**Database Project Charter**

1. **Business Case**

Currently, information related to our testing process is mainly being stored across multiple Excel files, at different locations inside our shared J drive. The problems with this are:

* These files or the information inside of them can be accidentally deleted; inaccurate edits and inconsistencies may go unnoticed especially when we have multiple copies of the same file like the Equipment Status Excel file where there’s a file for each day.
* Information can be hard to find across hundreds of folders.
* Updating these information can be time-consuming with the different file locations and the different formats across files.
* Some useful information may not be available or be up to date (e.g. equipment specifications).

1. **Objective Statement**

The objective of this project is:

* Have all information available at one single location that is the database.
* Have data sets be in a fixed, therefore consistent format.
* Have a user interface accompanying the database that allows for easy viewing of information.

1. **Opportunity Statement**

The completion of this project gives Hanon Systems Concord an all-in-one destination for viewing information related to our testing process. Among many things, the accompanying web application will also display equipment booking schedules, test schedules that will simplify test planning; also DUTs and harnesses’ history and storage locations, helping with general organization.

1. **Project Scope**

For site: Concord

Intended Users: Engineering, GVV

Information that will be transferred online: equipment specifications, TRs, test schedules, test running times, program information, equipment schedules, equipment running time.

1. **Project Team**

Project Lead: Simon Duley, David Li

Programmers: Quang Thang Nguyen, CS student

1. **Project Plan**

* Determine the concurrency requirement of usage. Concurrency means the ability to read, write to the database at the same time; we want to determine what should happen if this situation occurs, what happens when two writes occur at the same time. (2 days)
* Research the database management system suitable for application based on the concurrency requirement. It should be integrable with Python, and have a management application. (3 days)
* Create the database, set up links within the database, and fill the database tables with sample data. (2 weeks)
* List the pages of the web application, how they link to each other, what should display on them, what features should be on them. (2 week)
* Write the queries needed, test the queries. (1 week)
* Write the pages of the web application, link them to each other, link them to the database, and test the web pages. (4 weeks)
* Add log-in capability, user types. (2 weeks)
* Format and improve aesthetics of pages. (2 weeks)
* Find a host server or in-house server to run the web application. (1 week)