Ethan J. Ennis

ELEC 3225 – 01

Assignment 2

5/30/2021

**test**

**Incremental development**

1. *Initial version (Estimated time of completion of Initial version – 1-2 weeks)* 
   1. Specification
      1. The initial version will be the implementation of the base class and derived objects. There will be a base class for users and then derived into students, instructors, and admins. The blueprints for these classes are to be implemented and the member functions will not be completed at this stage. Ensuring these classes are structured properly and member functions can at least be called from the main code.
   2. Development
      1. Base class for user
         1. Attributes include: first/last name, ID number.
      2. Derived classes
         1. Students
            1. Included functions: search courses, add/drop courses, print schedule.
         2. Instructors
            1. Included functions: print class list, print schedule, and search courses.
         3. Admins
            1. Included functions: add/remove courses to system, add/remove users, add/remove students from courses, and search/print rosters and courses.
   3. Validation
      1. Hard coded objects are then created in the main program to ensure that there are no bugs in the initial version of the program thus far. These hard coded objects are instantiated with attributes to ensure the constructors work, and functions are called by these objects to validate they are called properly, and a print message proves this.
2. *Intermediate version (Estimated time of completion for Intermediate version – 8-9 weeks)*
   1. Specification
      1. All member functions are to be implemented. This will include all constructors/destructors, get/sets, prints, searches, add/removes.
      2. Databases are implemented to be able to work with up to 100 students, 10 instructors, and 1 admin. Other database will support all courses and include course names, CRNs, times, and instructors.
      3. User interface is implemented as a command line interface. Will be menu style where user will input choices that are then used to call functions to do desired tasks.
   2. Development
      1. Functions
         1. constructor – make sure base constructor is called properly to assign attributes the correct values.
         2. Destructor – make sure the destructor is called properly to avoid memory leaks.
         3. Get/set functions – make sure all get functions can return the desired values and that set functions are setting the correct values to the corresponding attributes.
         4. Add/remove functions – passing in int/string values that are to be used to add or remove users/courses.
         5. Search functions – ensuring that the passed in values (int/string) can locate the desired entity and return that it is either found or not.
         6. Print – making sure the right information is printed neatly.
      2. User Interface
         1. Command line, menu style interface for users
            1. Switch cases dependent on users’ choice

Function calls depending on choice to access member functions.

* + 1. Databases
       1. There will be two databases.
          1. One database to contain all users (students, instructors, and admins)
          2. Another database containing all courses and course information (CRNs, instructors, course name, time)
  1. Validation
     1. At this stage, all the program’s components will be tested. The user interface will have to be ran and this will be testing that there are not any runtime errors or formatting issues. The classes/objects are also going to be tested to make sure that all member functions are doing their desired function. The databases will be tested to ensure the information that is contained within is being access properly and without issues.

1. *Final version (Estimated completion time of final version – 2-3 weeks)*
   1. Specification
      1. After finding bugs from the individual components of the system, (UI, database, classes) these will be worked on to then produce a final version of the program.
   2. Development
      1. The program will be tested as one system. The individual components such as the UI, database, and the classes/objects, and their functions, will be tested to fix all existing bugs that remain.
   3. Validation
      1. Once the bugs found in the testing part of this version are resolved, the program will be run to ensure the system is completely working and all bugs are fixed. Any updates that are needed are implemented in this stage of debugging and maintenance to come out with a final product.