# Requirements Specification for Client-Server Submission System

## Objectives

Develop a system according to the customer’s requirements on time and within budget.

This system will meet the following requirements:

* be easy to maintain over time
* be of high quality
* be turn-key

## Customer Requirements

1. Client Server based system (non-concurrent server).
2. Submissions will be accepted based on class enrollment and assignment due dates.
   1. If the assignment is passed the due date, it will be rejected.
3. The system will make use of one-shot non-encrypted passwords to implement security for the students submitting the assignments. The passwords will be generated by the lecturer by means of the system and then handed out or emailed to the students.
4. No database will be used. The data and configuration settings should be stored in text files. The end user would like the ability to modify these files in order to adapt the default settings of the system, as well as defining the courses and students.
5. Automated testing of the submitted assignments is a key requirement. The end user will modify the necessary files to allow the execution and verification of the tests.
6. Reports detailing the number of tests passed or failed should be generated when doing the testing and be available on request.
7. The customer prefers a Perl implementation.

## Main Deliverables

1. Student Exercise Submission system
   1. Source code for main system and modules
   2. User Manual for the Submission System
   3. Module Guide
2. Automated Testing system to test the Submission system
   1. Module Test Plans(1 for each module)
   2. Module Test Implementations (1 for each module)
   3. System Test Plan and Test Implementation
3. Customer Demonstration Plan

## System Design of the Student Exercise Submission system

The Submissions System will be broken up into various modules and / or separate programs to facilitate the distribution of work and testing.

### The following modules will be developed:

* A password generation and verification module
* A reporting module

### The following stand-alone programs will be developed:

* A Submit Client program that will be used by the students to confirm their identity to the Submit Server and submit assignments.
* A Submit Server program that will run in the background and monitor for communication from the Submit Client program. The Submit Server program will:
  + accept input from the Submit Client program
  + verify the identity of the student using the Password Generation and Verification module
  + validate the student’s enrollment in the specific course
  + validate the due date of the assignment
  + only once all these requirements have been met will it transfer the assignment submission to the relevant folder
* A Main program that will tie all the modules together and allow the client to:
  + Generate all or a single password file for the student(s) using the Password Generation and Verification module
  + Clear all or a single student’s password file.
  + Start, stop and verify that status of the Submit Server
  + Test and report on a single assignment at a time.
  + Automatically generate the necessary directories required to operate the system based on configuration files stored in a configuration directory in the system. These configuration files will be maintainable by hand.

## System Design of Automated Testing system

### The following normal and exception testing programs will be developed:

* Testing for the Password Generation and Verification module
* Testing for the Reporting module
* Testing for the Submit Client
* Testing for the Main program
* A full system testing program that will drive the other testing programs and test the system as a whole.

## Customer Demonstration Plan