

Low Level Design

Omer Abbas, Jordan Atanassov, Sebastian Bukvic, Reagan Gary, Jake Lehoullier

As stated, the architectural design pattern we will implement is the client-server architecture, which focuses on data-centric and control flow.

In FIGURE 1, we have the pseudo code of what the server should be doing. First it waits for the form to be sent, from there, it authenticates the connection, making sure that connection is a trusted IP address and is whitelisted on the domain of the project. From there, it tries to process the Json. If it does not process the Json successfully, then it will not return anything from the method, If it does process it successfully, the server will send an OK response to the connection, showing that the request was processed. If the user is not authenticated, then the server will notify the admin, and will store a log of the connection and all the information it can observe about that said connection. Further action will then need to be taken by the administration with the logs that the server stores.

In FIGURE 2, the client is the focus. From here, it is simple and lightweight. The client will send the form to the server's predetermined connection point. From there, it will check if the requests own form is valid in format. If it is, it will return the error code that the server sends. From there it will be processed by other methods and the request will be communicated back to the user upon successful submission.

In FIGURE 3, the class diagram reflecting the basic pseudo code of the combined FIGURE 2 and FIGURE 1 is used to create a better visual understanding of the relationship between the client and the server. The receives a connection and stores it into the whitelisted connections List, and uses those connections to do the operations that the server class has access to. The client, being much simpler since it's just sending raw data, receiving formatted data, and processing error codes to relay back to the client.

```
scheduledForm(connection e){
    Form f = awaitForm(e)
    if(authenticate(e)):
        if(jsonProcess(f)):
            self.send(e, OK)
            return(f)
        else:
            notifyServer(e)
            storeLog(e)
    return NULL
}
```

FIGURE 1

```
sendForm(Form f, connection server){
    if(validateFormat(f)):
        return(self.send(f, server, POST))
    else:
        return False
}
```

FIGURE 2

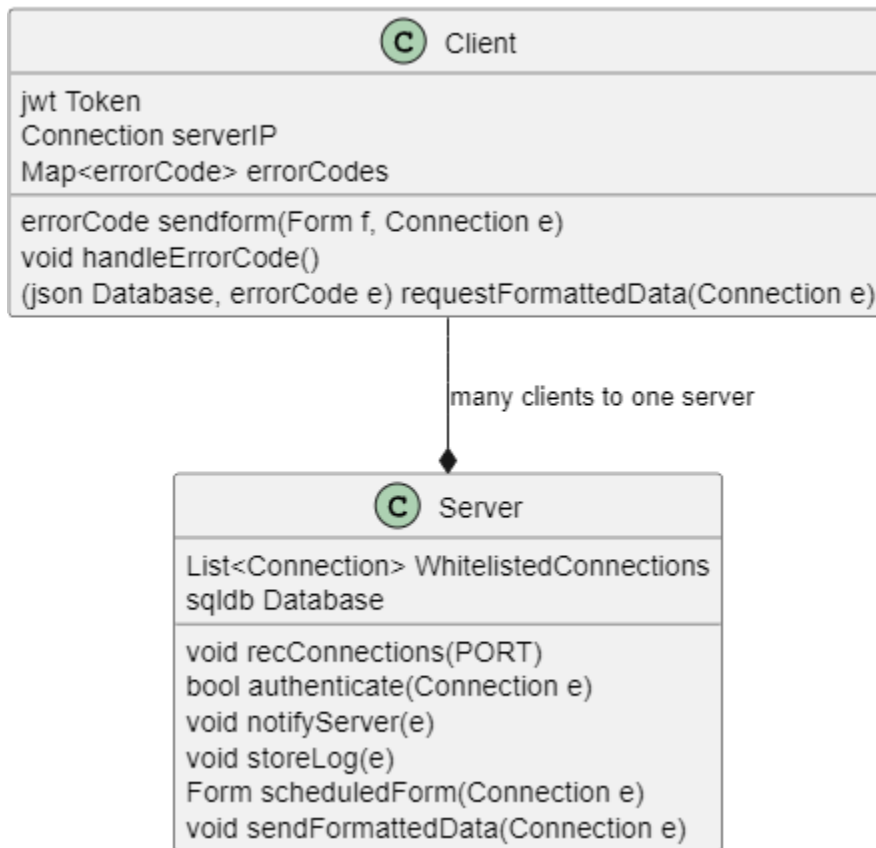


FIGURE 3