

## FIT3077 Assignment 2 Design Rationale

Google Doc Link:

<https://docs.google.com/document/d/1u-StasBbIKkpKSfexs4929j36D9goCc9OKkBCcMq6qw/edit?usp=sharing>

## FIT3077 Assignment 2 Design Rationale

Below is a list of all classes and their primary functions

- Application

Main driver of the program

- Web Service Server

Provides data from API and handles all requests.

- Login

Provide login and verify token functionality for user sign-in

- SelectAction

Performs action selection for user to interact with the system such as bidding, create request and viewing contracts.

- NewTutorRequest

Create a request based on competency, subject, preferred hours per lesson, number of session per week and preferred rate.

- BidOnStudentRequest

Create bid based on rate, duration of lesson and additional information that may help the bid.

- SeeContracts

View terms of the contract which includes the terms of engagement i.e., tutor's name, qualification, competencies, the lesson to be taught, time/day of the lesson, the number of session per week and the rate.

- UserCollection

Stores details of each user registered in the system.

- NewRequestDetails

Provides the details required for creating a new bidding request.

The Application class has the role as the main driver of the program. It uses the Login class to enable user to login and verify their login authentication. The classes are implemented this way as to adhere to the Single Responsibility Principle (SRP).

UserCollection Class holds instance of User objects. The User Class is designed to adhere to Open Close Principle. It is contained in a separate class to allow for extension by other classes while remained closed for modification.

Facade design pattern is used to allow SelectAction class to make calls to other classes such as NewTutorRequest, BidOnStudentRequest, SeeContracts. The advantage of this pattern is to provide an interface to the client to access the system more intuitively and hide the complexities of the system.

Observer Design Pattern is also used in this system. When there is interaction of bidding activity, its activities are notified to their respective dependent objects automatically. The Rest web service acts as the publisher and SelectAction class acts as Subscriber notifying the server that a change of bid or request has occurred.

