



GWINNETT COUNTY PUBLIC SCHOOLS REAL-TIME BUS MONITORING

Milestone I Presentation



TEAM 3
CS 4850 – PROFESSOR PERRY

SAM BOSTIAN



Team Role: Team Leader

Major: Computer Science

Graduation: Fall 2024

Experience:

- Head Brewer at Sweeten Creek Brewing
- Research Project – Parallel Processing

About me:

- Enjoys woodwork and house projects
- Reading Technical Books

sbostian@students.kennesaw.edu

BRIAN PRUITT



Team Role: Documentation

Major: Computer Science

Graduation: Fall 2024

Experience:

- Software Engineer Intern with The Home Depot for 3 Summers
- Primarily worked with front-end React applications and GCP database solutions.

About me:

- Getting married in October 2024!
- Has two dogs, Ella and Nova
- Part-time wedding DJ

bpruitt9@students.kennesaw.edu

MICHAEL RIZIG



Team Role: Developer

Major: Computer Science

Graduation: Spring 2025

Experience:

- Intern at Delta Community Credit Union for 2 years

About Me:

- Dual Degree Program at Kennesaw State University (BS-CS / MS-CS)
- Big Fan of the Office

mrizig@students.kennesaw.edu

CHARLIE MCLARTY



Team Role: Developer

Major: Computer Science

Graduation: Fall 2024

Experience:

- Full Stack Developer Intern at Renmatix
- Undergraduate Research Assistant Fall 2021 - Spring 2022

About Me:

- Enjoys disc golf and watching football

cmclart4@students.kennesaw.edu

ALLEN ROMAN



Team Role: Developer

Major: Computer Science

Graduation: Spring 2025

Experience:

- Intern at ADP in front-end and cloud development

About Me:

- Play bass in a band
- Love snowboarding

aroman14@students.kennesaw.edu

COLLABORATION TOOLS



Github:

Code Management and Version Control

- We use GitHub to track code changes, manage versions, and collaborate on code development.



Microsoft Office:

Document creation and project management

- We use Microsoft Office tools like Word and PowerPoint for documentation, reports, and team collaboration.

COMMUNICATION TOOLS



Microsoft
Teams:

Used for formal communication, weekly meetings with stakeholders (industry), and project status updates.



Discord:

Used for day-to-day team communication, quick messaging, and informal discussions.

PROJECT PLAN

GCPS Bus Monitoring System

Sam Bostian, Michael Rizig, Charlie McLarty, Brian Pruitt

Project Start					August		September		October		November		December										
					Plan		M1		M2		M3		M4		M5								
TASK	ASSIGNED TO	PROGRESS	START	END	18	25	1	8	16	23	30	7	14	21	28	1	8	15	22	2	9	16	23
Planning																							
Define goals and project scope	ALL	100%	8/12/2024	8/15/2024	2																		
Define timeline	Michael Rizig	100%	8/15/2024	8/17/2024	2	2																	
Set Milestone Dates	Sam Bostain	100%	8/17/2024	8/21/2024		2	2																
Define Resources Required	Charlie McLarty	100%	8/21/2024	8/24/2024			2	3															
Complete Project Plan	Michael Rizig	100%	8/18/2024	9/1/2024			4																
Design																							
Define Dependencies	Sam Bostian	0%	8/12/2024	8/31/2024	2	2																	
Create SDD Document	Brian Pruitt	0%	8/12/2024	8/28/2024	2	2	2																
Create SRS Documents	ALL	0%	8/12/2024	8/31/2024	2	3	2																
Define Design Elements	ALL	0%	8/12/2024	8/31/2024	2	3	3																
Compare/Analyze designs	Sam,Michael,Brian	0%	8/12/2024	9/2/2024		3	3																
Complete SDD	Sam,Charlie,Brian	0%	8/12/2024	9/2/2024		4	4																
Complete SRS	ALL	0%	8/12/2024	9/2/2024		4	4																
Execution																							
Create Development Document	Michael Rizig	0%	9/10/2024	9/30/2024				3	2	2													
Build Backend	Sam Bostian	0%	9/10/2024	9/30/2024				3	2	2	2												
Establish Database	Charlie McLarty	0%	9/15/2024	9/30/2024							2	2	2	2									
Containerization	Sam,Michael	0%	10/8/2024	10/29/2024										2	4								
Create Presentation and Demo	Brian Pruitt	0%	10/8/2024	10/29/2024										3	3	4							
Test, evaluate, and debug	Charlie McLarty	0%	9/20/2024	10/29/2024										3	3	4							
Evaluation																							
Create Software Test Plan	Charlie McLarty	0%	9/23/2024	11/27/2024													2	2	2				
Clean up Prototype	Sam,Michael,Brian	0%	9/28/2024	11/27/2024												2	2	2	2				2
Final debugging	Sam, Charlie	0%	9/28/2024	11/27/2024													2	2	2		2	2	2
Complete Testing Documentation	Michael, Charlie	0%	9/29/2024	11/27/2024																	2	2	2
Complete Final report	ALL	0%	9/17/2024	12/2/2024													2	2	2	2	2	2	2

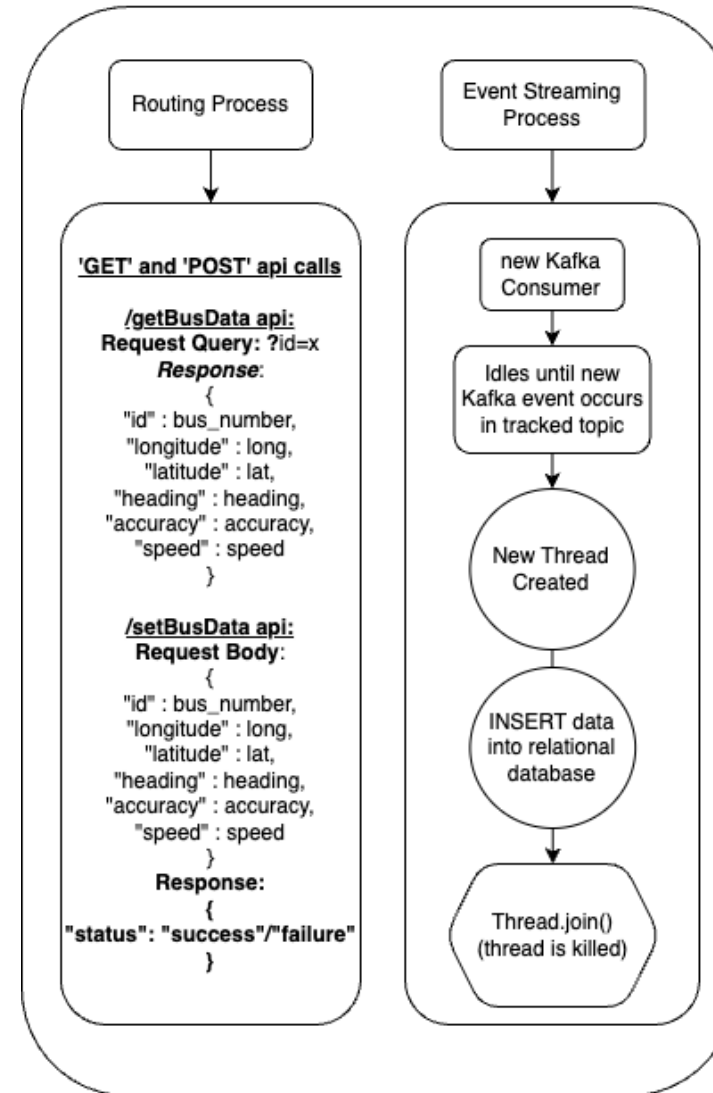
Milestone	Description	Due Date	Completion
Plan	Project Plan Document	9/1/2024	<input type="checkbox"/>
M1	Initial Development	9/16/2024	<input type="checkbox"/>
M2	Database and Event Lo	10/14/2024	<input type="checkbox"/>
M3	Containerization	10/28/2024	<input type="checkbox"/>
M4	C-Day	11/19/2024	<input type="checkbox"/>
M5	Final Deliverable	12/2/2024	<input type="checkbox"/>

9/2/2024
9/16/2024
10/14/2024
10/28/2024
11/19/2024
12/2/2024

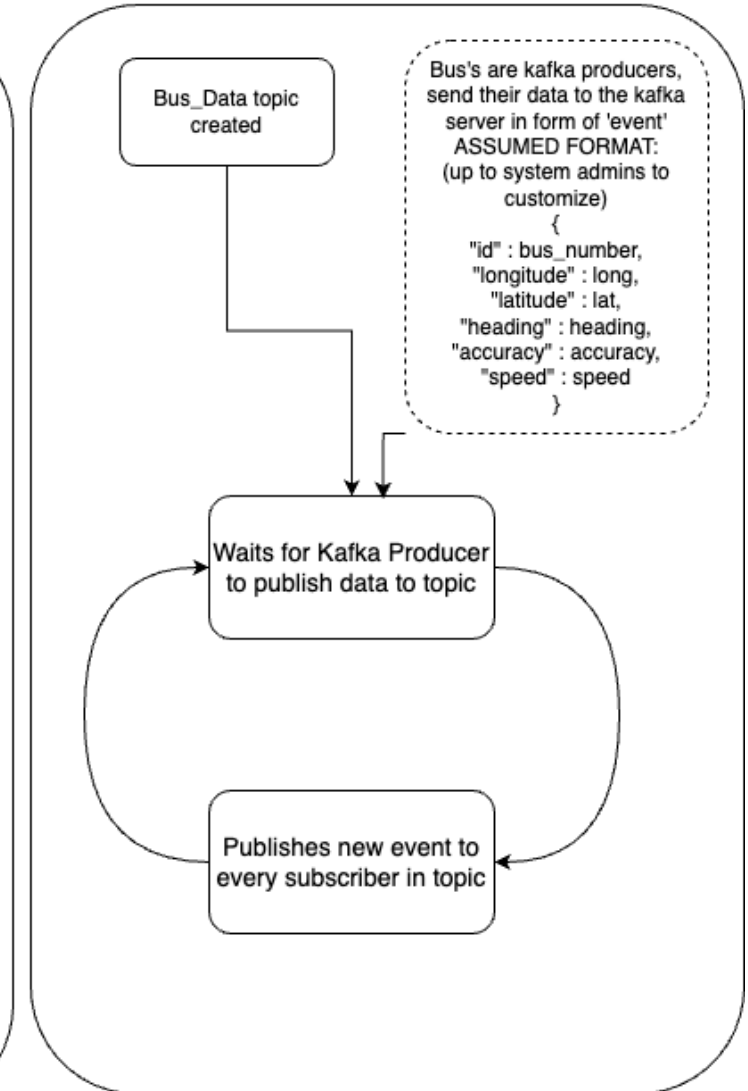
DESIGN

Server Process Design

main.py



Kafka Server



Query 1

Bus1 - Table

Bus1

Name: Bus1

Schema: test_bus_data

Column	Datatype	PK	NN	UQ	B...	UN	ZF	AI	G	Default / Expression
time	DATETIME	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
longitude	FLOAT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
latitude	FLOAT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
heading	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
accuracy	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
speed	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
<click to edit>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Column details 'time'

Column Name: time

Charset/Collation: Default Charset Default Collation

Comments:

Datatype: DATETIME

Default

Storage: VIRTUAL STORED

☒ Primary Key ☒ Not NULL ☐ Unique

☐ Binary ☐ Unsigned ☐ ZeroFill

☐ Auto Increment ☐ Generated

Columns

Indexes

Foreign Keys

Triggers

Partitioning

Options

Apply

Revert

Action Output

	Time	Action	Response	Duration / Fetch Time
16	12:53:24	USE test_bus_data	0 row(s) affected	0.00044 sec
17	12:53:24	INSERT INTO test_bus_data.Bus1 (time,longitude,latitude,heading,accuracy,speed) VALUES...	Error Code: 1305. FUNCTION test_bus_data.GETDATE...	0.0023 sec
18	12:54:30	INSERT INTO test_bus_data.Bus1 (time,longitude,latitude,heading,accuracy,speed) VALUES...	1 row(s) affected	0.0013 sec
19	13:19:06	Apply changes to Bus1	No changes detected	

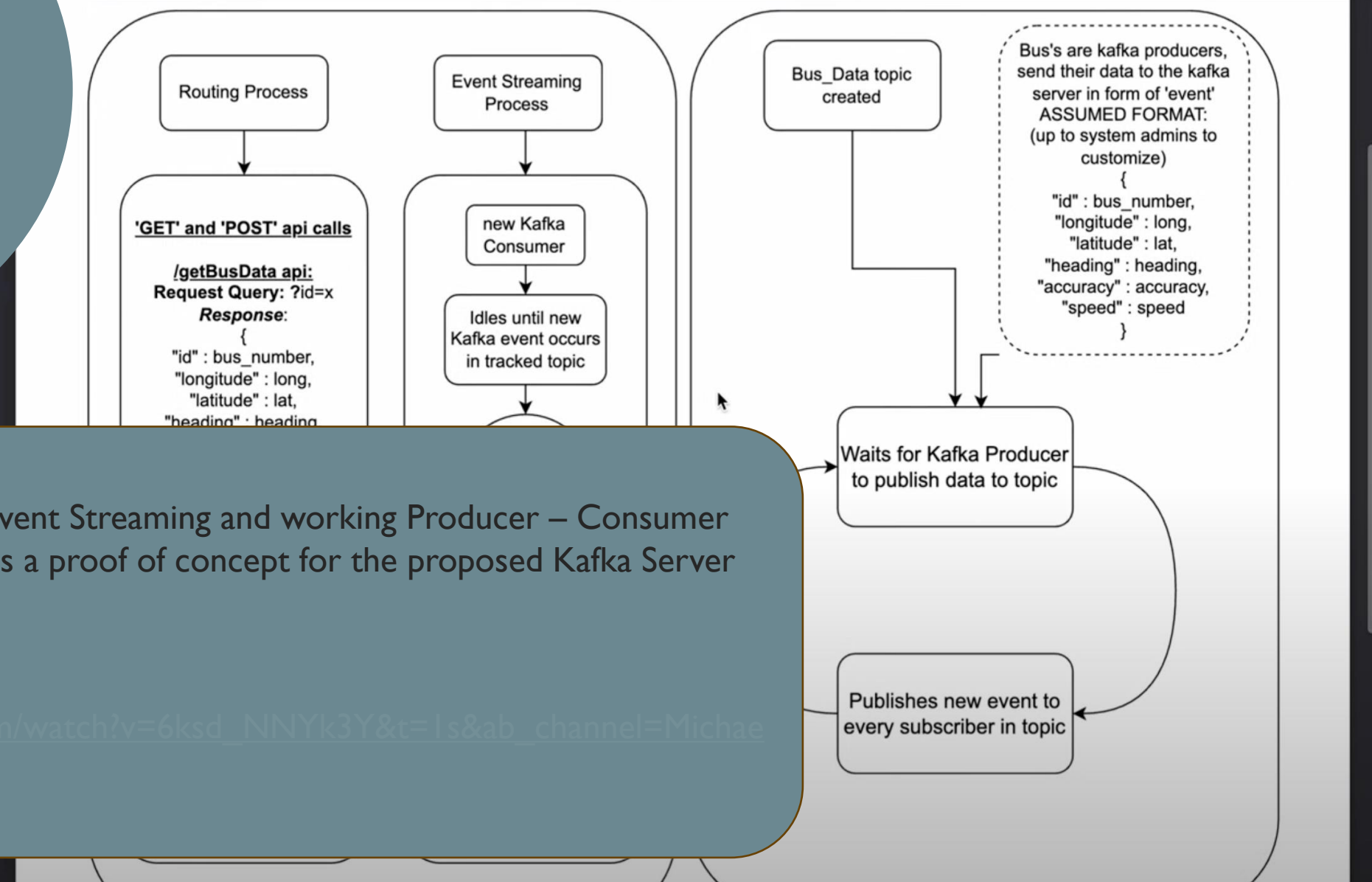
Automat disabled to manu the curre or to to

DATABASE SCHEMA

KAFKA DEMO

- A quick demo of Kafka Event Streaming and working Producer – Consumer data transfer. This demo is a proof of concept for the proposed Kafka Server functionality.

- https://www.youtube.com/watch?v=6ksd_NNYk3Y&t=1s&ab_channel=MichaelRizig



DATABASE DEMO

- A quick demo of database insertion, conditional queries with Microsoft SQL, as well as full stack communication via HTTPS requests with params for updated database records.

- https://www.youtube.com/watch?v=haKVglFf_5U&ab_channel=MichaelRizig

The screenshot displays a web application development environment. On the left, a map shows a city street grid with a yellow bus icon. The central part of the screen is divided into two main sections. The top section shows the 'EXPLORER' panel with a file tree containing files like 'main.py', 'main.js', 'insertDemo.py', 'DataMan...', 'static', 'css', 'js', 'res', 'templates', 'index.html', 'DatabaseSchema.png', 'insertDemo.py', 'main.py', 'README.md', 'Server_Process_Des...', 'Back-End', 'DataGen', 'DataManager', '__pycache__', 'DataManag...', 'ExceptionHandling', and 'consumer.py'. The bottom section shows the 'CODE EDITOR' with a file named 'insertDemo.py' open, displaying Python code for a REST API endpoint. The code includes comments, imports, endpoint definition, data formatting, and a POST request. The bottom right section shows the 'TERMINAL' panel with a series of GET requests and responses, indicating successful data retrieval from the API.

```
1 # this file is a demo of how to insert data into the database via our server
2
3 #import requests
4 import requests
5
6 # save api endpoint (update with used port)
7 ENDPOINT = 'http://localhost:3000/setBusData'
8
9 #format incoming data as json based on formatting instructions (see README)
10 busdata = {}
11     "id" : 1,
12     "latitude" : 37.8191,
13     "longitude" : -84.0155,
14     "heading" : 90,
15     "accuracy" : 20,
16     "speed" : 55
17
18
19 post = requests.post(url=ENDPOINT,json = busdata)
20
21 response = post.text
22
23 print(response)
24
25
```

```
(2024, 9, 12, 13, 8, 25, 443000), 33.9191, -84.0155, 90, 100, 55, 1)
Sep/2024 13:30:45] "GET /getBusData?id=1 HTTP/1.1" 200 -
Bus # 1
(2024, 9, 12, 13, 8, 25, 443000), 33.9191, -84.0155, 90, 100, 55, 1)
Sep/2024 13:30:48] "GET /getBusData?id=1 HTTP/1.1" 200 -
Bus # 1
(2024, 9, 12, 13, 8, 25, 443000), 33.9191, -84.0155, 90, 100, 55, 1)
Sep/2024 13:30:51] "GET /getBusData?id=1 HTTP/1.1" 200 -
Bus # 1
(2024, 9, 12, 13, 8, 25, 443000), 33.9191, -84.0155, 90, 100, 55, 1)
Sep/2024 13:30:54] "GET /getBusData?id=1 HTTP/1.1" 200 -
Bus # 1
(2024, 9, 12, 13, 8, 25, 443000), 33.9191, -84.0155, 90, 100, 55, 1)
Sep/2024 13:30:57] "GET /getBusData?id=1 HTTP/1.1" 200 -
Bus # 1
(2024, 9, 12, 13, 8, 25, 443000), 33.9191, -84.0155, 90, 100, 55, 1)
Sep/2024 13:31:00] "GET /getBusData?id=1 HTTP/1.1" 200 -
Bus # 1
(2024, 9, 12, 13, 8, 25, 443000), 33.9191, -84.0155, 90, 100, 55, 1)
Sep/2024 13:31:03] "GET /getBusData?id=1 HTTP/1.1" 200 -
Bus # 1
me(2024, 9, 12, 13, 8, 25, 443000), 33.9191, -84.0155, 90, 100, 55, 1)
[12/Sep/2024 13:31:06] "GET /getBusData?id=1 HTTP/1.1" 200 -
```