CROSS | OVER

Preconditions

1. You should work on your local machine.
2. You may use Visual Studio (any version) or any other C++ editor/IDE.
3. You may use SQL Server (any version) (Main or Express Editions) or any other database.
4. You may use Casablanca C++ REST SDK or any other REST API.
5. You may use Wt, C++ Web Toolkit or any other similar framework.

Instructions

- Try to complete as much as possible within the given time frame. If you need more time, please ask for an extension. You must complete full-functionality of the application with industry-level coding style/commenting. Unfinished assignments will not be considered.

- Please note that you are expected to work on the assignment independently. Discussing assignment details with colleagues or any indication of outside help will be considered cheating.

- Please do not expect too much hand-holding as this is an evaluation assignment.

- Read the complete assignment before you start. Understand clearly what is required so that your work will be appropriate and easier.

Requirements

Objective

Create a simple RESTful web service and a simple web application to simulate a stock exchange and a trader terminal.

Functional Specifications

Create a high performance RESTful web service

1. 1. The service should expose the following operations. Each operation accepts username and password to authenticate the request.
      1. Quote (userName, password, stockCode) <<< gives the last sale price of a stock
      2. Sell (userName, password, stockCode, quantity, price) <<< sell stocks out of a trader portfolio, only portfolio quantity is allowed
      3. Buy (userName, password, stockCode, quantity, price) <<< buy stocks into a trader portfolio. Each trader can have a total of 100,000 dollars initially to invest
      4. RegisterTrader (userName, password) <<< used to register a trader username and password
      5. Transactions (userName, password) <<< transactions executed by a trader with status
      6. PortfolioList (userName, password) <<< list of a trader's balance stocks along with quantity, cost and current value

Create a web application for stock traders

1. 1. Ability to register and log in
   2. Ability to buy / sell / quote stocks
   3. Ability to see transactions log
   4. Ability to see trader's portfolio

Technical Specifications

The following list of technical specifications should be adhered to

1. 1. The service should be created using Casablanca C++ REST SDK.
      1. The service responds in JSON and could be hosted using any web server.
   2. Create a database with tables
      1. transaction: username, stockcode, quantity, datetime, status (executed, pending)
      2. trader: username, password, balancecash
      3. portfolio: username, stockcode, quantity, totalcost
      4. stock: stockcode, lastsaleprice
   3. The application should be published using Wt and could be hosted using any web server.
      1. The application should not access the database directly.
   4. Apply input validations and constraints wherever necessary to create a stable service and application.
   5. Even if you are not able to complete all the tasks, try to achieve a working application.
   6. Add missing requirements to the implementation, according to your experience.

Deliverables

Demonstration Video

Record the video demonstration of the system using a screen-cast tool like Wink (or any other tool) intermixed with the execution of all components. Do not upload the video anywhere. Save it to your local machine and include it with the delivery package.

Database scripts

Create steps or a single SQL script file to create the database, its schema and any test data you may use.

Readme document

Create a txt file with the following information

1. 1. Instructions to install and configure any pre-requisites and dependencies to prepare the development environment
   2. Instructions to create and initialize the databases (if any)
   3. Instructions to configure and prepare the source code to build and run properly
   4. Any assumptions made and missing requirements that are not covered in the specifications
   5. Any issues faced and any constructive feedback you may wish to give about improving the assignment

Design document

Create a design document containing the following (in not more than 5-7 pages)

1. 1. Explain the technologies and design patterns used and why
   2. Explain how the components are broken down and implemented technically
   3. A few UML diagrams showing the components interaction and data/control flow (component, activity, or sequence diagrams)

Source Code

You should deliver the implemented source code including any dependencies. For the dependencies that could not be included due to size, the readme file should have proper instructions on how to download and install them.

To be evaluated

1. The completeness of functional implementation as per specifications
2. The quality of output and implemented code
3. The quality of design and technologies used
4. Properly completed documentation and demonstration video
5. Extra validations and assumptions which are not described

Delivery / How to submit

Please read and follow this section carefully. Any delivery that does not follow this section will score less or simply won't be evaluated.

Create and submit an archive named **<your_name>_CppAssignment3.zip** containing the following

<your_name>_CppAssignment3.zip
<your_name>_CppAssignment3.zip \Readme.txt
<your_name>_CppAssignment3.zip \Design.doc
<your_name>_CppAssignment3.zip \Demo\ < this folder contains the screen video recording
<your_name>_CppAssignment3.zip \Source\ < this folder contains the complete source code

Check that the size of the archive is less than 30MB. If not, reduce the size of the demo video by removing similar frames and remove the binary dependencies.

ATTENTION! YOUR APPLICATION WILL BE REJECTED IF IT:

- Does not compile
- Does not contain unit tests
- Unit tests are failing