



**VILNIUS UNIVERSITY  
ŠIAULIAI ACADEMY**

**BACHELOR PROGRAMME SOFTWARE ENGINEERING**

# **Object-Oriented Programming**

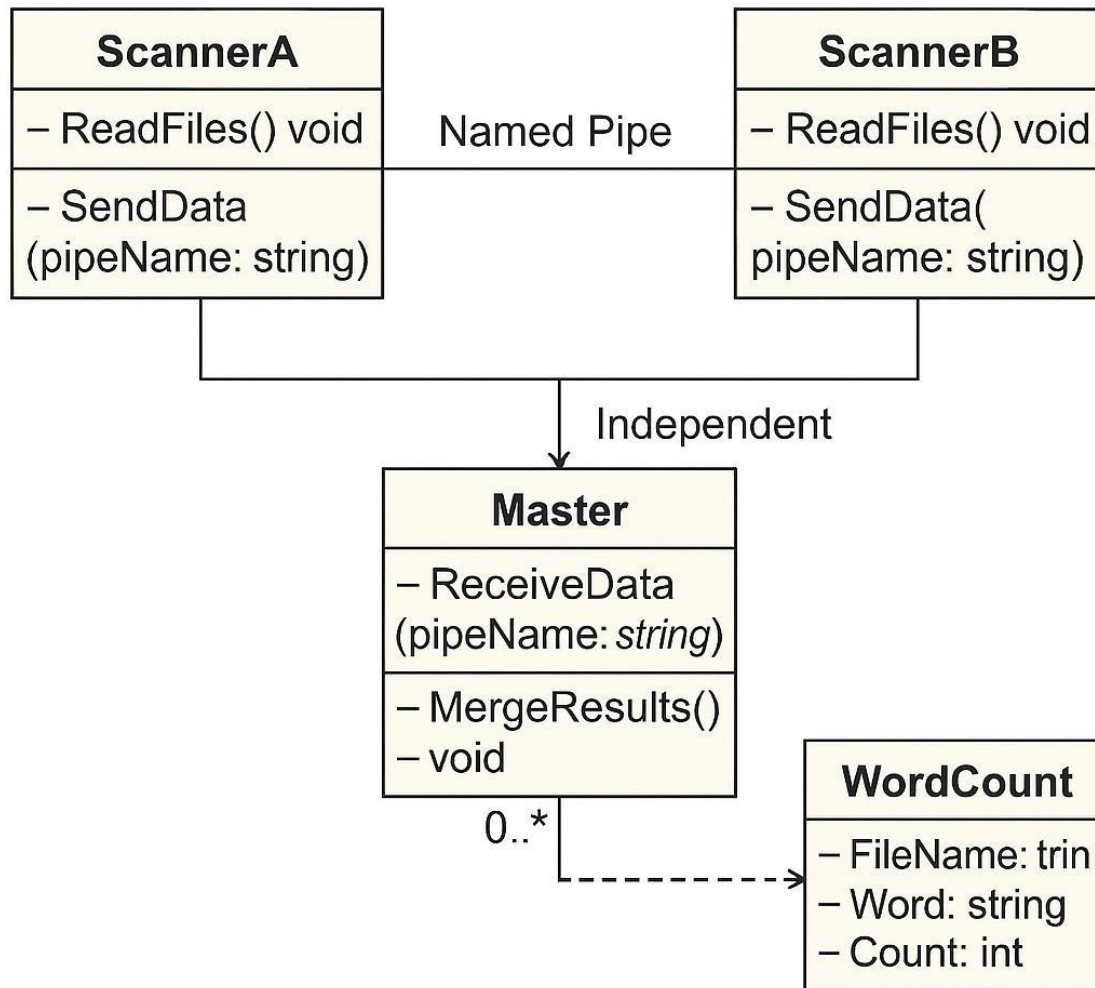
## **C#final task**

**student : Aayusha kattel**

**Leacture:DonatasDervinis**

**Šiauliai, 2025**

## UML diagram:



Git hub link: <https://github.com/ayusha5600/c-final-task>

### 1. What I Implemented

- Developed three separate C# console applications: **ScannerA**, **ScannerB**, and **Master**.
- **ScannerA** and **ScannerB** read .txt files from different folders and count how many times each word appears.
- Used a custom **WordCount** class that holds **FileName**, **Word**, and **Count** for each entry.

- **Used Named Pipes for communication:** ScannerA sends data through agent1, ScannerB through agent2.
  - **The Master program waits for both scanners, receives the data, merges them, and displays the final word count report.**
  - **All programs use multithreading to perform file reading and communication in parallel.**
- 

## **2. Functionality of Each Task**

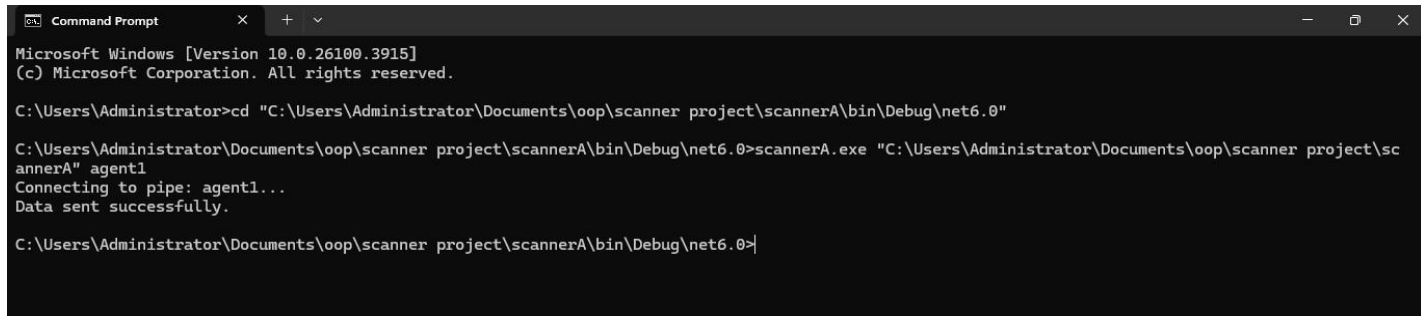
- **ScannerA:**
    - **Scans a folder for .txt files.**
    - **Reads content, counts each word, and sends a structured list of WordCount objects to Master via agent1.**
  - **ScannerB:**
    - **Similar to ScannerA, but uses a different folder.**
    - **Sends data via agent2.**
  - **Master:**
    - **Starts and waits for both agents to connect.**
    - **Receives and merges both word count lists.**
    - **Displays combined results with file name, word, and frequency.**
- 

## **3. Challenges Faced**

- **Pipe Busy Error:**
  - **Caused when Master was restarted while previous pipes were still open.**
  - **Fixed by closing running processes using Task Manager.**
- **Directory Not Found:**
  - **Happened when the given path was incorrect or the folder did not exist.**
  - **Solved by checking folder name carefully and quoting the path.**
- **Spaces in File Path:**
  - **Spaces in folder names caused command-line errors.**
  - **Solved by using double quotes .**

## Screenshoot:

### 1.ScannerA:



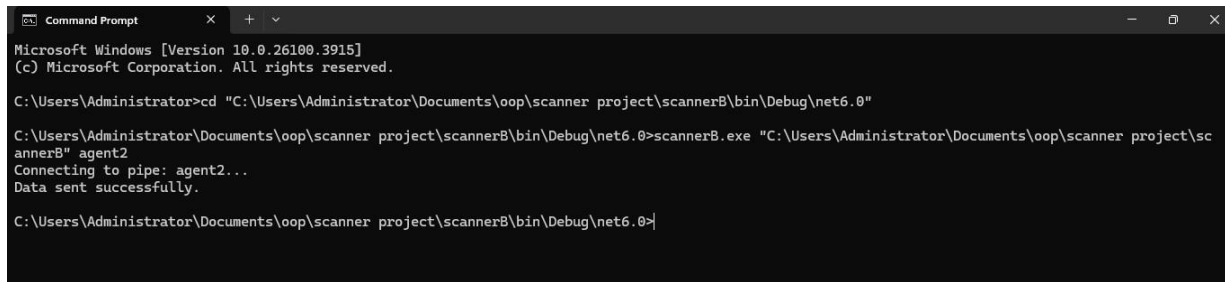
```
Microsoft Windows [Version 10.0.26100.3915]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator>cd "C:\Users\Administrator\Documents\oop\scanner project\scannerA\bin\Debug\net6.0"

C:\Users\Administrator\Documents\oop\scanner project\scannerA\bin\Debug\net6.0>scannerA.exe "C:\Users\Administrator\Documents\oop\scanner project\scannerA" agent1
Connecting to pipe: agent1...
Data sent successfully.

C:\Users\Administrator\Documents\oop\scanner project\scannerA\bin\Debug\net6.0>
```

### 2.ScannerB:



```
Microsoft Windows [Version 10.0.26100.3915]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator>cd "C:\Users\Administrator\Documents\oop\scanner project\scannerB\bin\Debug\net6.0"

C:\Users\Administrator\Documents\oop\scanner project\scannerB\bin\Debug\net6.0>scannerB.exe "C:\Users\Administrator\Documents\oop\scanner project\scannerB" agent2
Connecting to pipe: agent2...
Data sent successfully.

C:\Users\Administrator\Documents\oop\scanner project\scannerB\bin\Debug\net6.0>
```

### 3.Master:

```
Command Prompt
C:\Users\Administrator>cd "C:\Users\Administrator\Documents\oop\scanner project\master\bin\Debug\net6.0"
C:\Users\Administrator\Documents\oop\scanner project\master\bin\Debug\net6.0>master.exe agent1 agent2
Waiting for connection on pipe: agent1...
Waiting for connection on pipe: agent2...
Data received from agent2.
Data received from agent1.

--- Merged Word Count Results ---
scannerB.TXT:my:1
scannerB.TXT:surname:1
scannerB.TXT:is:1
scannerB.TXT:kattel:1
scannerB.TXT:and:1
scannerB.TXT:am:1
scannerB.TXT:very:1
scannerB.TXT:happy:1
scannerB.TXT:to:1
scannerB.TXT:see:1
scannerB.TXT:you:1
scannerB.TXT:all:1
scannerA.txt:my:1
scannerA.txt:name:1
scannerA.txt:is:1
scannerA.txt:aayusha:1
scannerA.txt:i:1
scannerA.txt:am:1
scannerA.txt:currently:1
scannerA.txt:studying:1
scannerA.txt:in:1
scannerA.txt:vilnius:1
scannerA.txt:university:1
```