## Client + login(int userId, String Password): bool + logout(): void + searchAvailableSharesByName( String partOfStockName): List<Shares> + buyStock(int UserId, int StockId): void + sellStock(int UserId, int StockId): void + showDepot(int UserID): List<Stock> EmployeeClient + searchUserByUserId(int UserId): User + searchUserByLastname(String LastName): User + createCustomer(String firstName, String lastName, String Password) + createEmployee(String firstName, String lastName, String Password) + getInvestibleAmount(): float

+ logout(): void

## <<Interface>> Bankinterface Bank + login(int userId, String Password): bool + login(int userId, String Password): bool + logout(): void + searchAvailableSharesByName( String partOfStockName): List<Shares> + searchAvailableSharesByName( String partOfStockName): List<Shares> + buyStock(int UserId, int StockId): void + buyStock(int UserId, int StockId): void + sellStock(int UserId, int StockId): void + sellStock(int UserId, int StockId): void + showDepot(int UserID): List<Stock> + showDepot(int UserID): List<Stock> + searchUserByUserId(int UserId): User + searchUserByUserId(int UserId): User + searchUserByLastname(String LastName): User + searchUserByLastname(String LastName): User + createCustumer(String firstName, String lastName, String Password) + createCustumer(String firstName, String lastName, String Password) + createEmployee(String firstName, String lastName, String Password) + createEmployee(String firstName, String lastName, String Password) + getInvestibleAmount(): float + getInvestibleAmount(): float **Entities**

## TradingWebService

- + tradingWebService()
- + buy(String symbol, int shares)
- + findStockQuotesByCompanyName(String partOfCompanyName): List<PublicStockQuote>
- + getStockQuoteHistory(String symbol): List<PublicStockQuote>
- + getStockQuotes(List<String> symbols): List<PublicStockQuote>
- + sell(String symbol, int shares): BigDecimal