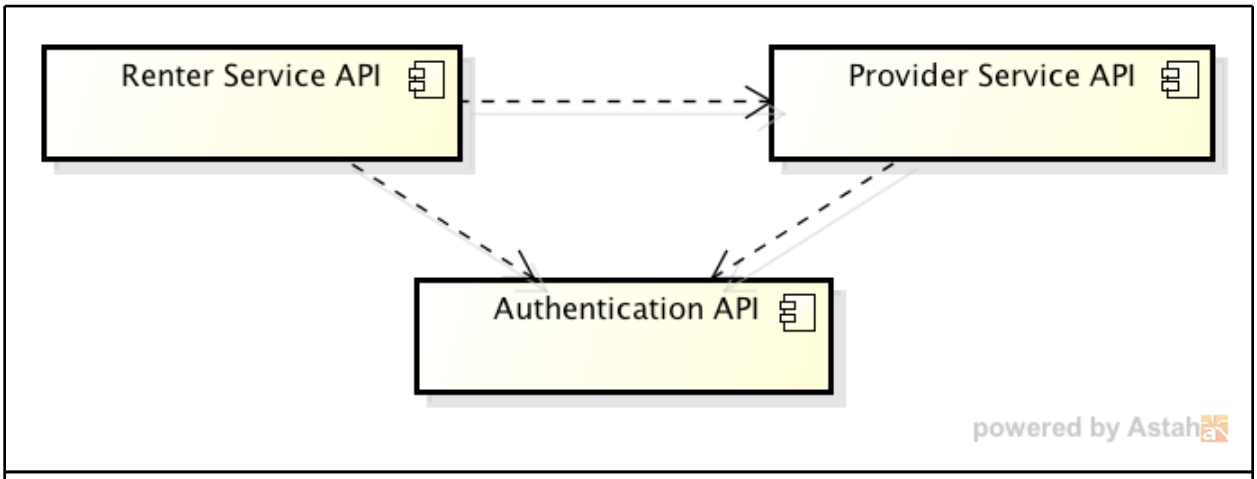
CSCI E-97
Assignment 4
Due: Thursday, 11/20/2014, 11:59pm EST

Introduction

In this assignment you will continue the development of the SquareDesk office space application. You will design and implement an Authentication Service.

Overview

The following diagram shows how the Authentication Service fits into the overall structure of the SquareDesk application.



Caption: SquareDesk modules and dependencies including Authentication API.

The Authentication Service supports controlling access to the SquareDesk application, specifically the restricted interfaces exposed by the Renter and Provider Services and the Authentication Service itself.

Please refer to the Authentication Service Requirements document for more details.

As part of your solution, you should apply the Visitor pattern to support traversing the objects of the Authentication Service with the following goals:

Provide an inventory of all Users, Groups, Services, Methods, Roles, and Permissions.

Apply the Factory Method pattern to create all instances of the various classes where appropriate.

Use the Singleton pattern to return a pointer to an implementation of the Authentication Service.

In the design portion of the assignment, you will create a software design document that satisfies the Authentication Service requirements.

In the implementation portion of the assignment, implement your design and test your solution.

You will have 2 documents as input to your design:

- Authentication Service Requirements Document describing the functional requirements.
- Software design template (from assignment 2) as a base for your design.

Development Process

This will be the 3rd of 3 sprints to implement the SquareDesk application. We will continue to follow the development process outlined in assignment 2.

For this assignment, a peer design review is required. New peer design review partners will be assigned. If you have any questions regarding the peer design reviews, please contact the teaching staff. Please work with your partner to complete the design reviews by Monday, November 10th. This should provide enough time to incorporate the comments from the design review and complete the implementation before the due date, Thursday, November 20th.

Design reviews can be conducted via:

- email
- screen sharing
- in person
- google docs
- other

Assignment Notes:

The goal of this assignment is to design and implement within the context of a collaborative agile development environment.

Reuse the design template from assignment 2. Your design document should include the following:

- UML Use Case Diagram (with descriptions for each use case)

- UML Class Diagram
- Class Dictionary
- UML Sequence Diagram(s) (showing the flow of events for checking Access)
- Activity Diagram documenting how the permissions, roles, and users are created.

You should implement the Authentication Service classes as defined by the class diagram and class dictionary specified in your design document. All authentication classes should be defined within the package “cscie97.asn4.squaredesk.authentication”.

Reuse your Provider Service API and Renter Service API from assignments 2 and 3.

Update the implementations of Provider Sevice and Renter Service restricted methods to delegate to the Authentication Service checkAccess method. Update the calling methods to handle a possible AccessDeniedException.

Reuse your TestDriver class from assignment 3 to load in the Proider, Office Space, and Renter information. Modify the TestDriver to import the Authentication Service information.

In the Test Driver, create the Authentication Service Services, Permissions, Roles, and Users first. Then login to create an accessToken, and use this accessToken to pass to the restricted access the methods.

The new TestDriver should be placed in the package: “cscie97.asn4.test”.

When implementing your design, please document any variances from the design, provide justification for your changes and describe how your changes continue to support the requirements.

Remember to use Java doc to document all classes and methods. Add java comments inline where appropriate to explain code logic.

Sample Data

The following input file will provide data for your TestDriver class to load authentication information.

authentication.csv	authentication data

What To Turn In

You’ll turn in a zip file containing

- Your source code (no .class files)

- Your data files
- Sample output
- Your design document (in pdf format)
- Include a document (in pdf format) describing your results:
    - Comments from peer design review and optionally the functional review
    - Any changes that you made to your design and how they continue to support the requirements
    - Did the design document make the implementation easier?
    - How could the design have been better, clearer, or made the implementation easier?
    - Did the design review help improve your design?

We should be able to unzip your file into a directory, then cd into that directory and compile your program with the command.

- javac cscie97/asn4/squaredesk/provider/*.java cscie97/asn4/squaredesk/renter/*.java cscie97/asn4/squaredesk/authentication/*.java cscie97/asn4/test/*.java

We should be able to run your program with the command

- java -cp . cscie97.asn4.test.TestDriver authentication.csv

where authentication.csv contains a set of commands to configure the Authentication Service.

Caution: When you believe you're done, try zipping your files, then unzipping them into a totally different directory and following the steps above. In other words, test your packaging before you submit your assignment.

A grade sheet specifying the criteria for grading this assignment will be posted on the course website.