

# Report “Microservices\_Basics”

**Student Name:** Denis Humeniuk

**Date:** 17.02.2025

---

## Overview

### FacadeService

- Acts as an entry point for clients
- Provides an API to send messages ( `POST /api/Facade/send` )
- Fetches logs and messages from `LoggingService` and `MessagesService` ( `GET /api/Facade/fetch` )
- Uses `HttpClientFactory` to send HTTP requests to `LoggingService` and `MessagesService`

### LoggingService

- Handles logging of messages received from `FacadeService`
- Implements deduplication to avoid duplicate log entries
- Exposes two endpoints:
  - `POST /api/Logging/log` to store messages
  - `GET /api/Logging/logs` to retrieve logs

### MessagesService

- Currently serves as a placeholder for future message storage and retrieval functionality
- Exposes one endpoint:
  - `GET /api/Messages/message` returning a placeholder response

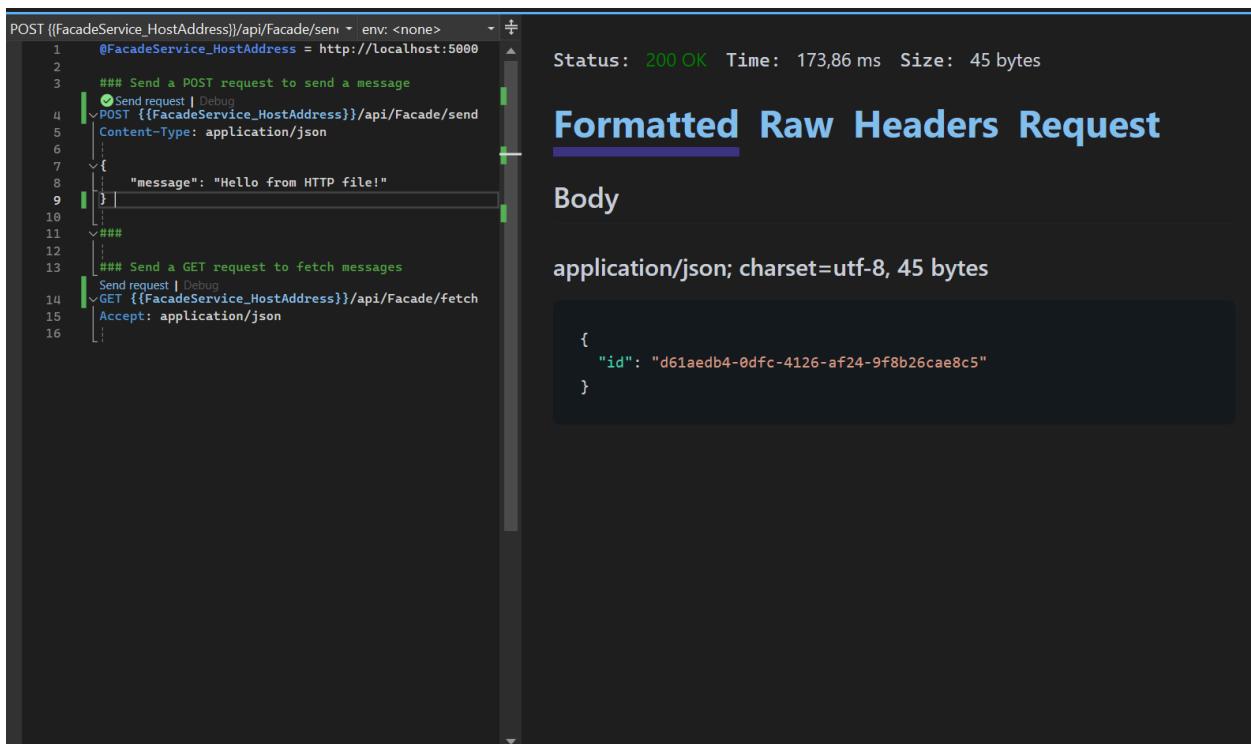
# Implemented additional tasks

## Retry Mechanism

- Implemented using **Polly library**
- If `LoggingService` is unavailable, `FacadeService` retries the request up to 3 times with exponential backoff

## Photos

### 1. Using `POST /api/Facade/send`



```
POST {{FacadeService_HostAddress}}/api/Facade/send env: <none>
1   @FacadeService_HostAddress = http://localhost:5000
2
3   ### Send a POST request to send a message
4   POST {{FacadeService_HostAddress}}/api/Facade/send
5   Content-Type: application/json
6
7   {
8     "message": "Hello from HTTP file!"
9   }
10
11  ###
12
13  ### Send a GET request to fetch messages
14  GET {{FacadeService_HostAddress}}/api/Facade/fetch
15  Accept: application/json
16
```

Status: 200 OK Time: 173,86 ms Size: 45 bytes

### Formatted Raw Headers Request

#### Body

application/json; charset=utf-8, 45 bytes

```
{
  "id": "d61aedb4-0dfc-4126-af24-9f8b26cae8c5"
}
```

FacadeService output:

```
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\Users\Admin\source\repos\microservices\MicroservicesSolution\FacadeService
warn: Microsoft.AspNetCore.HttpsPolicy.HttpsRedirectionMiddleware[3]
      Failed to determine the https port for redirect.
[FacadeService] Sending message to LoggingService: Hello from HTTP file!
info: System.Net.Http.HttpClient.LoggingServiceClient.LogicalHandler[100]
      Start processing HTTP request POST http://localhost:5001/api/Logging/log
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[100]
      Sending HTTP request POST http://localhost:5001/api/Logging/log
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[101]
      Received HTTP response headers after 55.914ms - 200
info: System.Net.Http.HttpClient.LoggingServiceClient.LogicalHandler[101]
      End processing HTTP request after 78.1216ms - 200
[FacadeService] LoggingService Response: OK
```

## LoggingService output:

```
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\Users\Admin\source\repos\microservices\MicroservicesSolution\LoggingService
warn: Microsoft.AspNetCore.HttpsPolicy.HttpsRedirectionMiddleware[3]
      Failed to determine the https port for redirect.
[LoggingService] New log received. ID: d01aedb4-0dfc-4126-af24-9f8b26cae8c5, Message: Hello from HTTP file!
```

## 2. Using [GET /api/Facade/fetch](#)

```

POST {{FacadeService_HostAddress}}/api/Facade/send - env: <none>
1   @FacadeService_HostAddress = http://localhost:5000
2
3   ### Send a POST request to send a message
4   ↘Send request | Debug
5   ✓POST {{FacadeService_HostAddress}}/api/Facade/send
6   Content-Type: application/json
7
8   {
9     "message": "Hello from HTTP file!"
10    }
11
12  ###
13  ### Send a GET request to fetch messages
14  ↘Send request | Debug
15  ✓GET {{FacadeService_HostAddress}}/api/Facade/fetch
16  Accept: application/json

```

Status: 200 OK Time: 127,14 ms Size: 47 bytes

## Formatted Raw Headers Request

### Body

application/json; charset=utf-8, 47 bytes

```
"Hello from HTTP file!\nNo messages available."
```

FacadeService output:

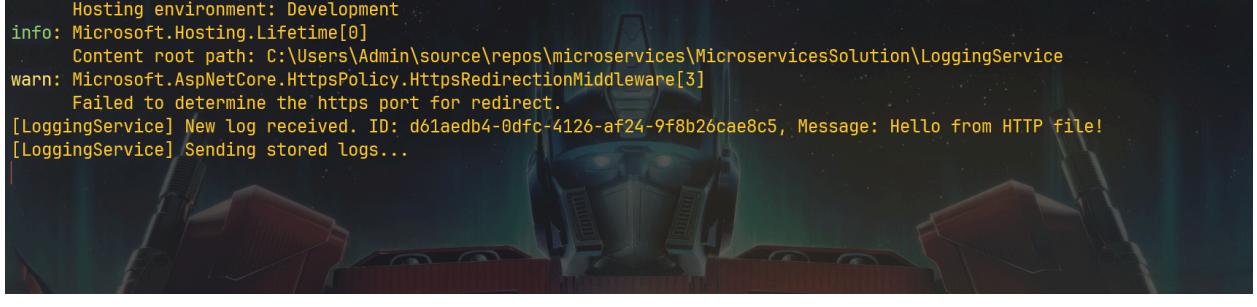
```

[FacadeService] Sending message to LoggingService: Hello from HTTP file!
info: System.Net.Http.HttpClient.LoggingServiceClient.LogicalHandler[100]
  Start processing HTTP request POST http://localhost:5001/api/Logging/log
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[100]
  Sending HTTP request POST http://localhost:5001/api/Logging/log
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[101]
  Received HTTP response headers after 55.9144ms - 200
info: System.Net.Http.HttpClient.LoggingServiceClient.LogicalHandler[101]
  End processing HTTP request after 78.1210ms - 200
[FacadeService] LoggingService Response: OK
[FacadeService] Fetching messages...
info: System.Net.Http.HttpClient.LoggingServiceClient.LogicalHandler[100]
  Start processing HTTP request GET http://localhost:5001/api/Logging/logs
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[100]
  Sending HTTP request GET http://localhost:5001/api/Logging/Logs
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[101]
  Received HTTP response headers after 30.6828ms - 200
info: System.Net.Http.HttpClient.LoggingServiceClient.LogicalHandler[101]
  End processing HTTP request after 40.9458ms - 200
[FacadeService] Received Logs: Hello from HTTP file!
info: System.Net.Http.HttpClient.LoggingServiceClient.LogicalHandler[100]
  Start processing HTTP request GET http://localhost:5002/api/Messages/message
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[100]
  Sending HTTP request GET http://localhost:5002/api/Messages/message
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[101]
  Received HTTP response headers after 43.88ms - 200
info: System.Net.Http.HttpClient.LoggingServiceClient.LogicalHandler[101]
  End processing HTTP request after 48.5917ms - 200
[FacadeService] Received Message from MessagesService: No messages available.

```

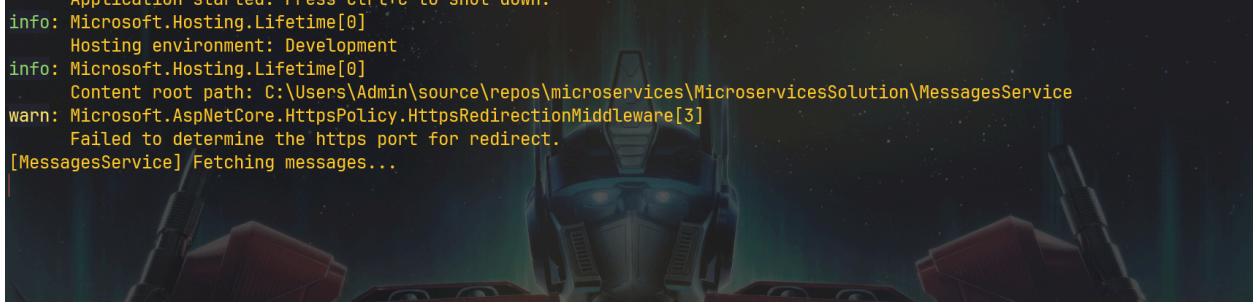
LoggingService output:

```
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\Users\Admin\source\repos\microservices\MicroservicesSolution\LoggingService
warn: Microsoft.AspNetCore.HttpsPolicy.HttpsRedirectionMiddleware[3]
      Failed to determine the https port for redirect.
[LoggingService] New log received. ID: d61aedb4-0dfc-4126-af24-9f8b26cae8c5, Message: Hello from HTTP file!
[LoggingService] Sending stored logs...
```



## MessagesService output:

```
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5002
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\Users\Admin\source\repos\microservices\MicroservicesSolution\MessagesService
warn: Microsoft.AspNetCore.HttpsPolicy.HttpsRedirectionMiddleware[3]
      Failed to determine the https port for redirect.
[MessagesService] Fetching messages...
```



### 3. Using `POST /api/Facade/send` when the LoggingService is down

POST {{FacadeService\_HostAddress}}/api/Facade/send env: <none>

```

1  @FacadeService_HostAddress = http://localhost:5000
2
3  ### Send a POST request to send a message
4  ⚡Send request | Debug
5  ✓POST {{FacadeService_HostAddress}}/api/Facade/send
6  Content-Type: application/json
7  :
8  {
9    "message": "Hello from HTTP file!"
10   }
11  :
12  ###
13  ### Send a GET request to fetch messages
14  ⚡Send request | Debug
15  ✓GET {{FacadeService_HostAddress}}/api/Facade/fetch
16  Accept: application/json

```

Status: 500 Internal Server Error Time: 31950,26 ms Size: 6217 bytes

## Formatted Raw Headers Request

### Body

text/plain; charset=utf-8, 6217 bytes

```

System.Net.Http.HttpRequestException: No connection could be made because
---> System.Net.Sockets.SocketException (10061): No connection could be
     made because the target machine actively refused it.
     at System.Net.Sockets.Socket.AwaitableSocketAsyncEventArgs.ThrowException()
     at System.Net.Sockets.Socket.<>ConnectAsync>g__WaitForConnectWithCancellation|1
     at System.Net.Http.HttpConnectionPool.ConnectToTcpHostAsync(String hostName,
     String portNumber)
     --- End of inner exception stack trace ---
     at System.Net.Http.HttpConnectionPool.ConnectToTcpHostAsync(String hostName,
     String portNumber)
     at System.Net.Http.HttpConnectionPool.ConnectAsync(HttpRequestMessage
     request)
     at System.Net.Http.HttpConnectionPool.CreateHttp1ConnectionAsync(IHttp
     connectionHandler)
     at System.Net.Http.HttpConnectionPool.AddHttp1ConnectionAsync(Quarantine
     entry)
     at System.Threading.Tasks.TaskCompletionSourceWithCancellation`1.W
     at System.Net.Http.HttpConnectionPool.HttpConnectionWaiter`1.WaitFor
     Connection()
     at System.Net.Http.HttpConnectionPool.SendWithVersionDetectionAndRe
     try(QuarantineEntry entry)
     at System.Net.Http.DiagnosticsHandler.SendAsyncCore(HttpContext context)
     at System.Net.Http.RedirectHandler.SendAsync(HttpContext context)
     at Microsoft.Extensions.Http.Logging.LoggingHttpMessageHandler.<Send>
     async(HttpContext context, HttpRequestMessage request, HttpResponseMes
     sage response)
     at Microsoft.Extensions.Http.Policy.HttpMessageHandler.SendCoreAsyn

```

## FacadeService output:

```

info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\Users\Admin\source\repos\microservices\MicroservicesSolution\FacadeService
warn: Microsoft.AspNetCore.HttpsPolicy.HttpsRedirectionMiddleware[3]
      Failed to determine the https port for redirect.
[FacadeService] Sending message to LoggingService: Hello from HTTP file!
info: System.Net.Http.HttpClient.LoggingServiceClient.LogicalHandler[100]
      Start processing HTTP request POST http://localhost:5001/api/Logging/log
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[100]
      Sending HTTP request POST http://localhost:5001/api/Logging/log
[FacadeService] Retry attempt 1 due to No connection could be made because the target machine actively refused it. (localhost:5001)
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[100]
      Sending HTTP request POST http://localhost:5001/api/Logging/log
[FacadeService] Retry attempt 2 due to No connection could be made because the target machine actively refused it. (localhost:5001)
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[100]
      Sending HTTP request POST http://localhost:5001/api/Logging/log
[FacadeService] Retry attempt 3 due to No connection could be made because the target machine actively refused it. (localhost:5001)
info: System.Net.Http.HttpClient.LoggingServiceClient.ClientHandler[100]
      Sending HTTP request POST http://localhost:5001/api/Logging/log
fail: Microsoft.AspNetCore.Diagnostics.DeveloperExceptionPageMiddleware[1]
      An unhandled exception has occurred while executing the request.
      System.Net.Http.HttpRequestException: No connection could be made because the target machine actively refused it.

```