

Exception Handling,

Our implementation of stocktrader is designed for users to buy and sell investments in a simulated market. Brokers fill the market with bonds, stocks, and reits that account holders can purchase. Many transactions occur and, in each case, we need to ensure that each transaction occurs successfully and is accurately reflected to the user. Possible conditions that our software will have to handle include:

- ❖ Users failing to login within a certain amount of attempts
 - Login screen is loaded up. After three unsuccessful attempts, inform user of system unavailability and close login.
- ❖ Investment purchases by account holders
 - Account holders wishing to buy investments must have sufficient funds. We would check the database for current funds. If the value of the investment is greater than the current funds, a dialog will be generated reporting insufficient funds.
- ❖ Communication between the backend of the database and application model is dropped.
 - Detect that a response has timed out. Retry request after communication is restored. Report system unavailability to user and report backend request status. *The user will make a request to save their current market. Connection ~~is not~~ available will be checked in the technical services layer and a report will be passed back about system unavailability.*

- Situation when
 - x data inconsistency when showing data
 - Buying an investment when it no longer exist.
- Unrecoverable system errors, etc.
- These 3 does not seem enough.