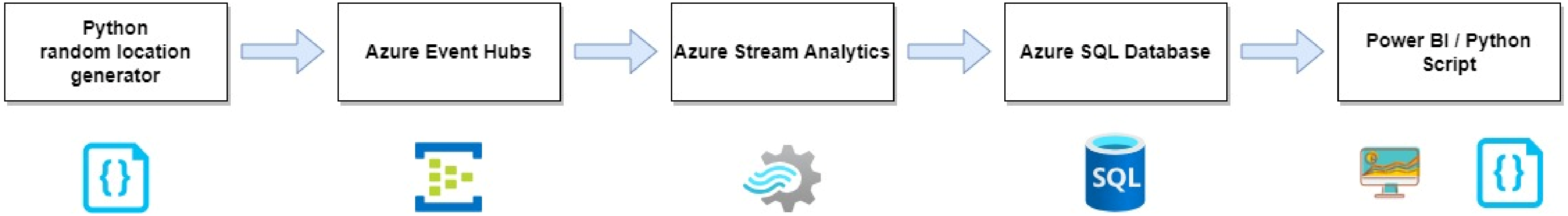
Regards,
Your Sports Team

USE CASE 2: USING REAL-TIME LOCATION ON A MATCHDAY TO INTERACT WITH THE FANS

LIVE ACTIONS

Action 1
If the distance of the fan to Estádio da Luz is more than 5km and it is less than 3 hours to the kick-off

Action 2
If the fan is less than 5km away from the stadium and the game starts within 3 hours.

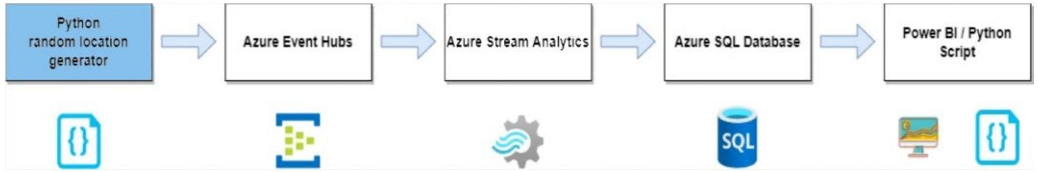


Architecture benefits:

- Architecture enables the ingestion of real-time fan location and facilitates immediate actions
- Capacity to store data for a future in-depth analysis (SQL Database)
- Real time data visualization (Power BI)

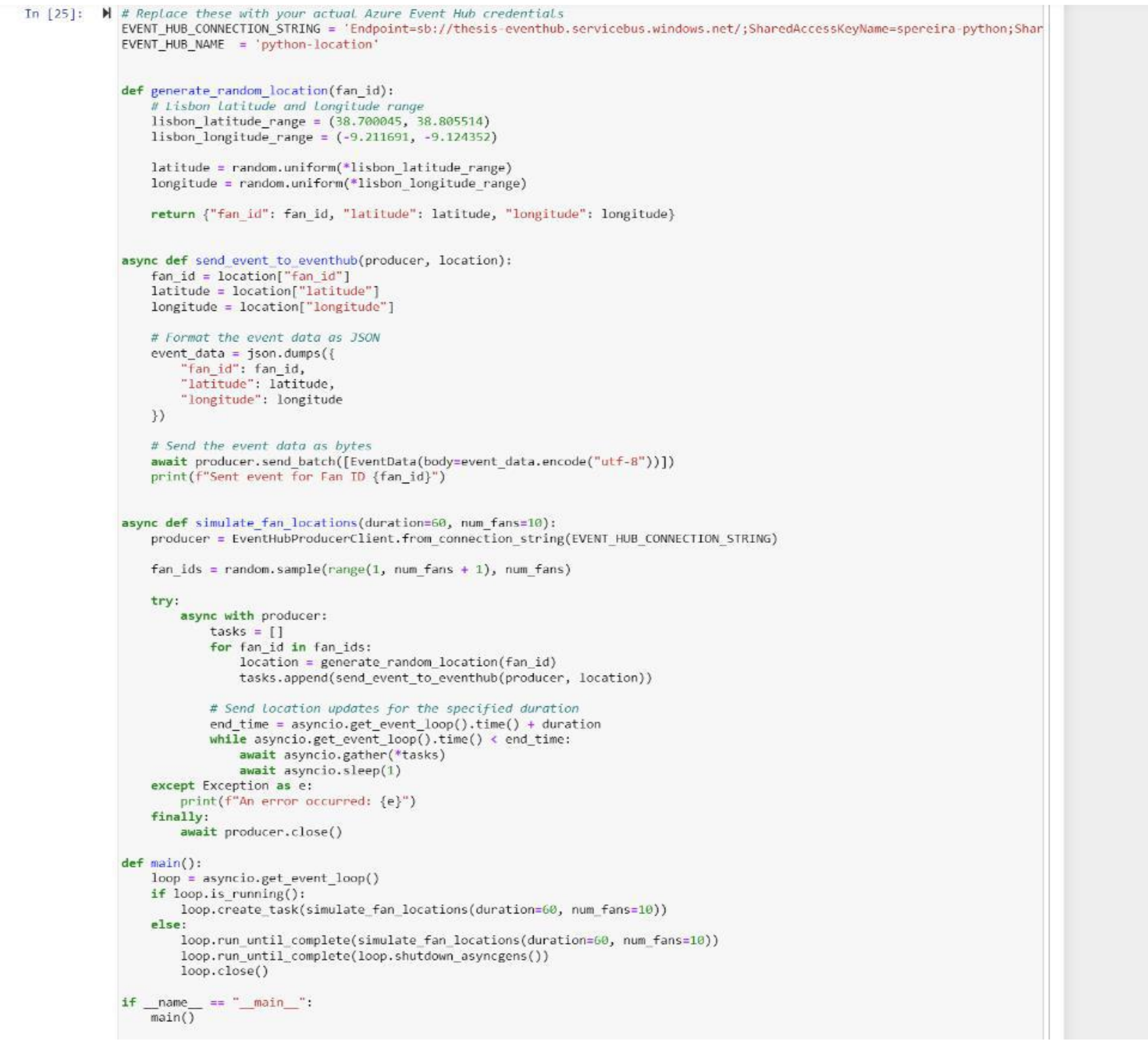
METHODOLOGY

USE CASE 2: USING REAL-TIME LOCATION ON A MATCHDAY TO INTERACT WITH THE FANS



- The code generates random location coordinates(latitude and longitude) for 10 fans in the Lisbon area.

- Python sends event data at regular intervals through its capability to connect to an azure event hub using the provided string and event hub name. connection

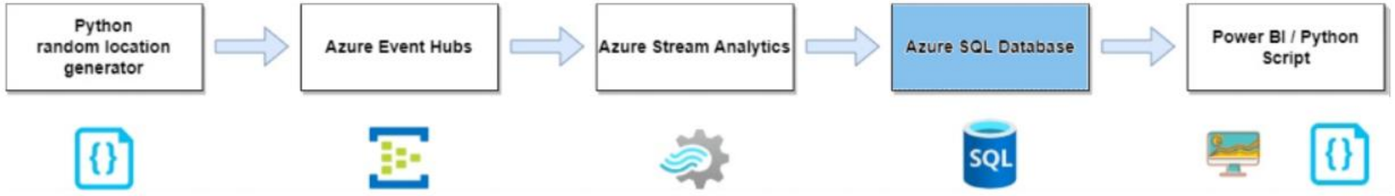


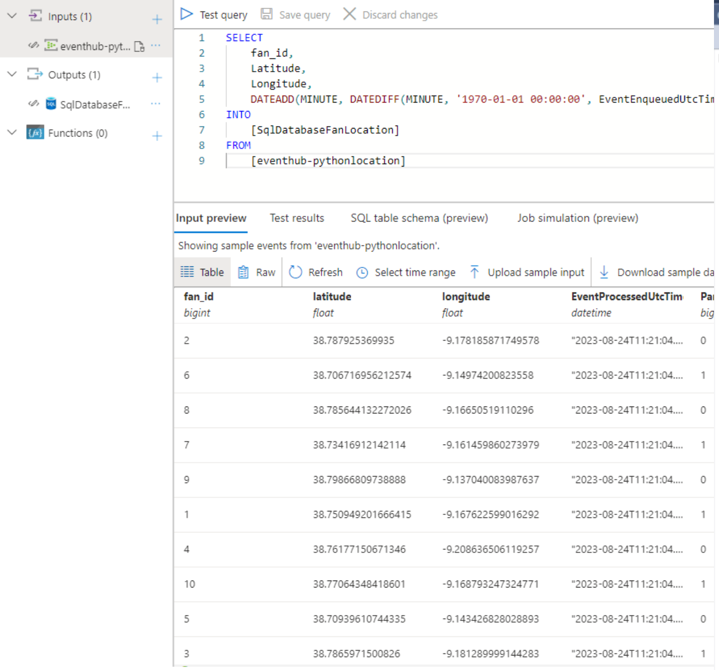
In [25]:

```python
# Replace these with your actual Azure Event Hub credentials
EVENT_HUB_CONNECTION_STRING = 'Endpoint=sb://thesis-eventhub.servicebus.windows.net/;SharedAccessKeyName=spereira-python;Shar
EVENT_HUB_NAME  = 'python-location'


def generate_random_location(fan_id):
    # Lisbon latitude and longitude range
    lisbon_latitude_range = (38.700045, 38.805514)
    lisbon_longitude_range = (-9.211691, -9.124352)

    latitude = random.uniform(*lisbon_latitude_range)
    longitude = random.uniform(*lisbon_longitude_range)

    return {"fan_id": fan_id, "latitude": latitude, "longitude": longitude}


async def send_event_to_eventhub(producer, location):
    fan_id = location["fan_id"]
    latitude = location["latitude"]
    longitude = location["longitude"]

    # Format the event data as JSON
    event_data = json.dumps({
        "fan_id": fan_id,
        "latitude": latitude,
        "longitude": longitude
    })

    # Send the event data as bytes
    await producer.send_batch([EventData(body=event_data.encode("utf-8"))])
    print(f"Sent event for Fan ID {fan_id}")


async def simulate_fan_locations(duration=60, num_fans=10):
    producer = EventHubProducerClient.from_connection_string(EVENT_HUB_CONNECTION_STRING)

    fan_ids = random.sample(range(1, num_fans + 1), num_fans)

    try:
        async with producer:
            tasks = []
            for fan_id in fan_ids:
                location = generate_random_location(fan_id)
                tasks.append(send_event_to_eventhub(producer, location))

            # Send location updates for the specified duration
            end_time = asyncio.get_event_loop().time() + duration
            while asyncio.get_event_loop().time() < end_time:
                await asyncio.gather(*tasks)
                await asyncio.sleep(1)
    except Exception as e:
        print(f"An error occurred: {e}")
    finally:
        await producer.close()

def main():
    loop = asyncio.get_event_loop()
    if loop.is_running():
        loop.create_task(simulate_fan_locations(duration=60, num_fans=10))
    else:
        loop.run_until_complete(simulate_fan_locations(duration=60, num_fans=10))
        loop.run_until_complete(loop.shutdown_asyncgens())
        loop.close()

if __name__ == "__main__":
    main()
```

RESULTS

USE CASE 2: USING REAL-TIME LOCATION ON A MATCHDAY TO INTERACT WITH THE FANS

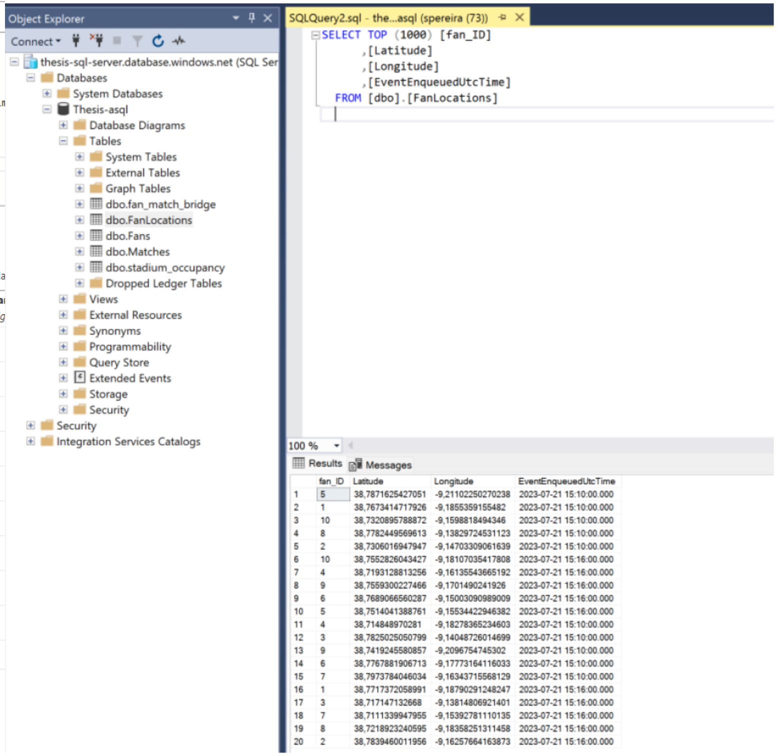




```sql
SELECT
    fan_id,
    Latitude,
    Longitude,
    DATEADD(MINUTE, DATEDIFF(MINUTE, '1970-01-01 00:00:00', EventEnqueuedUtcTim
INTO
    [SqlDatabaseFanLocation]
FROM
    [eventhub-pythonlocation]
```

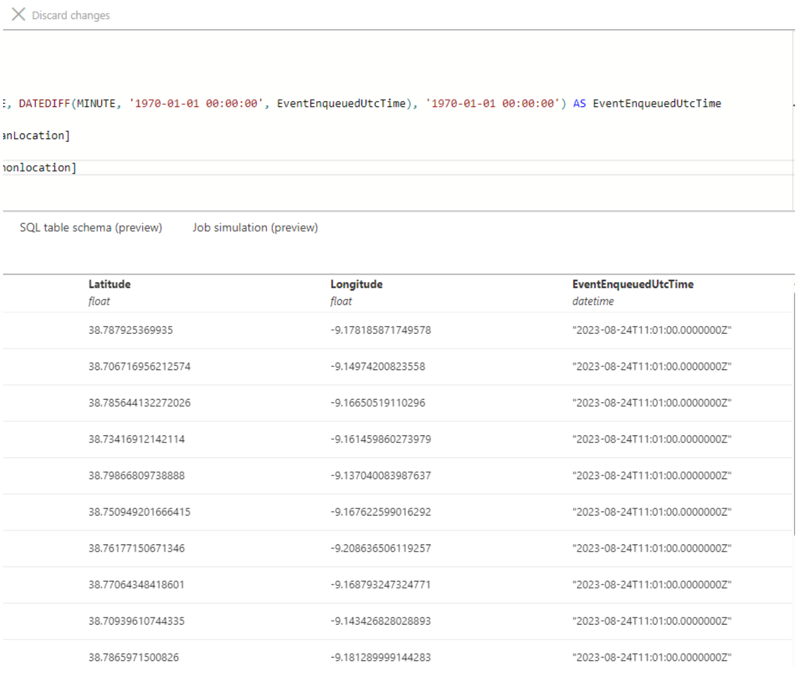
Showing sample events from 'eventhub-pythonlocation'.

fan_id *bigint*	latitude *float*	longitude *float*	EventProcessedUtcTim *datetime*	Pa *big*
2	38.787925369935	-9.178185871749578	"2023-08-24T11:21:04....	0
6	38.706716956212574	-9.14974200823558	"2023-08-24T11:21:04....	1
8	38.785644132272026	-9.16650519110296	"2023-08-24T11:21:04....	0
7	38.73416912142114	-9.161459860273979	"2023-08-24T11:21:04....	1
9	38.79866809738888	-9.137040083987637	"2023-08-24T11:21:04....	0
1	38.750949201666415	-9.167622599016292	"2023-08-24T11:21:04....	1
4	38.76177150671346	-9.208636506119257	"2023-08-24T11:21:04....	0
10	38.77064348418601	-9.168793247324771	"2023-08-24T11:21:04....	1
5	38.70939610744335	-9.143426828028893	"2023-08-24T11:21:04....	0
3	38.7865971500826	-9.181289999144283	"2023-08-24T11:21:04....	1



```sql
SELECT TOP (1000) [fan_ID]
      ,[Latitude]
      ,[Longitude]
      ,[EventEnqueuedUtcTime]
  FROM [dbo].[FanLocations]
```

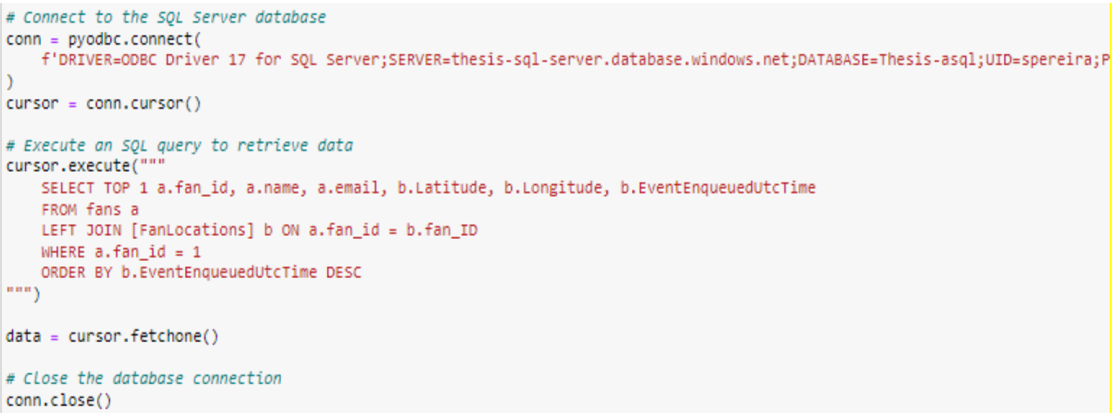
	fan_ID	Latitude	Longitude	EventEnqueuedUtcTime
1	5	38,7871625427051	-9,21102250270238	2023-07-21 15:10:00.000
2	1	38,7673414717926	-9,1855359155482	2023-07-21 15:10:00.000
3	10	38,7320895788872	-9,1598818494346	2023-07-21 15:10:00.000
4	8	38,7782449569613	-9,13829724531123	2023-07-21 15:10:00.000
5	2	38,7306016947947	-9,14703309061639	2023-07-21 15:10:00.000
6	10	38,7552826043427	-9,18107035417808	2023-07-21 15:16:00.000
7	4	38,7193128813256	-9,16135543665192	2023-07-21 15:16:00.000
8	9	38,7559300227466	-9,1701490241926	2023-07-21 15:16:00.000
9	6	38,7689066560287	-9,15003090989009	2023-07-21 15:16:00.000
10	5	38,7514041388761	-9,15534422946382	2023-07-21 15:16:00.000
11	4	38,714848970281	-9,18278365234603	2023-07-21 15:10:00.000
12	3	38,7825025050799	-9,14048726014699	2023-07-21 15:10:00.000
13	9	38,7419245580857	-9,2096754745302	2023-07-21 15:10:00.000
14	6	38,7767881906713	-9,17773164116033	2023-07-21 15:10:00.000
15	7	38,7973784046034	-9,16343715568129	2023-07-21 15:10:00.000
16	1	38,7717372058991	-9,18790291248247	2023-07-21 15:16:00.000
17	3	38,717147132668	-9,13814806921401	2023-07-21 15:16:00.000
18	7	38,7111339947955	-9,15392781110135	2023-07-21 15:16:00.000
19	8	38,7218923240595	-9,18358251311458	2023-07-21 15:16:00.000
20	2	38,7839460011956	-9,16257664163873	2023-07-21 15:16:00.000



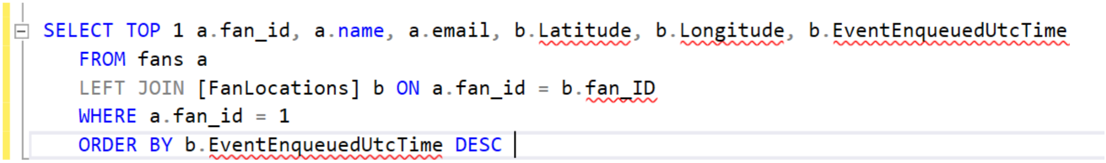
```sql
E, DATEDIFF(MINUTE, '1970-01-01 00:00:00', EventEnqueuedUtcTime), '1970-01-01 00:00:00') AS EventEnqueuedUtcTime

anLocation]

nonlocation]
```

Latitude *float*	Longitude *float*	EventEnqueuedUtcTime *datetime*
38.787925369935	-9.178185871749578	"2023-08-24T11:01:00.0000000Z"
38.706716956212574	-9.14974200823558	"2023-08-24T11:01:00.0000000Z"
38.785644132272026	-9.16650519110296	"2023-08-24T11:01:00.0000000Z"
38.73416912142114	-9.161459860273979	"2023-08-24T11:01:00.0000000Z"
38.79866809738888	-9.137040083987637	"2023-08-24T11:01:00.0000000Z"
38.750949201666415	-9.167622599016292	"2023-08-24T11:01:00.0000000Z"
38.76177150671346	-9.208636506119257	"2023-08-24T11:01:00.0000000Z"
38.77064348418601	-9.168793247324771	"2023-08-24T11:01:00.0000000Z"
38.70939610744335	-9.143426828028893	"2023-08-24T11:01:00.0000000Z"
38.7865971500826	-9.181289999144283	"2023-08-24T11:01:00.0000000Z"

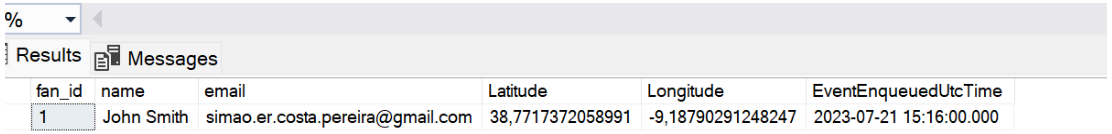
USE CASE 2: USING REAL-TIME LOCATION ON A MATCHDAY TO INTERACT WITH THE FANS



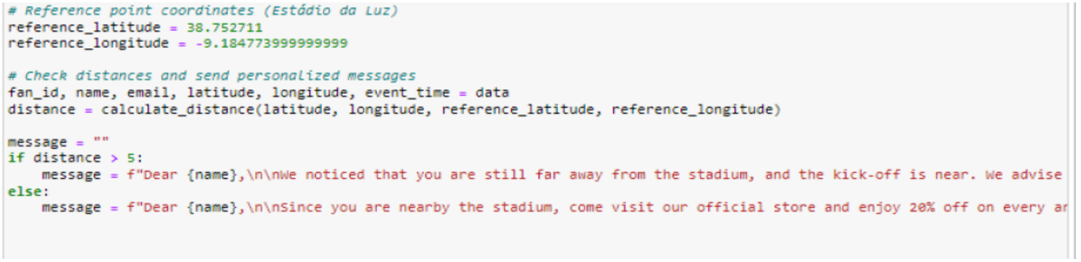
```
# Connect to the SQL Server database
conn = pyodbc.connect(
    f'DRIVER=ODBC Driver 17 for SQL Server;SERVER=thesis-sql-server.database.windows.net;DATABASE=Thesis-asql;UID=spereira;P
)
cursor = conn.cursor()

# Execute an SQL query to retrieve data
cursor.execute("""
    SELECT TOP 1 a.fan_id, a.name, a.email, b.Latitude, b.Longitude, b.EventEnqueuedUtcTime
    FROM fans a
    LEFT JOIN [FanLocations] b ON a.fan_id = b.fan_ID
    WHERE a.fan_id = 1
    ORDER BY b.EventEnqueuedUtcTime DESC
""")

data = cursor.fetchone()

# Close the database connection
conn.close()
```



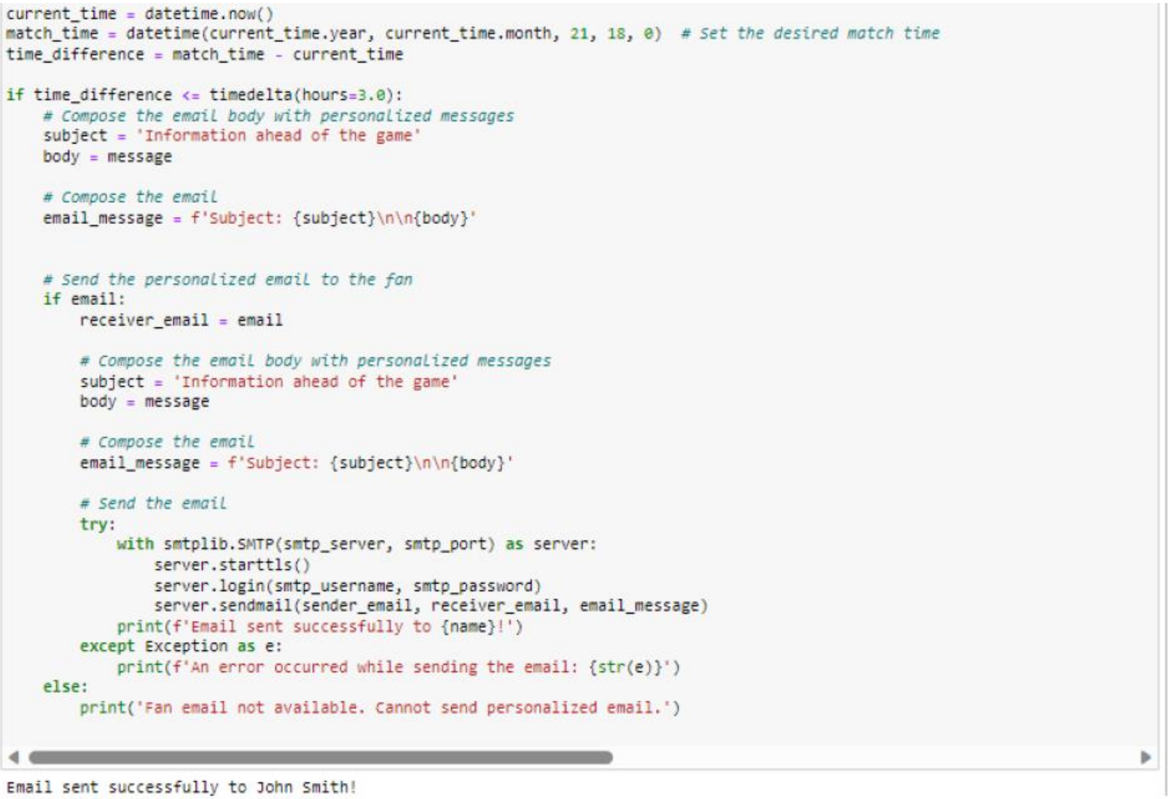
```
SELECT TOP 1 a.fan_id, a.name, a.email, b.Latitude, b.Longitude, b.EventEnqueuedUtcTime
   FROM fans a
   LEFT JOIN [FanLocations] b ON a.fan_id = b.fan_ID
   WHERE a.fan_id = 1
   ORDER BY b.EventEnqueuedUtcTime DESC
```



fan_id	name	email	Latitude	Longitude	EventEnqueuedUtcTime
1	John Smith	simao.er.costa.pereira@gmail.com	38,7717372058991	-9,18790291248247	2023-07-21 15:16:00.000



```
# Reference point coordinates (Estádio da Luz)
reference_latitude = 38.752711
reference_longitude = -9.184773999999999

# Check distances and send personalized messages
fan_id, name, email, latitude, longitude, event_time = data
distance = calculate_distance(latitude, longitude, reference_latitude, reference_longitude)

message = ""
if distance > 5:
    message = f"Dear {name},\n\nWe noticed that you are still far away from the stadium, and the kick-off is near. We advise
else:
    message = f"Dear {name},\n\nSince you are nearby the stadium, come visit our official store and enjoy 20% off on every ar
```

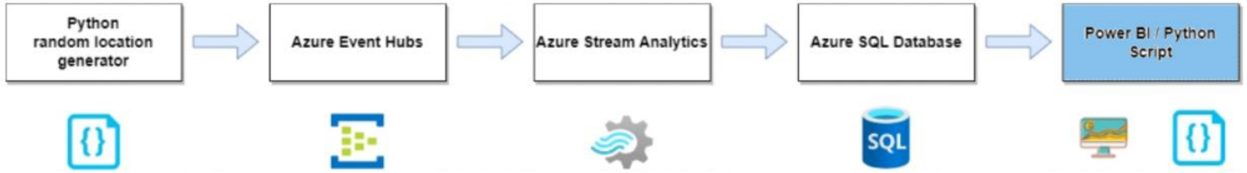


```
current_time = datetime.now()
match_time = datetime(current_time.year, current_time.month, 21, 18, 0)  # Set the desired match time
time_difference = match_time - current_time

if time_difference <= timedelta(hours=3.0):
    # Compose the email body with personalized messages
    subject = 'Information ahead of the game'
    body = message

    # Compose the email
    email_message = f'Subject: {subject}\n\n{body}'


    # Send the personalized email to the fan
    if email:
        receiver_email = email

        # Compose the email body with personalized messages
        subject = 'Information ahead of the game'
        body = message

        # Compose the email
        email_message = f'Subject: {subject}\n\n{body}'

        # Send the email
        try:
            with smtplib.SMTP(smtp_server, smtp_port) as server:
                server.starttls()
                server.login(smtp_username, smtp_password)
                server.sendmail(sender_email, receiver_email, email_message)
            print(f'Email sent successfully to {name}!')
        except Exception as e:
            print(f'An error occurred while sending the email: {str(e)}')
    else:
        print('Fan email not available. Cannot send personalized email.')
```

```
Email sent successfully to John Smith!
```

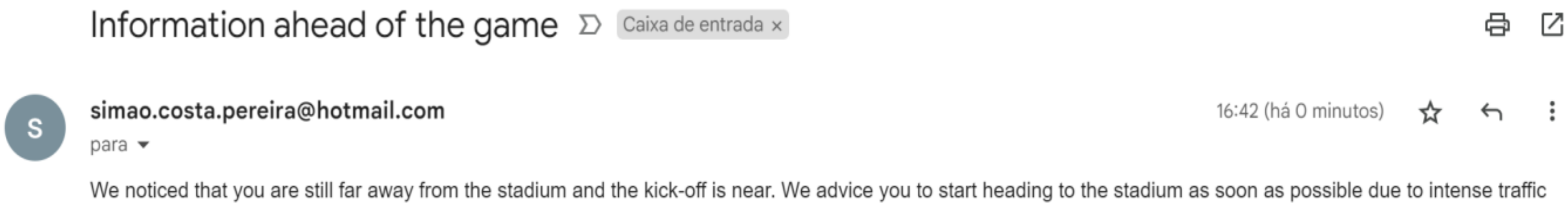
RESULTS

USE CASE 2: USING REAL-TIME LOCATION ON A MATCHDAY TO INTERACT WITH THE FANS



Action 1

- 3 hours before the match
- More than 5 km away from Estádio da Luz



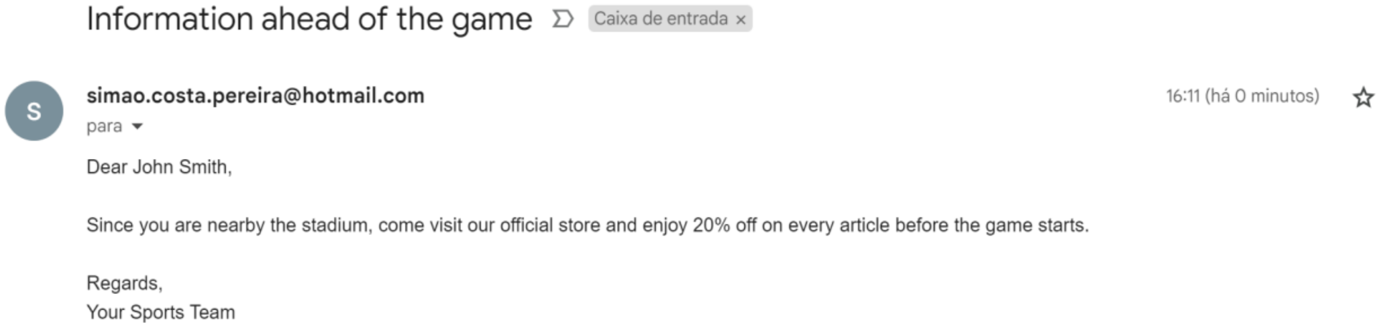
Information ahead of the game

simao.costa.pereira@hotmail.com

We noticed that you are still far away from the stadium and the kick-off is near. We advice you to start heading to the stadium as soon as possible due to intense traffic

Action 2

- 3 hours before the match
- Less than 5 km away from Estádio da Luz



Information ahead of the game

simao.costa.pereira@hotmail.com

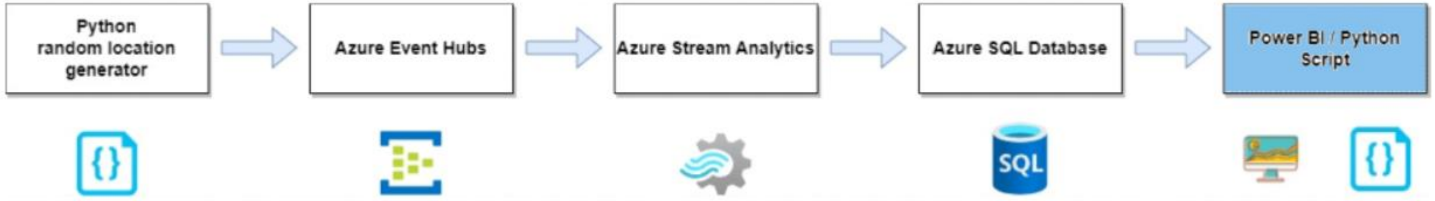
Dear John Smith,

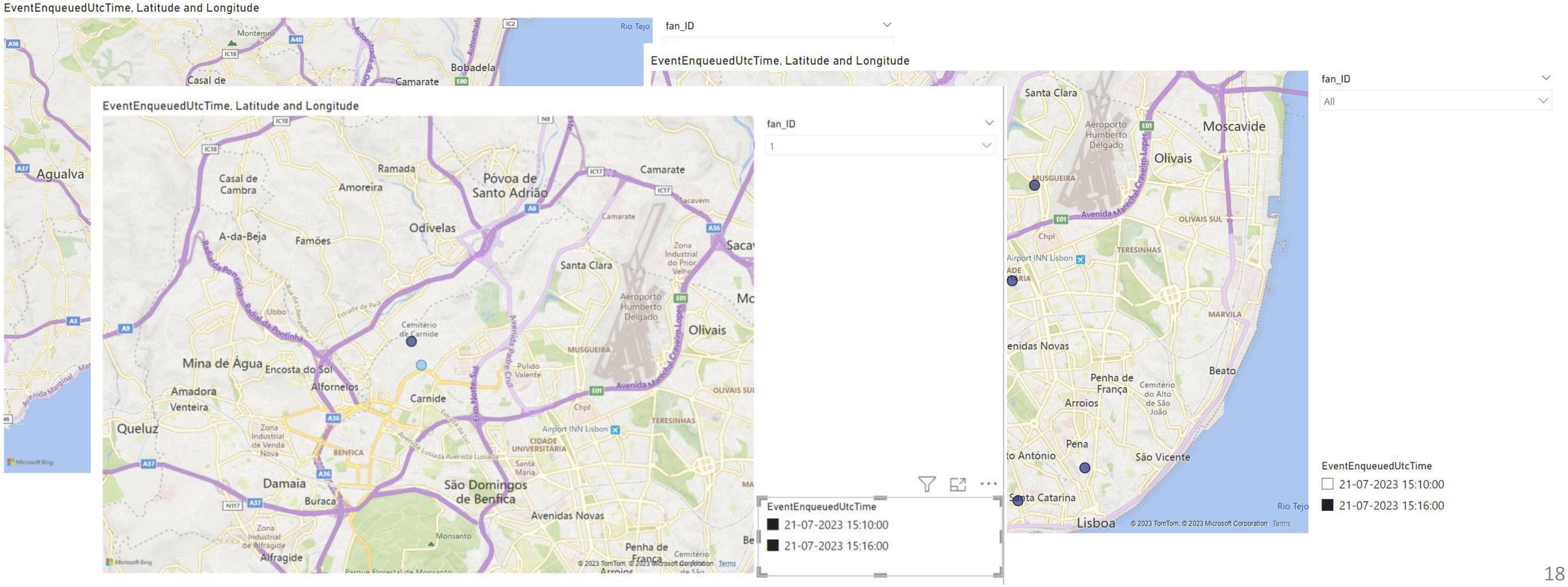
Since you are nearby the stadium, come visit our official store and enjoy 20% off on every article before the game starts.

Regards,
Your Sports Team

RESULTS

USE CASE 2: USING REAL-TIME LOCATION ON A MATCHDAY TO INTERACT WITH THE FANS




EventEnqueuedUtcTime, Latitude and Longitude

fan_ID

EventEnqueuedUtcTime, Latitude and Longitude

fan_ID

All

EventEnqueuedUtcTime, Latitude and Longitude

fan_ID

1

EventEnqueuedUtcTime

■ 21-07-2023 15:10:00

■ 21-07-2023 15:16:00

EventEnqueuedUtcTime

□ 21-07-2023 15:10:00

■ 21-07-2023 15:16:00

CONCLUSION

Accomplishments

- Illustrated how **data and analytics** can be used to optimize operations, **fan engagement** as well as **organizations revenue**
- Showcased some of **possibilities** on how the sports industry can benefit from **leveraging its data**.
- **Leveraging historical databases** of sport entities **and perform live actions** for optimizing club operations, merchandising strategies and fans experience.
- **Ingestion of real time events** (live location) were used to **immediately interact with the fans** and it was shown how this data can be used for instant actions and also for future analysis.
- The tecnhonologies used such as **Azure Cloud** create a **innovative and effective solution** in the area of study.

FUTURE WORK

Expanding Use Cases: Explore more aspects of fan engagement for a comprehensive understanding of BI applications.

Advanced Cloud Features: Allocate more resources to examine more sophisticated and different cloud service capabilities.

Data Diversity and Volume: Integrate various data types for example like live traffic, ticket information, and weather data for richer analysis.

Practical Application Testing: Develop or utilize an existing sports mobile app to test the BI solution on a larger dataset for empirical validation.

Thank you!

Address: Campus de Campolide, 1070-312 Lisboa, Portugal
Phone: +351 213 828 610 Fax: +351 213 828 611

Acreditações e Certificações

Instituto Superior de Estatística e Gestão da Informação
Universidade Nova de Lisboa