Submission Worksheet

CLICK TO GRADE

https://learn.ethereallab.app/assignment/IT114-006-S2024/it114-project-milestone-1/grade/owe

IT114-006-S2024 - [IT114] Project Milestone 1

Submissions:

Submission Selection

1 Submission [active] 4/25/2024 11:38:30 PM

Instructions

COLLAPSE

Create a new branch called Milestone1
At the root of your repository create a folder called Project if one doesn't exist yet
    You will be updating this folder with new code as you do milestones
    You won't be creating separate folders for milestones; milestones are just branches
Create a pull request from Milestone1 to main (don't complete/merge it yet, just have it in open status)
Copy in the latest Socket sample code from the most recent Socket Part example of the lessons
    Recommended Part 5 (clients should be having names at this point and not ids)
    https://github.com/MattToegel/IT114/tree/Module5/Module5
Fix the package references at the top of each file (these are the only edits you should do at this point)
Git add/commit the baseline and push it to github
Create a pull request from Milestone1 to main (don't complete/merge it yet, just have it in open status)
Ensure the sample is working and fill in the below deliverables
    Note: The client commands likely are different in part 5 with the /name and /connect options instead of just "connect"
Generate the worksheet output file once done and add it to your local repository
Git add/commit/push all changes
Complete the pull request merge from step 7
Locally checkout main
git pull origin main

Branch name: Milestone1

Tasks: 9 Points: 10.00

Start Up (3 pts.)

COLLAPSE

COLLAPSE

Task #1 - Points: 1

Text: Server and Client Initialization

Checklist

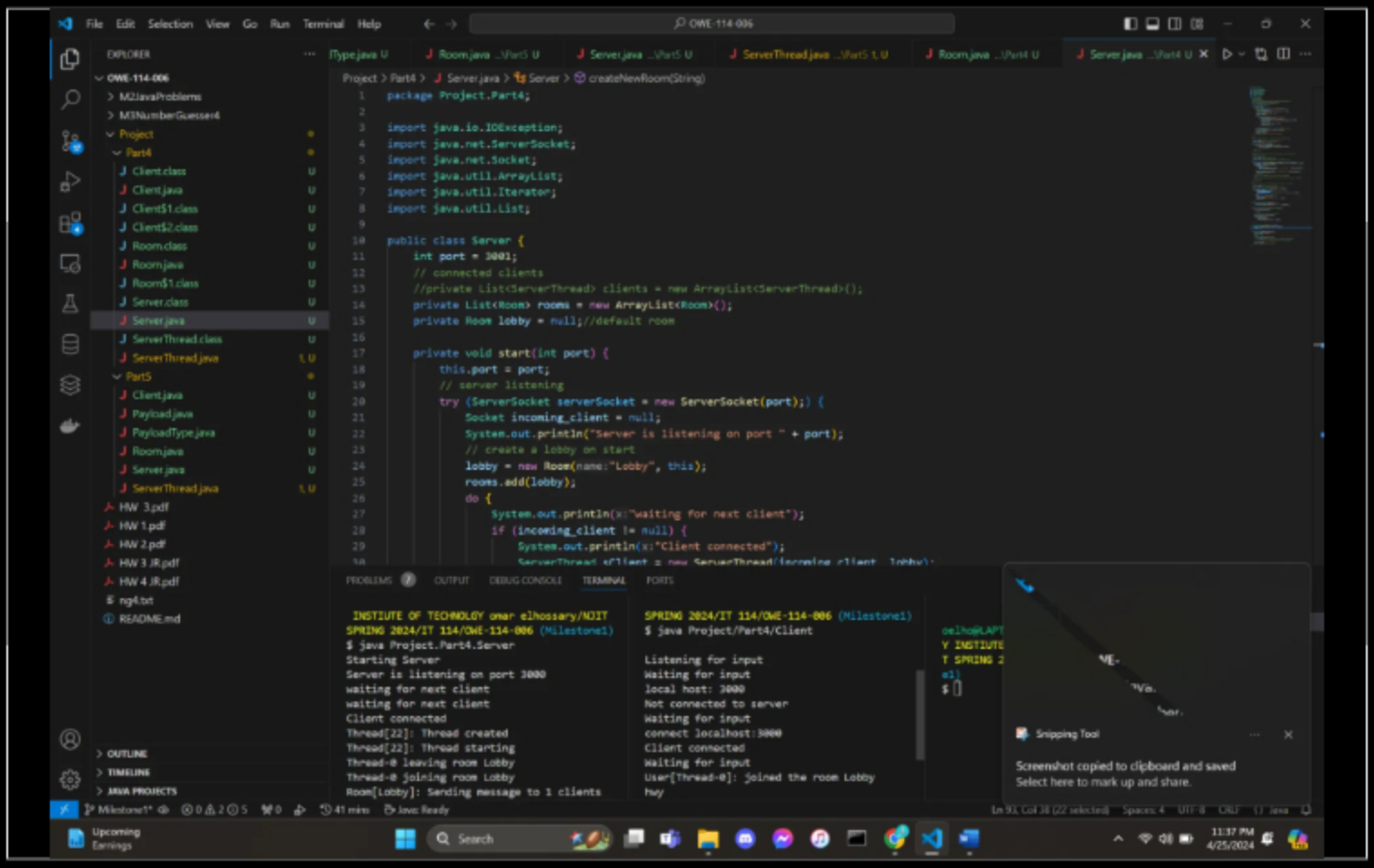
*The checkboxes are for your own tracking

#	Points	Details
☐ #1	1	Server should properly be listening to its port from the command line (note the related message)
☐ #2	1	Clients should be successfully waiting for input
☐ #3	1	Clients should have a name and successfully connected to the server (note related messages)

Task Screenshots:

Gallery Style: Large View

Small Medium Large



here is the server and client connected succesfully

Checklist Items (0)

COLLAPSE

Task #2 - Points: 1

Text: Explain the connection process

Details:

Note the various steps from the beginning to when the client is fully connected and able to communicate in the room.

Emphasize the code flow and the sockets usage.

Checklist

*The checkboxes are for your own tracking

#	Points	Details
☐ #1	1	Mention how the server-side of the connection works
☐ #2	1	Mention how the client-side of the connection works
☐ #3	1	Describe the socket steps until the server is waiting for messages from the client

Response:

A client attempts to connect to the server before attempting to communicate with it. Like the boss in the office, the server stands by and waits for people to come talk to it. The server responds to a client's attempt to connect by providing a dedicated line for communication and by saying, "Okay, come on in!" Once the client has this unique line, it can communicate with the server by sending messages and receiving messages in return. A phone call between the client and the server is similar to this special line. In summary, the server waits for clients to establish a connection before assigning them a dedicated channel to use for communication.

Communication (3 pts.)

COLLAPSE

COLLAPSE

Task #1 - Points: 1

Text: Add screenshot(s) showing evidence related to the checklist

Checklist

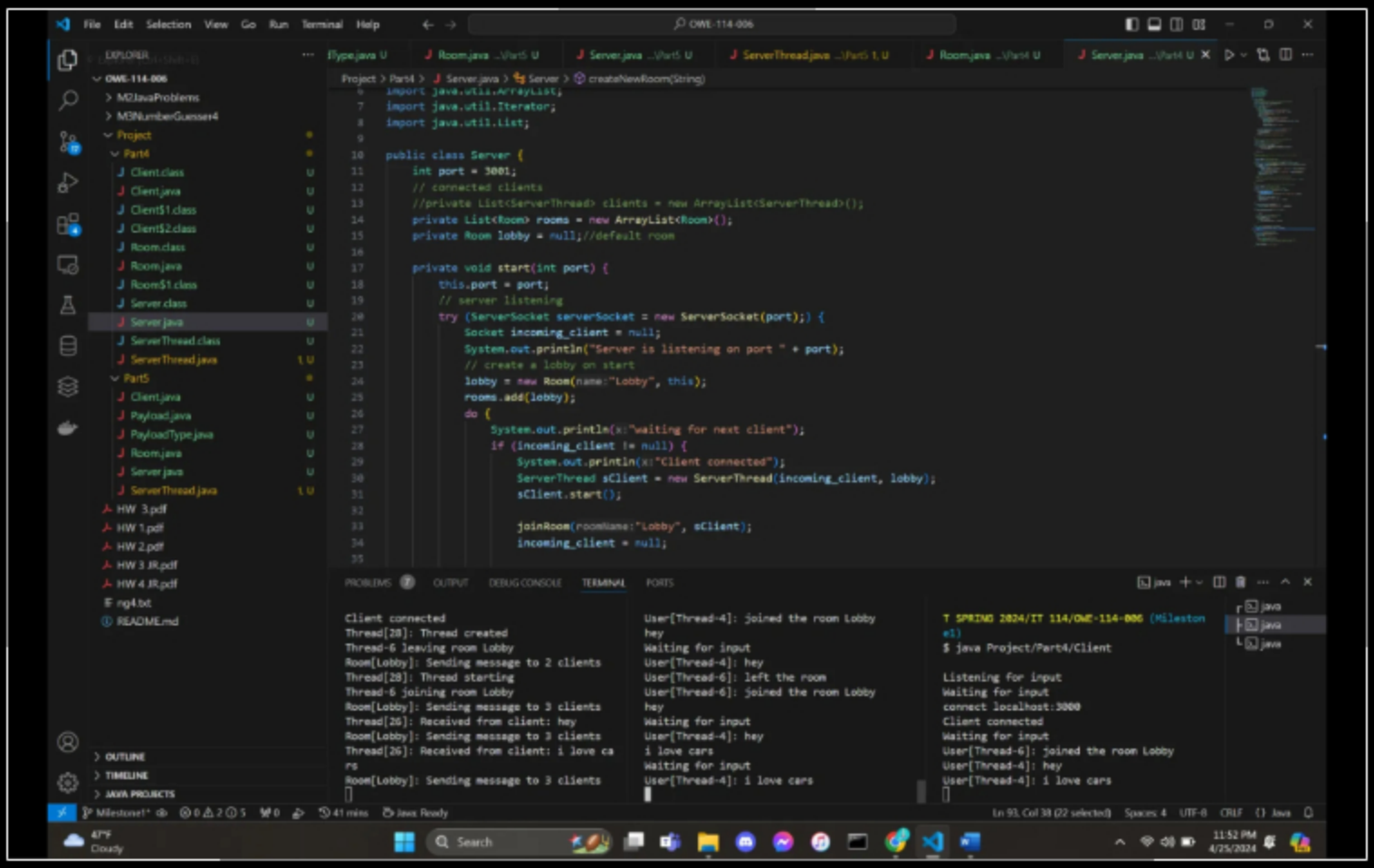
*The checkboxes are for your own tracking

#	Points	Details
☐ #1	1	At least two clients connected to the server
☐ #2	1	Client can send messages to the server
☐ #3	1	Server sends the message to all clients in the same room
☐ #4	1	Messages clearly show who the message is from (i.e., client name is clearly with the message)
☐ #5	2	Demonstrate clients in two different rooms can't send/receive messages to each other (clearly show the clients are in different rooms via the commands demonstrated in the lessons
☐ #6	1	Clearly caption each image regarding what is being shown

Task Screenshots:

Gallery Style: Large View

Small Medium Large



so here I have multiple clients connected and can see the messages

Checklist Items (0)

COLLAPSE

Task #2 - Points: 1

Text: Explain the communication process

Details:

How are messages entered from the client side and how do they propagate to other clients?

Note all the steps involved and use specific terminology from the code.
Don't just translate the code line-by-line to plain English, keep it concise.

Checklist *The checkboxes are for your own tracking

#	Points	Details
☐ #1	1	Mention the client-side (sending)
☐ #2	1	Mention the ServerThread's involvement
☐ #3	1	Mention the Room's perspective

#4	1	Mention the client-side (receiving)

Response:

A socket is a special line that is used to carry messages from a user's keyboard to the server when they type something and hit enter in the chat program. A ServerThread, akin to a server assistant, is in charge of this unique line. The message is received by the ServerThread and sent to the individual's room. After that, the room uses each person's unique line to send the message to the others in the room, displaying it on their screens. Messages are sent in this manner to every member of the chat room!

Disconnecting/Termination (3 pts.)

COLLAPSE

COLLAPSE

Task #1 - Points: 1

Text: Add screenshot(s) showing evidence related to the checklist

Checklist

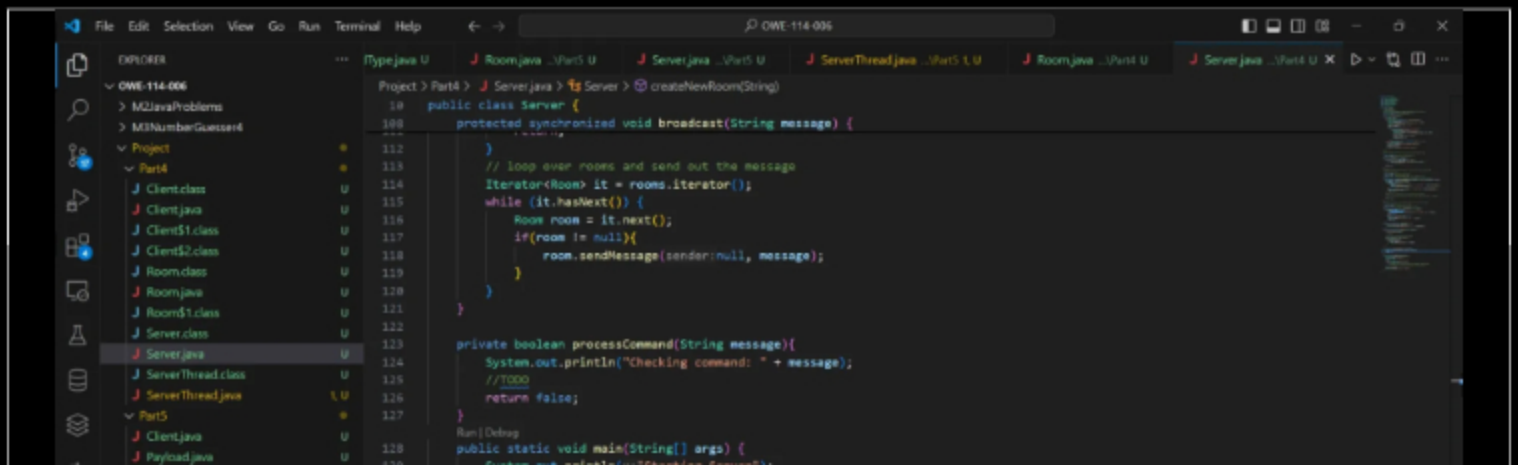
*The checkboxes are for your own tracking

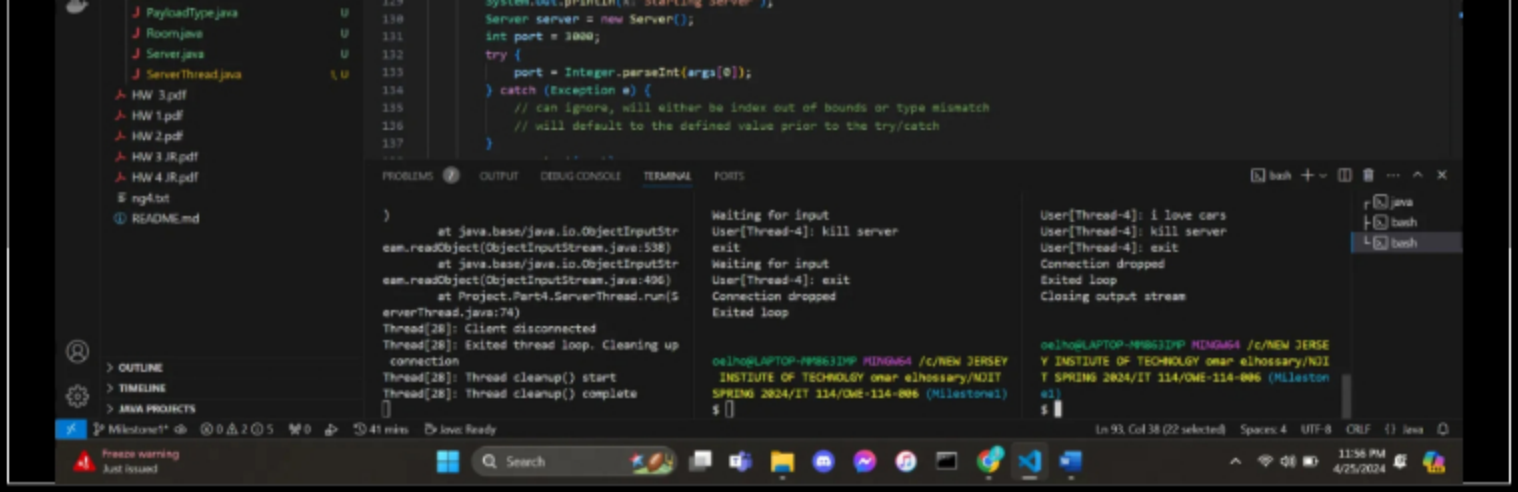
#	Points	Details
#1	1	Show a client disconnecting from the server; Server should still be running without issue (it's ok if an exception message shows as it's part of the lesson code, the server just shouldn't terminate)
#2	1	Show the server terminating; Clients should be disconnected but still running and able to reconnect when the server is back online (demonstrate this)
#3	1	For each scenario, disconnected messages should be shown to the clients (should show a different person disconnected and should show the specific client disconnected)
#4	1	Clearly caption each image regarding what is being shown

Task Screenshots:

Gallery Style: Large View

Small Medium Large





here I used cntrl+C to kill the server

Checklist Items (0)

COLLAPSE

Task #2 - Points: 1

Text: Explain the various Disconnect/termination scenarios

Details:
Include the various scenarios of how a disconnect can occur. There should be around 3 or so.

Checklist *The checkboxes are for your own tracking

#	Points	Details
☐ #1	1	Mention how a client gets disconnected from a Socket perspective
☐ #2	1	Mention how/why the client program doesn't crash when the server disconnects/terminates.
☐ #3	1	Mention how the server doesn't crash from the client(s) disconnecting

Response:

In the world of disconnect scenarios, users can disconnect themselves voluntarily by closing their browser or unintentionally by having a program crash or network outage. I in this example and screenshot I Showed I ended it on my own. Client programs are made to withstand these possible interruptions; in the unlikely event that the server goes down, they handle the situation calmly, providing users with options for reconnecting without crashing. In a similar vein, servers continue to function even in the event that clients disconnect. Servers maintain uninterrupted service for other clients by means of effective resource management and error handling mechanisms. In introductory courses, this dependability is fundamental, highlighting the significance of error control and strong design in networked applications.

Misc (1 pt.)

COLLAPSE

COLLAPSE

Task #1 - Points: 1

Text: Add the pull request link for this branch

URL #1

https://github.com/WayguBeef5/OWE-114-006/pull/8

COLLAPSE

Task #2 - Points: 1

Text: Talk about any issues or learnings during this assignment

Details:

Few related sentences about the Project/sockets topics

Response:

Thankfully did not have a lot off issues with the milestone. The previous lessons helped me to understand how the sever and client work. I did not do my java sockets homework so again thankfully to previous lessons and a bit of help it was not hard to pick up.

COLLAPSE

Task #3 - Points: 1

Text: WakaTime Screenshot

Details:

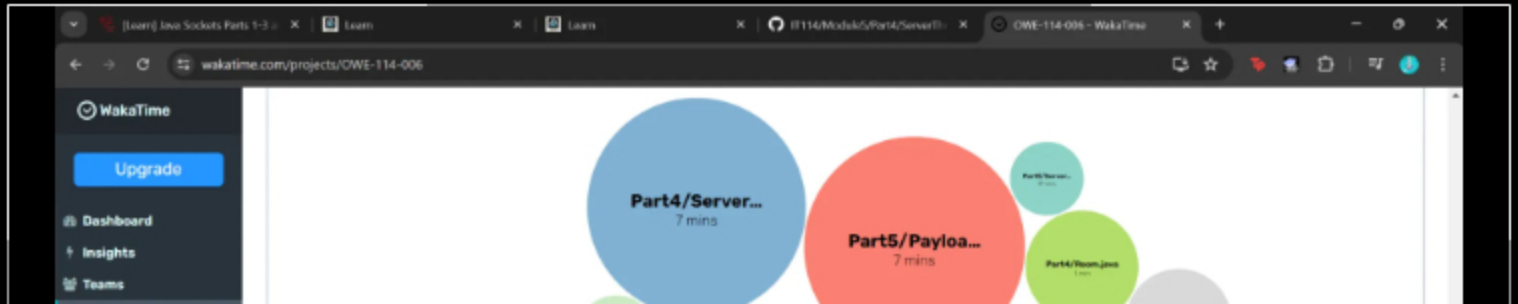
Grab a snippet showing the approximate time involved that clearly shows your repository.

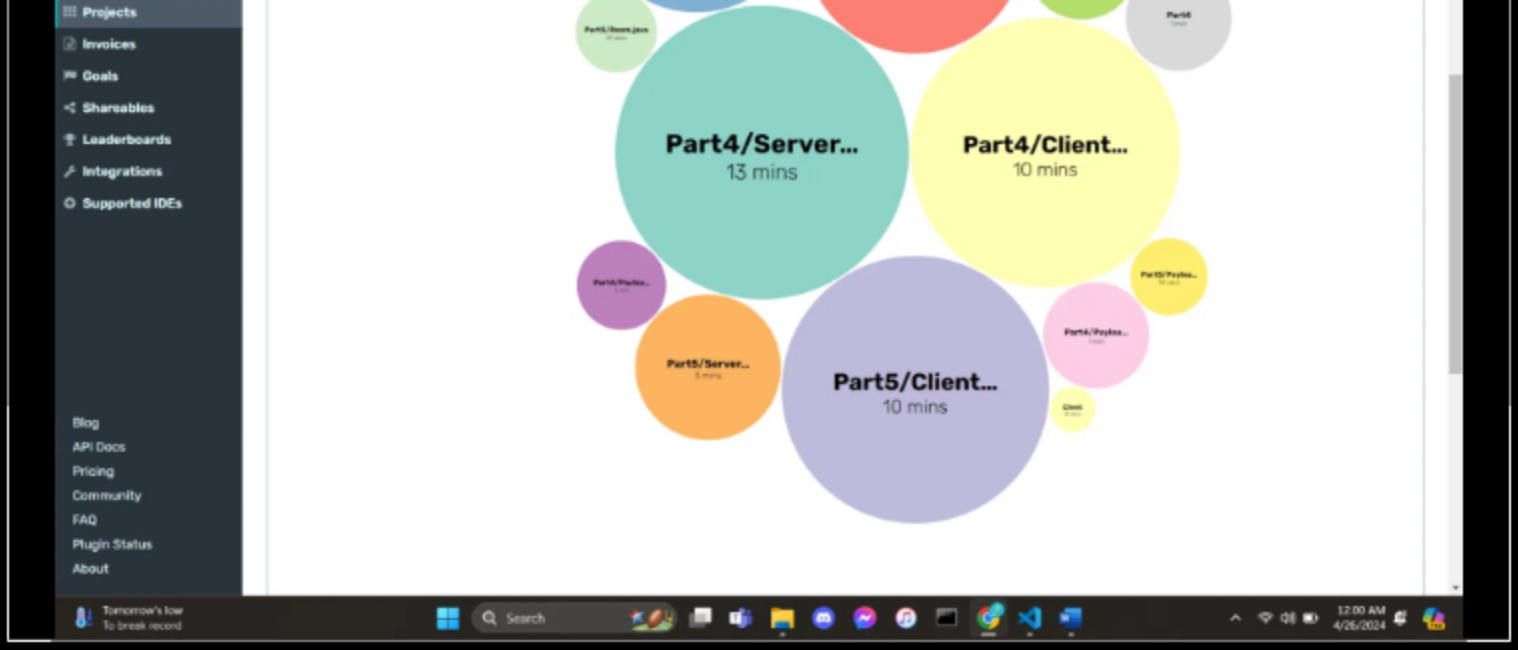
The duration isn't considered for grading, but there should be some time involved.

Task Screenshots:

Gallery Style: Large View

Small Medium Large





MY WAKA TIME

End of Assignment