

## Criterion E: Evaluation

### Meeting the Success Criteria

- Creation of database to store students
  - ✓ Met as program checks to see if a program has a “database.db” file. And if not, will make one.
- Creation of a graphical user interface where all of the information from the database is presented along with programmed controls
  - ✓ Met as JTables for the students, classes, assigned classes, assignments along with JButtons to add, edit, delete, and “go to student’s page” are all present.
- Client will be able to to add edit and delete students
  - ✓ Met as all the buttons made for these actions work with it being updating in the database
- Client will be able to add edit and delete classes
  - ✓ Met as all the buttons made for these actions work with updating it in the database.
- Successful implementation of a class assigner that can add or remove classes from the selected student
  - ✓ Met as the client can properly remove or add assigned classes which saves and updates the selection in the database and JComboBox.
- Client will be able to add edit and delete assignments
  - ✓ Met as all the buttons complete these actions properly and are correctly saved to the database.
- Client will be able to see calculated percentage and grade once an assignment has been added to the program
  - ✓ Met as calculated percentage and grade is seen in a JLabel once an assignment is added to the program along with it constantly updating.
- Database is used to present information back to the program when it is restarted
  - ✓ Met as the program once restarted transfers all the info from the database into the corresponding JTables in the program .
- Creation of search tool to search between for students, classes and assignments

- x Not met as the program does not include a search bar anywhere in the program.

## **Recommendations for Further Development**

### **- Minor Improvements –**

- A better layout of the GUI as instead of having the JButtons open a new frame every time, they could instead place a new panel on top of the frame so my client can not worry about exiting out of multiple frames.

### **- Major Improvements –**

- An inclusion of a search bar in the program as it will make looking for students a lot easier when around 100+ students end up getting added to the program. This will also complete my last required success criteria.
- Major bug fixes: Although all of the features of the program work individually, when I use them in conjunction with each other may cause a few errors (ex. changing the ID of a class when that class is already assigned to a student). By being able to fix those bugs, it will make the product a lot more usable for my client.

## **Extensibility**

Although my client had enjoyed the product, I still wanted to talk with her about some ways to improve it through my interview located in Appendix 2. Firstly she noticed that she could interact with all the frames opened in my product and if she wanted to delete a student while I was on their student page, could cause major issues. Secondly, there were still some issues with the ID, as the program didn't display every error to the user for example when attempting to add a new user with the same ID. By applying patches to these issues in the future, I can make my program a lot more usable to my client and improve her experience with it overall.

**Word Count: 557**