Assumptions & Clarifications

**Advisees** –

**Assumption**: In the design of our database, we are assuming that the passwords being stored in the Advisees table are “hashed” and stored as a “digest”. This is because actually hashing the passwords would fall out of scope of the project and isn’t the main focus. The purpose of the username/password combo is to provide a login ability for the front-end user.

**Issue**: Passwords are not securely stored within our database. Anyone with access to the table on the backend, has access to the passwords stored within that table.

**Resolution**: In this iteration of the advising app we have created (as a proof of concept), this method of storing passwords is adequate. In future work, we would have to pass the passwords through a hashing algorithm and store the passwords’ digest rather than the password itself.

**-------------------------------------------------------------------------------------------------**

**Assumption**: In the design of our database, we are assuming that every student is from the United States, it was easier to create mock data this way.

**Issue**: No issue.

**Resolution**: Nothing to resolve.

**----------------------------------------------------------------------------------------------------**

**Assumption**: In the design of our database, we are assuming that every student has a phone number and that it is the same format, it was easier to create mock data this way.

**Issue**: If there were students from a different country with an international phone number format, it could cause an error with the AdviseePhone column.

**Resolution**: Assumption was made to limit students being from the United States, if we wanted to work on this project in the future, we would need to adjust the AdviseePhone to accept a wider range of phone numbers.

**----------------------------------------------------------------------------------------------------**

**Advisors** -

**Assumption**: In the design of our database, we are assuming that the passwords being stored in the Advisors table are “hashed” and stored as a “digest”. This is because actually hashing the passwords would fall out of scope of the project and isn’t the main focus. The purpose of the username/password combo is to provide a login ability for the front-end user.

**Issue**: Passwords are not securely stored within our database. Anyone with access to the table on the backend has access to the passwords stored within that table.

**Resolution**: In this iteration of the advising app we have created (as a proof of concept), this method of storing passwords is adequate. In future work, we would have to pass the passwords through a hashing algorithm and store the passwords’ digest rather than the password itself.

**---------------------------------------------------------------------------------------------**

**Assumption**: In the design of our database, we are assuming that advisors default location is one that is provided in the Location table and not an office number in an existing building. This was done due to having all mock data at a mock university.

**Issue**: Unrealistic default locations for some advisors.

**Resolution**: Random default locations were assigned to some advisors, to create an illusion that some advisors have default locations and others don’t. This was done to somewhat mimic realistic choices of location for academic advising.

**---------------------------------------------------------------------------------------------**

**Locations** -

**Assumption**: In the design of our database, we assumed that any location could be a location for advising.

**Issue**: Some locations are unrealistic advising locations.

**Resolution**: Unrealistic advising locations are not necessarily a problem. The purpose of the Locations and Buildings table was to showcase a relationship between the location and building for the means of an appointment. Since there is a relationship for every location, the realism of said locations is irrelevant for this project.

**---------------------------------------------------------------------------------------------**

**Buildings**-

**Assumption**: In the design of our database, we assumed that any building could be a building for advising.

**Issue**: Some buildings are unrealistic advising buildings.

**Resolution**: Unrealistic advising buildings are not necessarily a problem. The purpose of the Locations and Buildings table was to showcase a relationship between the location and building for the means of an appointment. Since there is a relationship for every location, the realism of said locations is irrelevant for this project.

**---------------------------------------------------------------------------------------------**

**Scheduled Appointments**-

**Assumption**: In the design of our database, we assumed that any advisement appointment, regardless of duration indicated, would use up an entire hour block for appointment scheduling. This was done to simplify the process of creating appointments and avoiding scheduling conflicts.

**Issue**: This is an unrealistic approach, as a 15-minute period of advising makes it so no other student may schedule in that hour block and forces the advisor to get paid for the full hour despite only working 15 minutes.

**Resolution**: For the scope of this project, this approach is acceptable. In future work, something we could improve upon is implementing logic to allow for multiple appointments to be created in an hour block if the entire hour block is not already scheduled.

**---------------------------------------------------------------------------------------------**

**Assumption**: In the design of our database, we assumed that advisors only work on business days. This was to avoid having the advisors working on the weekends.

**Issue**: No issue.

**Resolution**: Nothing to resolve.

-------------------------------------------------------------------------------------------------------------------------------

**Assumption**: In the design of our database, we assumed that all advisors have the same schedulable hours. This was to simplify the GUI for our front-end.

**Issue**: It is unrealistic to have all advisors have the same hours.

**Resolution**: This is adequate for the scope and time constraints of this project. In future development, it would be better to add logic to dynamically change the options of the office hours for each individual advisor, but this falls far out of scope of this project.