**4.3 Security Requirements**

User account authentication will be handled by Temple’s server. The admin Username and Password will be Provided by Temple Authority, where he/she only can login to system.

Backup and recovery of data will be handled by TU Cloud Services.

This system will be run through a series of security checks at the university to assure the security of our system, and will then be hosted on the TUcloud, and secured by this server as well, as is guaranteed by TUcloud. Even non-TUcloud systems will go through Temple’s standard security scans to ensure that there are no inherent security problems. Fields that raise security risks are the user logins, which if targeted could lead to unauthorized access to the system, also any field containing personal information of students or other users of the system. The system will use a SQL database, so we will have to assure the prevention of injection throughout the system.

The production system will use Secure Socket Layer encryption technology (SSL) requiring login via HTTPS security protocols.

## 4.4 Software Quality and Testing

**Unit Testing** – We will use unit testing to verify data flow through the course of our system operations during runtime. We will create unit tests to check against each piece of data involved in any system execution, to verify that our code is correct. We will write unit tests ourselves and run them to make sure that our system is flowing data correctly.

**System Testing –** We will use system testing to verify that our systems software and hardware that have been fully integrated, comply with the requirements for the system. We will test the fully developed system in a black box manner to make sure that the system works entirely and correctly after being implemented. We will ensure system consistency in the user interface and functions.

**Performance/Loading Testing –** We will use performance testing to test that the system works under all possible conditions relating to workload at a normal and high capacity level. Performance testing will test whether our system works at a reasonable pace doing normal operations, at various levels of workload. Load testing will test how efficient our application is at the anticipated level of use. We will execute this testing by running and using the application in the manner that we anticipate it will be used in, to identify where any problems may occur.

**User/Acceptance Testing -** We will use acceptance testing, to determine if the developed system meets the requirements agreed upon for the client. The client will physically use the system and point out any missing requirements or pieces within the system. This type of testing will be used to determine if various operation scenarios are capable of completion, which will tell us if the system is functional enough to use or not. We will put the application in the client’s hands and allow them to use it the way they would normally. The users will identify and report to us where any bugs or errors are occurring in the application, or anything that they dislike about the system.

## 4.5 Costs

There are no expected costs to the client at this time.

## 4.6 Conversion of Data

There is no current system exists at this moment. The admin will be able to upload documents to the system such Excel Sheets.

## 5.1 Client Commitments

**This semester:**

* Attendance at scheduled meetings.
* Review any new documentation between meetings.
* Review of all system documentation in detail and provide feedback where necessary.
* Review of all user-interface prototypes and provide feedback where necessary.
* Providing all essential information that should be included in reports and forms.

**Next semester:**

* Attendance at scheduled meetings.
* Review any new documentation between meetings.
* Review user interface and report design.
* Prompt response for email communications.
* User Quality Assurance Testing required during the semester at predetermined times.
* Perform acceptance testing by using various components of the system.