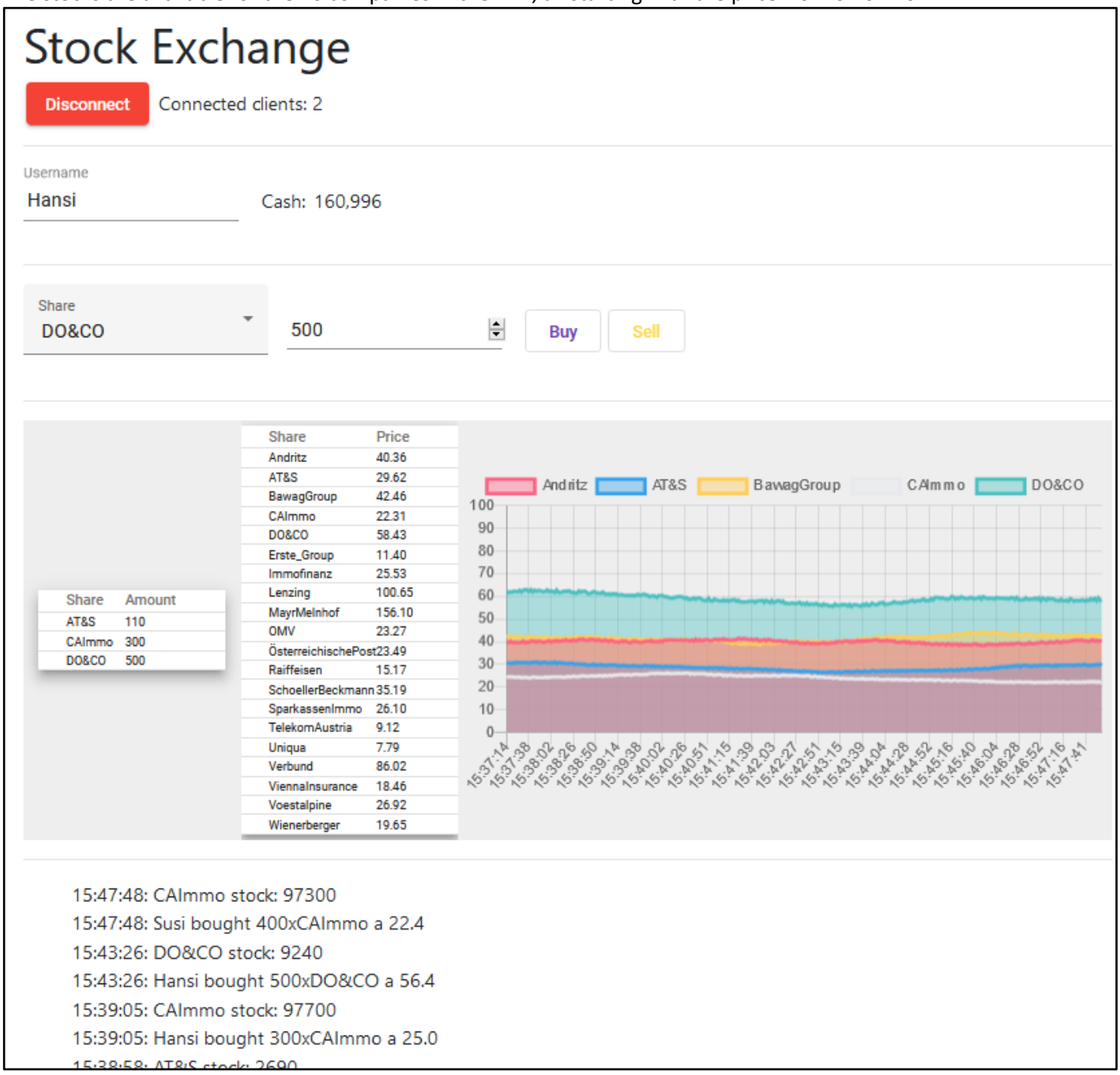
# ATX Stock Exchange with SignalR

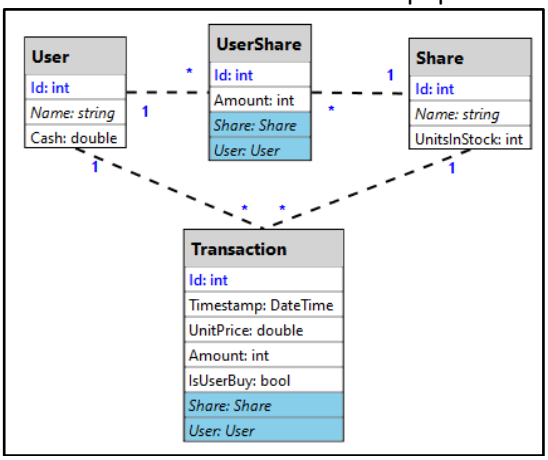
Program an Angular App to buy and sell shares. The current stock exchange prices are sent from the backend using SignalR. There are some predefined users in the database who all start with 200000€ cash.
The stocks are available for the 20 companies in the ATX, all starting with the price from 01.01.2021.




## General

The backend is attached as zip to this document. It already contains the following:

- "Database"-Entities in the folder Entities. These classes are not connected to a physical database. Only in the Context class the "tables" are populated with some dummy values




- The Service StockTickerService is available. It produces dummy values for each stock. It tries to produce reasonable curves in interpolating some fixed values with noisy cubic splines.
- Some DTOs are available. Feel free to adapt them for your needs.

## Backend

The following functionality shall be implemented:

- Create a Hub to communicate with the frontend
- Adapt Startup.cs as shown in the tutorial
- Use this Hub in StockTickerService to distribute the produced values to the clients
- Count the number of connected clients in the Hub (register it as Singleton!!!!!) and inform all clients with the current number whenever a client connects or disconnects
- Add methods BuyShare and SellShare in the Hub. Inform all clients with SendAsync("transactionReceived,transaction)
- Add two GET methods in StockController:
  - Get required data for one user
  - Get all shares from the database

## Frontend

### General

Try to place all code according the HubConnection in a service very much like you do with other REST-calls to the backend. Use one of the following two options (or perhaps find yourself a better solution):

Callback

```
onTransactionReceived(fct: (x: TransactionDto) => void): void {
  this.hubConnection.on('transactionReceived', fct);
}
```

Observable

```
onNewStocks(): Observable<ShareTickDto[]> {
  const subject = new Subject<ShareTickDto[]>();
  this.hubConnection.on('newStocks', x => subject.next(x));
  return subject.asObservable();
}
```

### Functionality

On the fronted display the following:

- Current cash of the displayed username

- Possibility to buy/sell an arbitrary amount of a specific share. Perhaps check if you have enough cash to do so. On the other hand, on the backend you should assert, that the required number of shares is available.
- Display the current stock exchange prices that are updated with SignalR.
- Display the price history of the first 5 stocks in a chart
- All the shares of the current user.
- All messages received from the backend with SignalR shall be logged at the bottom of the page