1. **PROJECT TITLE**

SUBJECT’S TASK MANAGEMENT SYSTEM

1. **INTRODUCTION**

“The Roots of Education is bitter, but the Fruit is Sweet,” is probably a well-known saying by one of the greatest philosophers of his time; Aristotle. Although quite compelling, the hardships that are being poured down when it comes to education should not be taken for granted. In fact, due to the progressive passing of time and advancements of technology in our current generation, it is quite undeniable the struggles of fitting something traditional into a contemporary thing. Just taking education for example, traditional learning was great during the old times, but right now, with the ever-growing population plus the different complications around the environment, learning is becoming duller and duller. This is where contemporary learning comes. With it so many forms, the one that might be well-known is E-Learning. E-Learning is a learning system based on formalized teaching but with the help of electronic resources (“What is E-Learning?”, n.d.). Since the advent of e-learning, gaining useful information and learning something has not been any easier. In fact, e-learning is becoming and becoming more ubiquitous because it scales, saves and money and time (“What is elearning?”, 2018). Because through electronic media and technology students and educators worry less of the topics and things that they might miss. Because one thing’s for sure is that technology will not be going away so soon, thus the benefits of e-learning.

Moving forward, education escalates faster and further as a person’s achievements in life go on. College students of Cavite State University – Bacoor City Campus are not excused in the said predicament. Just only imagining the paper works, the tasks and much more that a college student indulges is a no joke. Students of CvSU are feeling this at first hand. In the campus, nothing much is done automatically, because of the lack using systems and concepts of technology that might ease the operations on the said campus. That being said, even being an ordinary college student has its processes per se. Before learning something new, instructors distribute hand-outs that students study and pay for. This is exactly what is happening even when the examinations come. Students must be ready to pay an amount of money to just get a piece of paper and answer some questions. Might be a little less important, but it tolls overtime. Other than that, above all, having a good coordinated system between students and instructors might solve such issues.

Regarding this, a Management System of Tasks will be a good supplement for the current situation of the students and instructors of CvSU - Bacoor City Campus. Nonetheless, having such tool will ease the processes of distributing tasks and avoid lapses in dissemination of different tasks on each student.

1. **SYSTEM CONCEPT AND REQUIREMENTS**

The System has only two levels of access – the student and the instructor. The creation of account for the system is for both the students and the instructors. In the instructors’ side, they are the ones who will manage the creation of the courses. Each student is able to join in a course that is created and handled by an instructor. The instructor will be notified about the students who want to join in his or her class in a specific handled course. The task creation system enables the instructor to create a simple multiple-itemed task. These tasks can be scheduled or disseminated immediately depending on the choices of the instructors. They will be able to view the scores of each student enrolled in his or her handled course in a particular task. In the student’s side, a notification will be sent to notify the students of the incoming or to do tasks. This includes the tasks’ deadline and availability. On the other hand, the instructor will receive a notification about each student that has finished a task in their handled courses. The students are able to view the results of his or her finished tasks.

In our Entity Relationship Diagram using Reingruber’s notation, (see Figure 16) we made twelve entities which are TBL\_EDUCATOR, TBL\_STUDENT, TBL\_YEARL\_LEVEL, TBL\_TASK, TBL\_SECTION, TBL\_PROGRAM, TBL\_NOTIFICATION, TBL\_NOTIFICATION\_ENTRY TBL\_DEPARTMENT, TBL\_COURSE, TBL\_CATEGORY, TBL\_CATEGORIZED\_TASK, TBL\_CLASS, TBL\_CLASS\_STUDENT. All of these have relationships with each other. In entity TBL\_STUDENT which signifies the students who enroll in CvSU Campus. The student have the attributes Student Number which his or her unique identity inside the school, first name, middle name, last name, gender, address with block and lot number, street, barangay, city and province, contact number, email, section and the password for the creation of account of the user for the system. The student must enroll in just one program – student’s chosen field of study like Bachelor Science in Information Technology, Bachelor Science in Computer Science and etc. and each program can be enrolled by one or more students. Each program can have one or more sections and each section contains students who enroll in a specific program. A student who belongs in a section has a specific year level whether he or she is a 1st year, second year and so on.

In the entity TBL\_EDUCATOR is all about the instructors of CvSU Bacoor Campus. Same as the process with the students, the creation of the account for the system will get the personal information of the instructor and only his or her username will serve as a unique identification for the system. Each instructor must belong in just one department like the Department of Computer Studies, Department of Education and so on, and each department can have one or many instructors. The instructor can create courses with a unique and specific course code in the system and all the instructors can see the courses that have been created by any instructor. But once it is created, any instructor cannot create a course with same course code anymore. Only the creator of the course can change all of the details except the course code which is a unique identifier for the system. The instructor can choose the course that he or she wants to handle based on what his or her department head has given. Once an instructor has handled a course, he or she can now make a multiple choice task and categorize it according to his or her liking. In addition, the instructor can handle a class that refers to a prior section. The students can immediately see all the courses that have been created by an instructor and they are able to join in a course’s particular class. Once an instructor created a task for a handled course, all of students who have joined in his or her classes in that particular course will receive a notification about the specific details of the task including the deadline and completion days of the particular task.

Figures 17 – 26 show the database normalization of the system. Figure 17 represents the table of the educators or instructors. The fields or attributes of the TBL\_EDUCATOR are the personal information of the instructor and each instructor has its unique identification which is his or her username. Figure 17 has many fields and shows improper creation of the table for the educator. The columns Block Number, Lot Number, Street, Subdivision, Barangay, City and Province in Figure 1 can be combined into one column. Figure 18 is the revised table for TBL\_EDUCATOR with proper creation of columns. As stated in Figure 18, column Address is a multi-valued attribute because the attributes Block Number, Lot Number, Street, Subdivision, Barangay, City and Province are all in this column just to avoid too much creation of fields. Table TBL\_DEPARTMENT which is about the name of departments that are present in CvSU Bacoor Campus (see Figure 19) is connected to the TBL\_EDUCATOR because each of the educator belongs to a specific department. With the creation of the table TBL\_DEPARTMENT, the attributes have been organized such as the attribute Department ID which refers to the more thorough details about a department.

TBL\_STUDENT has the same concept of the table TBL\_EDUCATOR (see Figure 20) as the column Student Number is the student’s unique key for his or her account in the system. Additionally, a new column has been created for the combined details of the address of a student but it is a multi-valued attribute and has been named as Address. In table TBL\_PROGRAM (see Figure 22), a new table for column Program ID in the table TBL\_STUDENT (see Figure 21) has been created for the other details of the program that a student is enrolled to. But in Figure 23, in case of the attribute Program ID, it is referenced in the table TBL\_SECTION, and this table is referenced in the table TBL\_STUDENT and named as Section ID in its columns (see Figure 23). Table TBL\_SECTION is for the details of the student’s program, section and year level. Table TBL\_SECTION has columns Section ID which a primary key, Program ID which is a foreign key for the table TBL\_PROGRAM which is about the other description of the student’s program, Year Level which is a foreign key for the table TBL\_YEAR\_LEVEL which pertains to the descriptions of a student’s year level and section number which is about block section number.

Table TBL\_TASK (see Figure 24) is about the created tasks of the instructors. It has fields named Task ID which is the primary key, Class ID which is a foreign key for the table TBL\_CLASS (see Figure 25). The Content, Status, Date of Dissemination and Time Limit are all about the details of the particular task. Table TBL\_CLASS is for the classes that will be made in a particular handled course (see Figure 25). It has columns Class ID which is the primary key or unique key, Handled Course ID which is the foreign key for table TBL\_HANDLED\_COURSE (see Figure 26) and Section ID which is a foreign key for table TBL\_SECTION (see Figure 23). Table TBL\_HANDLED\_COURSE is about contains the courses that are handled by a specific instructor (see Figure 26). It has columns Handled Course ID which is the primary key, Course Code which a foreign key for TBL\_COURSE and Handler Username which refers to the username of the instructor in the table TBL\_EDUCATOR. Table TBL\_COURSE is all about the created courses of an instructor. The examples of the details are the columns Course Code which is a unique key for creating a course, Course Suffix, Course Title, Course Creator – name of the instructor who created the course and so on.

1. **SYSTEM DESIGN SPECIFICATION AND CODING**

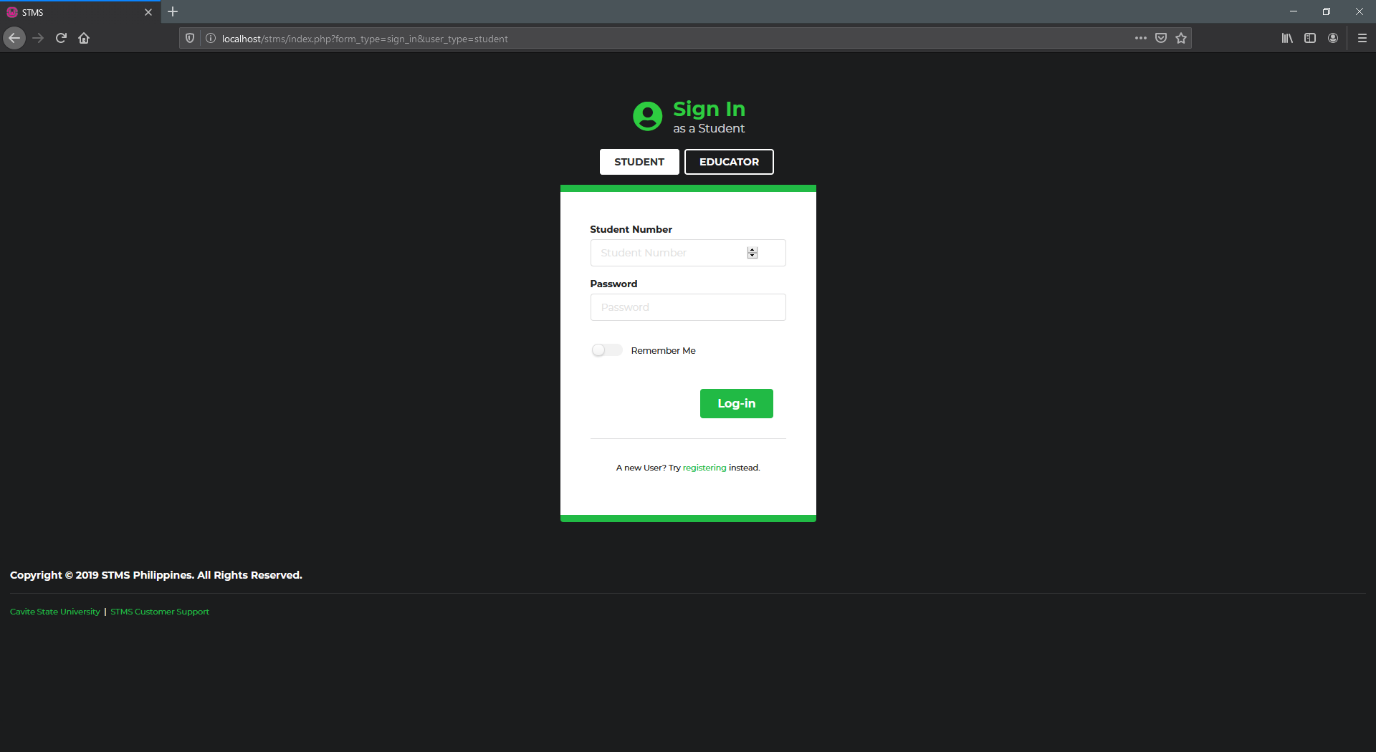
The system is planned out of the goal of creating a system that will ease the dissemination of tasks from the instructors to the students. It is planned to run across different platforms, thus yielding to the choice of using web development tools in order to develop the desired system.

Technically, the system is meant to be up and online in the cloud, but since doing so will result in using a custom or free domain name, the system was later on developed using a local server. An AMP stack (Apache, MySQL and PHP) was used to develop the system through a local server. Specifically speaking, **WAMP 3.2.0 – 64 bit** is used as the official AMP stack application for the system. In this stack, includes the following: **Apache 2.4.41**, **PHP 7.4.0** and **MySQL 8.0.18**. Apache is used as the main server for the local emulation of the website, while PHP is used as the main server-side scripting language for the system, and lastly, MySQL is the official database of the system. Some DBMS tools are also used such as **PhpMyAdmin 4.9.2** for the management of MySQL database.

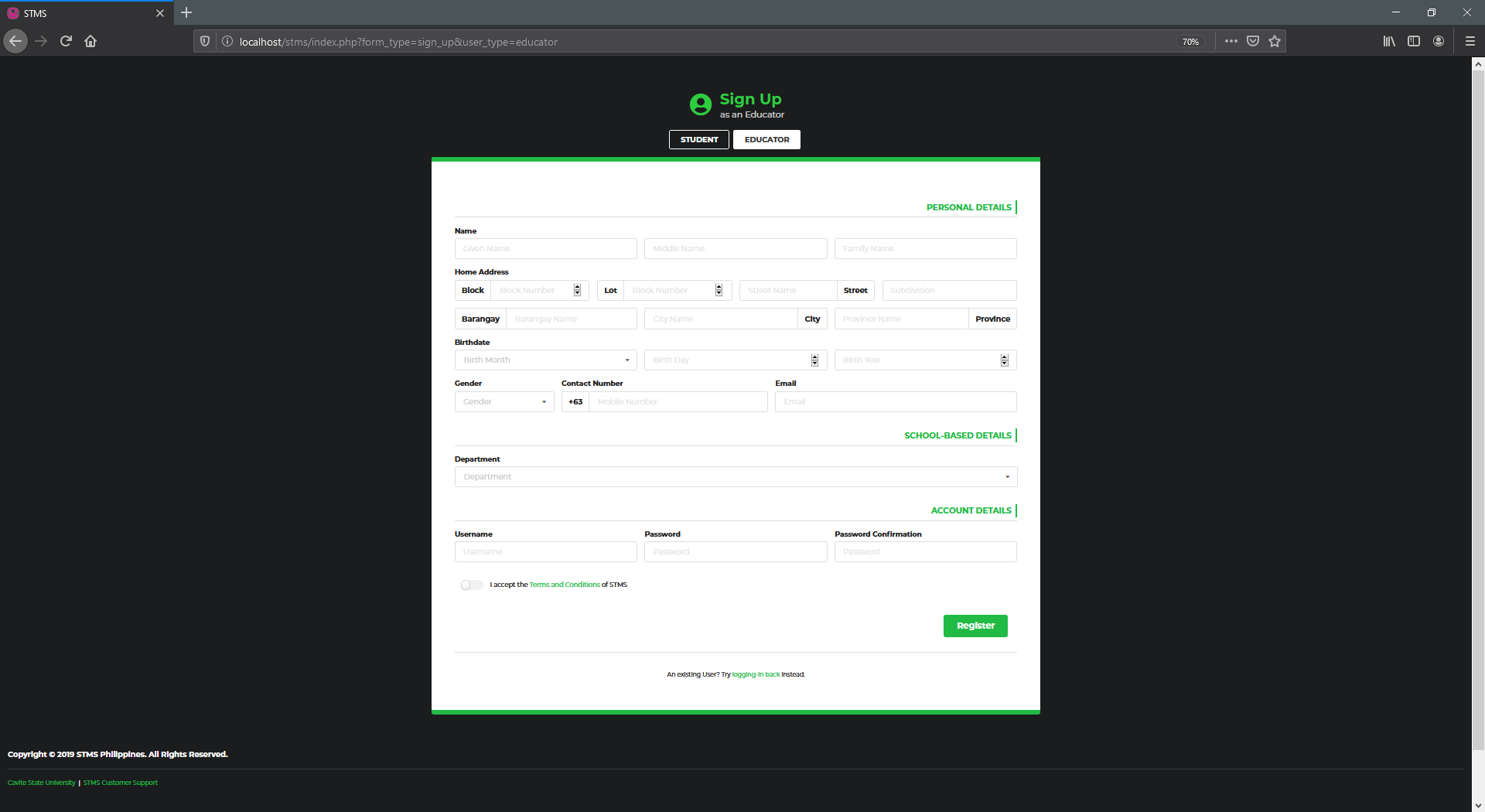
To make the process of development less tedious and laborious, the authors of the system uses a front-end framework for the design and interactive experience. **Fomantic UI 2.8.0** is the framework that gives the system some leeway when it comes to the design and interactivity. Additionally, some manual adjustments must be made to some of the prebuilt designs of the framework. These design adjustments are created through the use of **SASS 1.23.0** which is an implementation and extension of the CSS framework.

The frameworks that are already used in the system maybe seems good on the default basis, but since the authors want to decide some interactivity in the system, some scripting languages are used. **JQuery 3.4.1**, a library for the Javascript client-side scripting language, is used mainly for the custom behaviors and interactions around the system. Lastly, the browser used for the development is **Mozilla Firefox 71.0 – 64 bit**.

Going through, the system has main pages that will be visited by the end-user frequently when using the system. First is the sign in and sign up pages for both the instructors and educators.

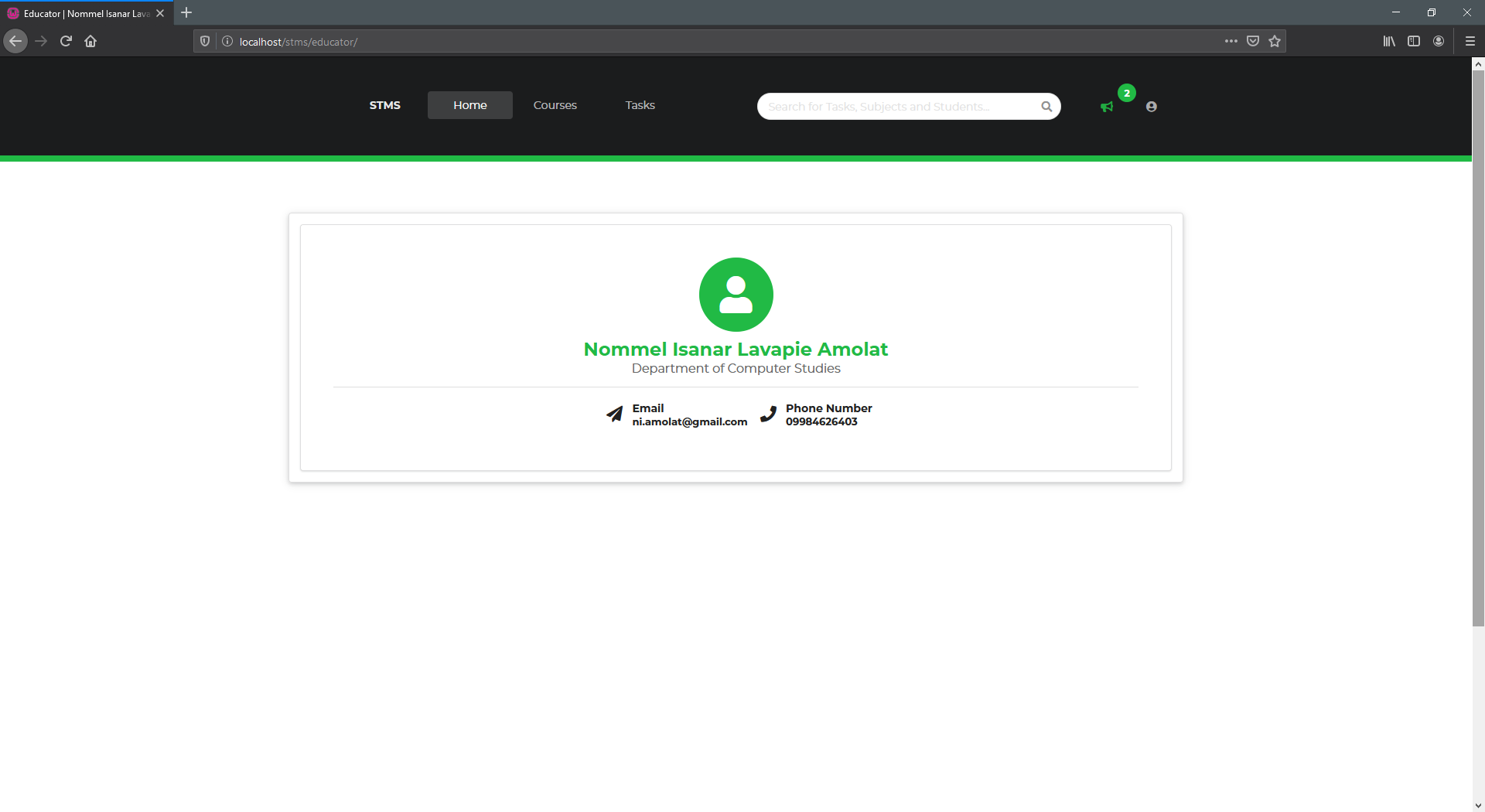


**Figure 1.** Sign In Form of Returning Users



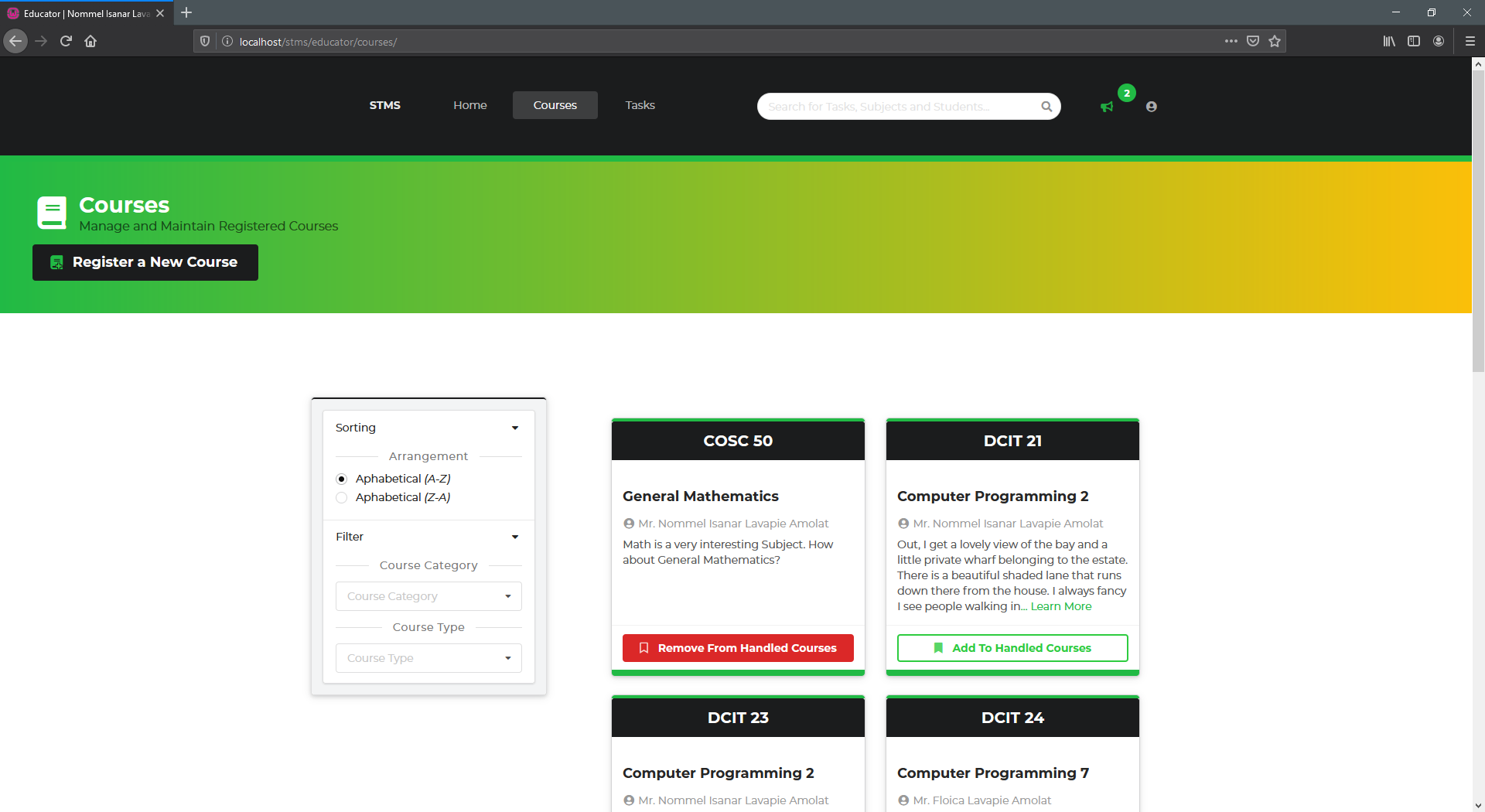
**Figure 2.** Sign Up Form of New Users

Having successful inputs and correct credentials will redirect users to their proper home pages depending on their type.



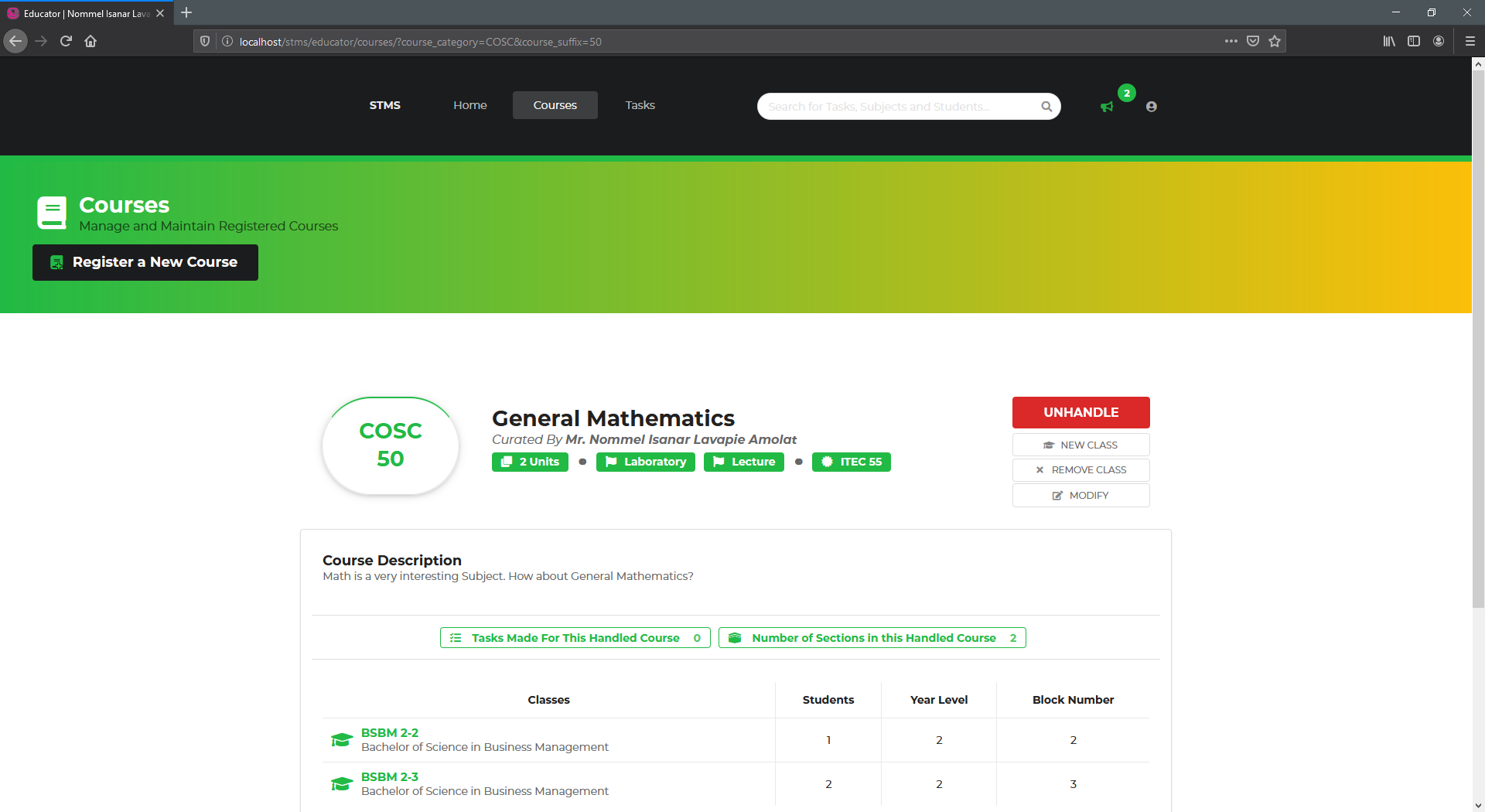
**Figure 3.** Home Page of Logged In Users

In the Educator’s side, the page Course will allow the Educators to create a course and add classes to it. A class refers to a section that is handled by an educator in his or her specific handled course. It is also worth mentioning, that once the course has been created, no Educator is allowed to tamper the information with regards to that course, except if the Educator who is modifying it is also the curator of the course.



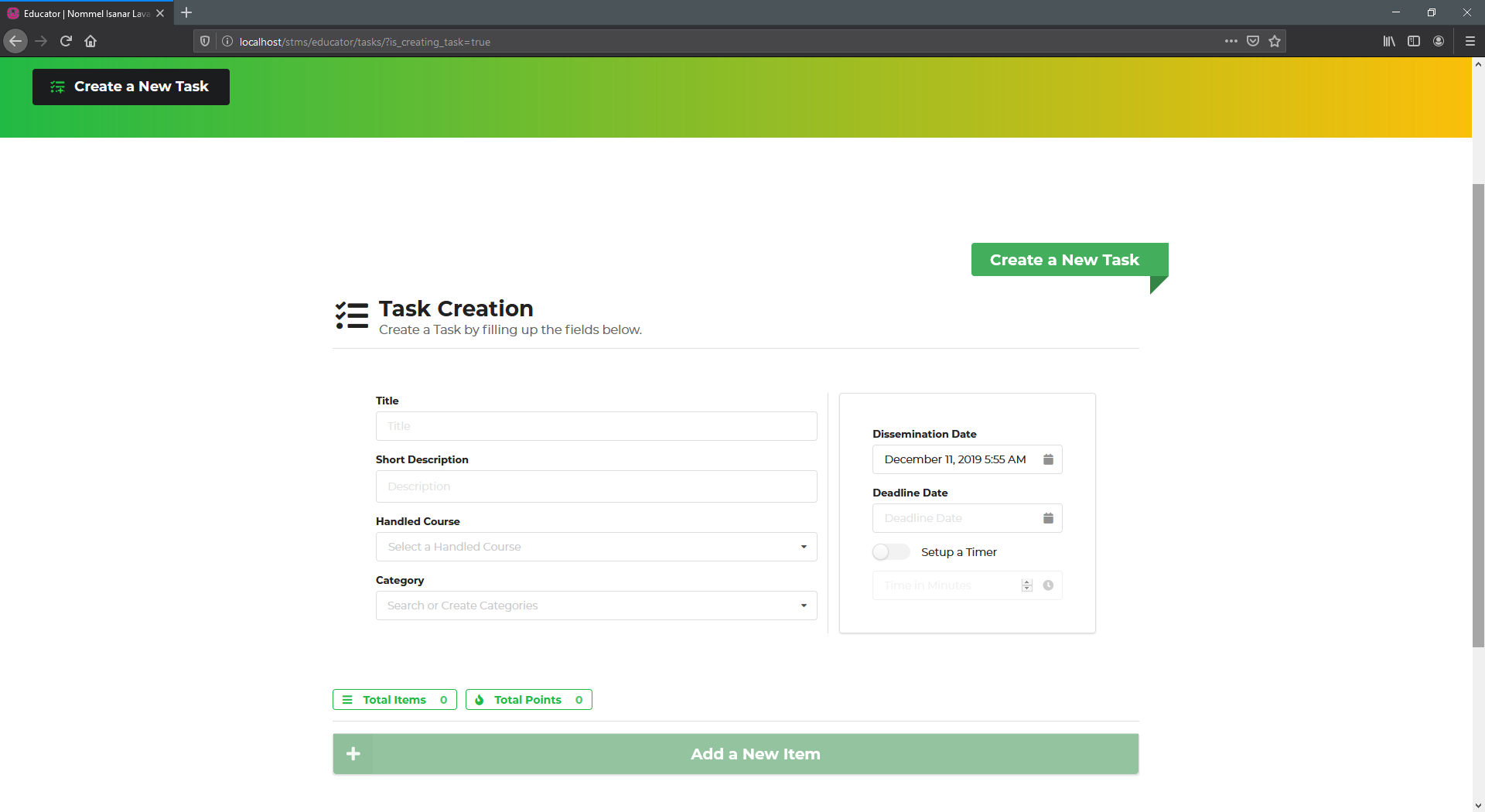
**Figure 4.** Educator’s side ofGeneral Course View

In the General Course View, the courses are displayed as cards and can be filtered and sorted using the panel on the left side right before the courses. The black labels on each card are link to their respective specific course view where more options are available with regards to that specific course.



**Figure 5.** Educator’s side **of** Specific Course View

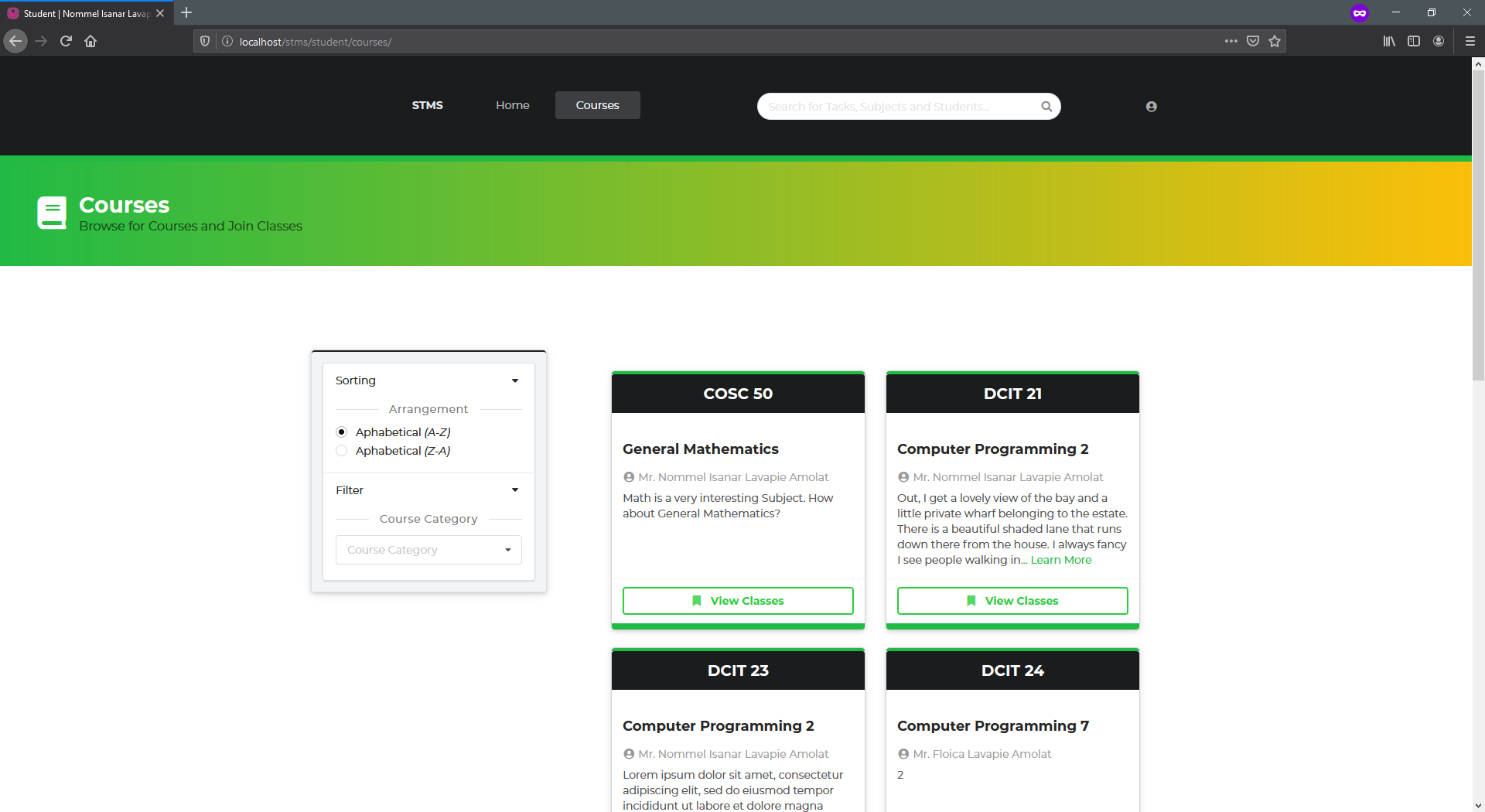
Once a course is handled by an instructor, it will be added to his or her pool of options to distribute in, in the task creation module of the system.



**Figure 6.** Task Creation Module of the System

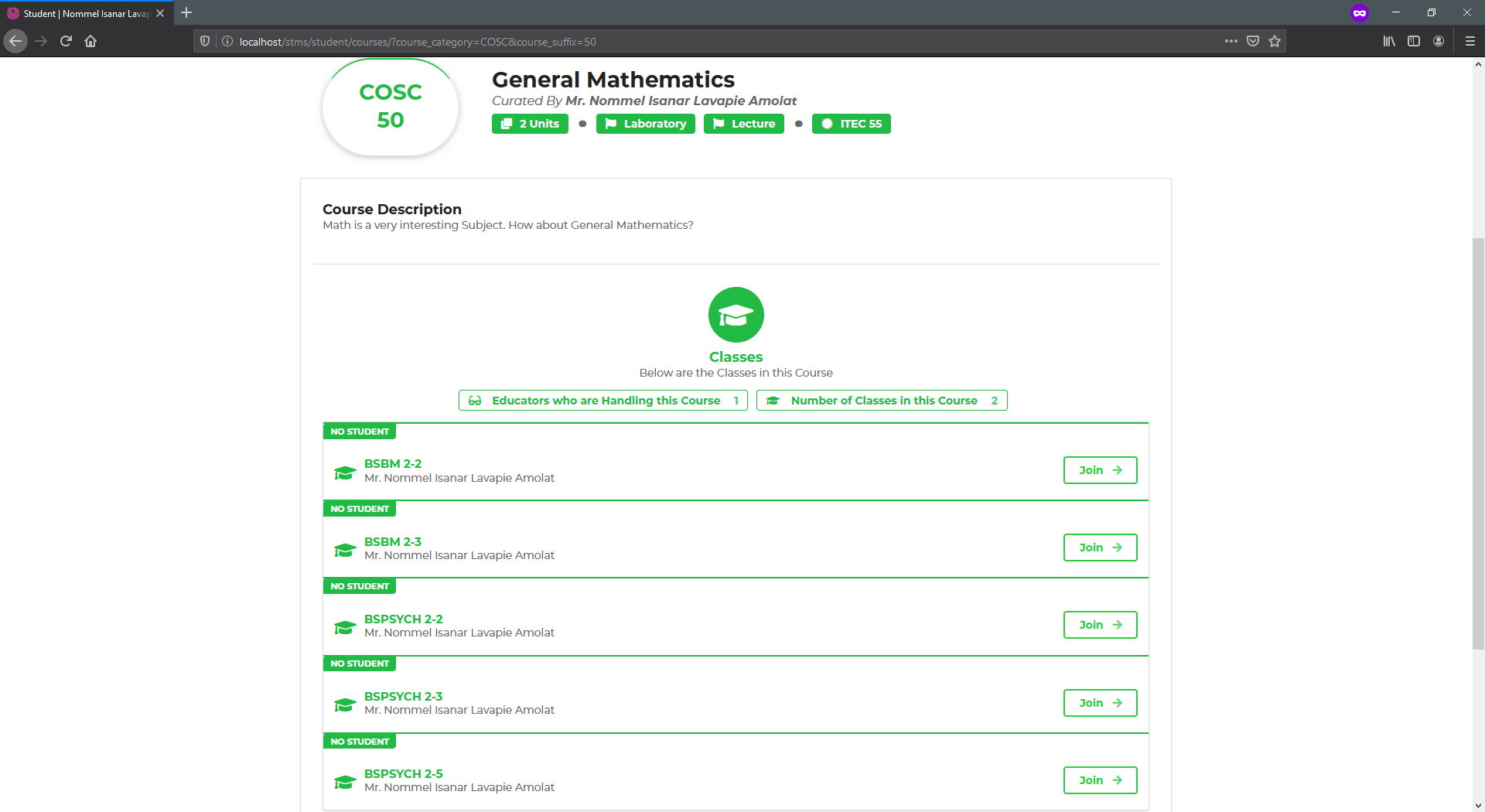
The task creation module enables the educators to create a single task having multiple-itemed questions. He or she can create as many as possible items for his or her questionnaire and he or she can customize the points for each item and mark the right choice for each item. Once submitted, it will be added to the pool of tasks that the educator has made. He can then delete it according to his need.

Going on to the Student’s side, some functions are quite similar with the educator, except only that the student cannot create courses or modify them, he or she can only view it and the classes available to it.



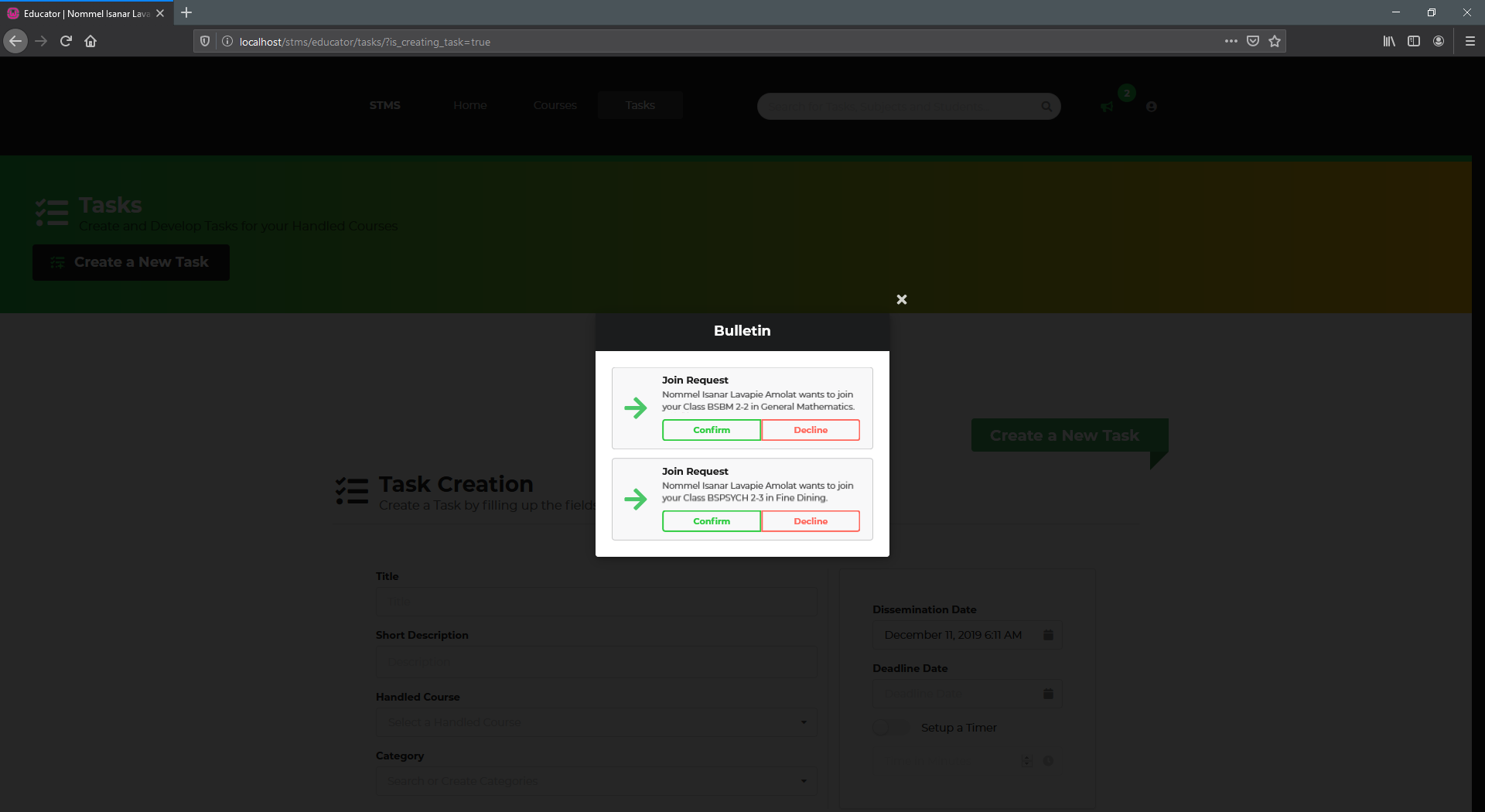
**Figure 7.** Student’s side **of** Specific Course View

The same as the educator’s side of navigating through the courses. The student can specifically isolate the view only to one course and view the classes in it and join a specific class made in that specific course.

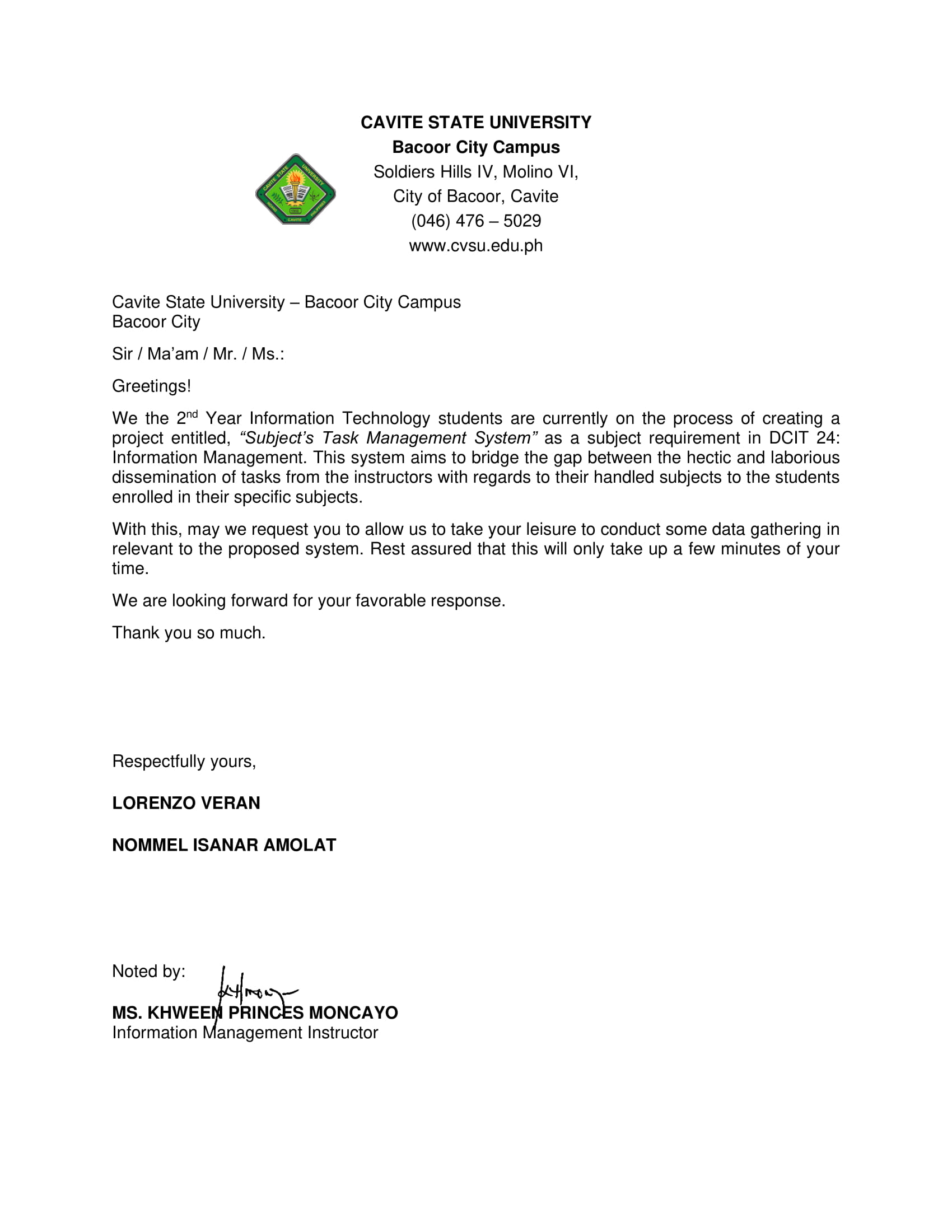


**Figure 8.** Student’s side **of** Specific Course View

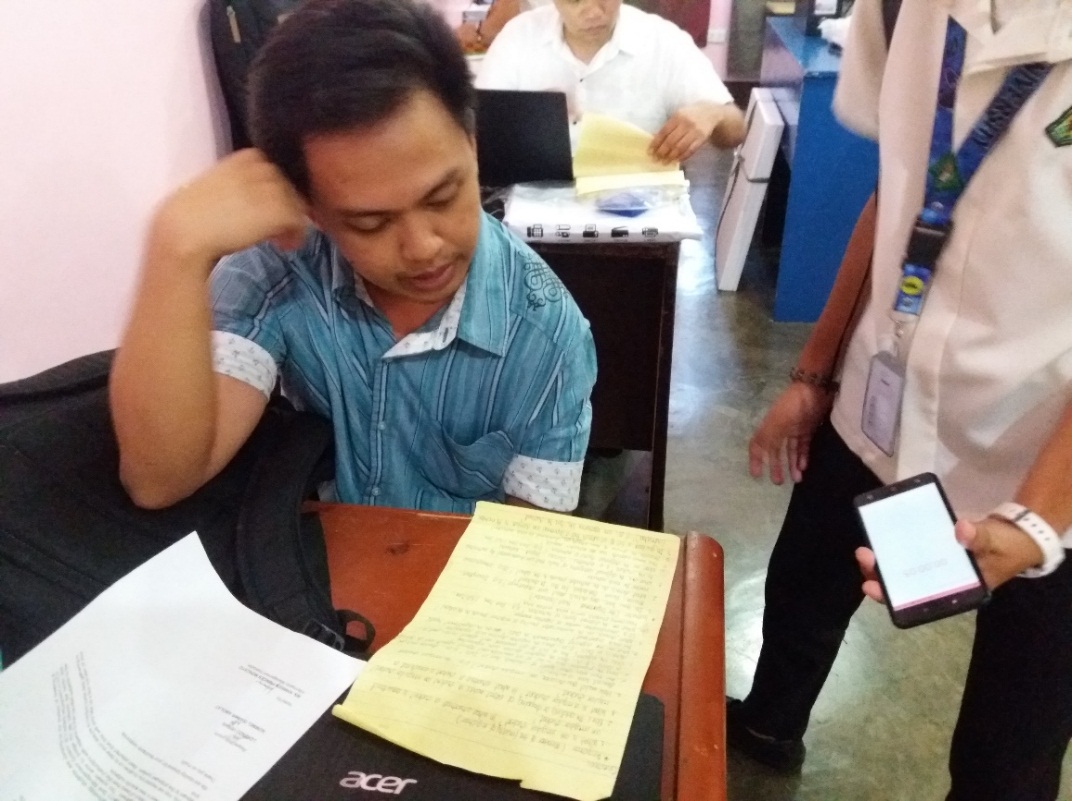
Once the student settled a permission to join a class, the educator will be notified immediately with regards to the student who asked to join in one of his or her classes.

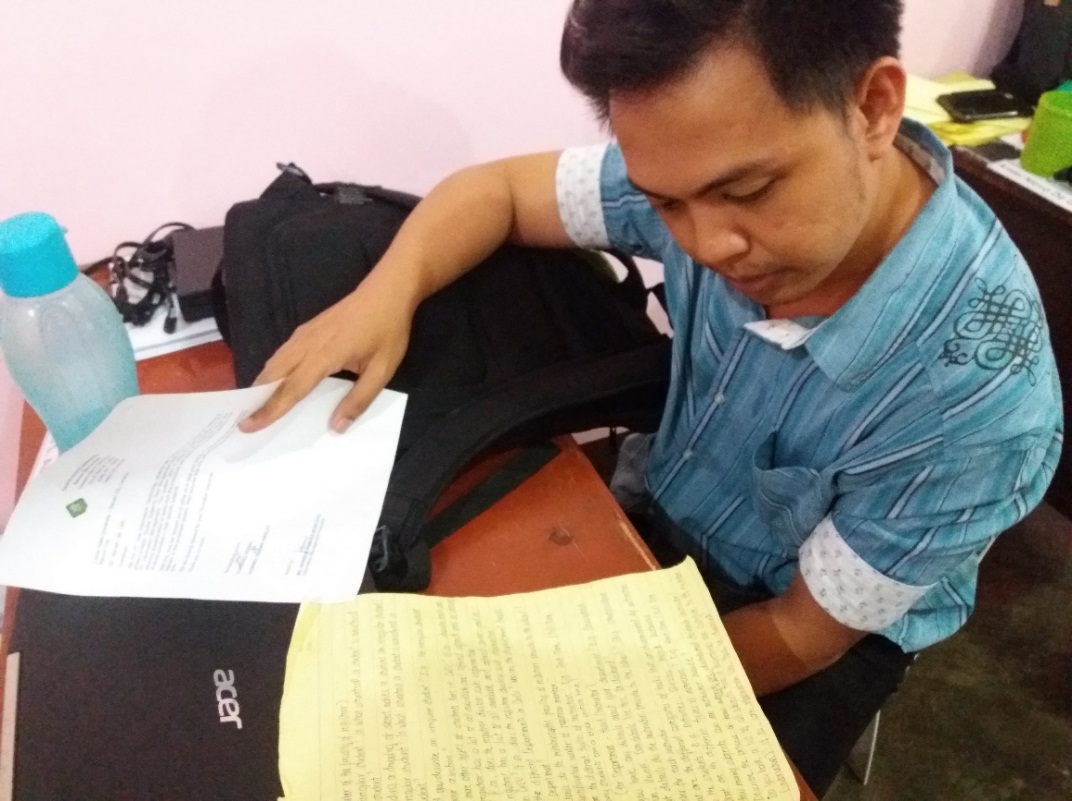


**Figure 9.** Educator receiving Notification right after a Student asked for Permission

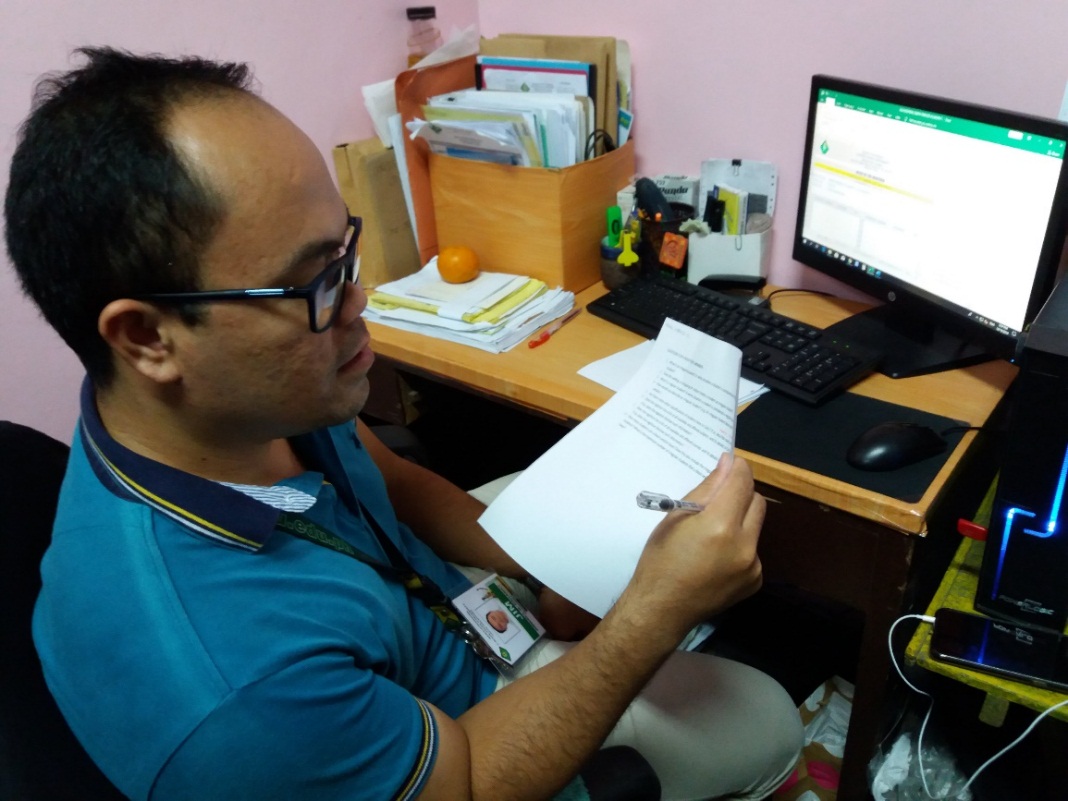
1. **SUPPORTING DOCUMENTS**

**Figure 10.** The Approval Letter approved by the Instructor

****

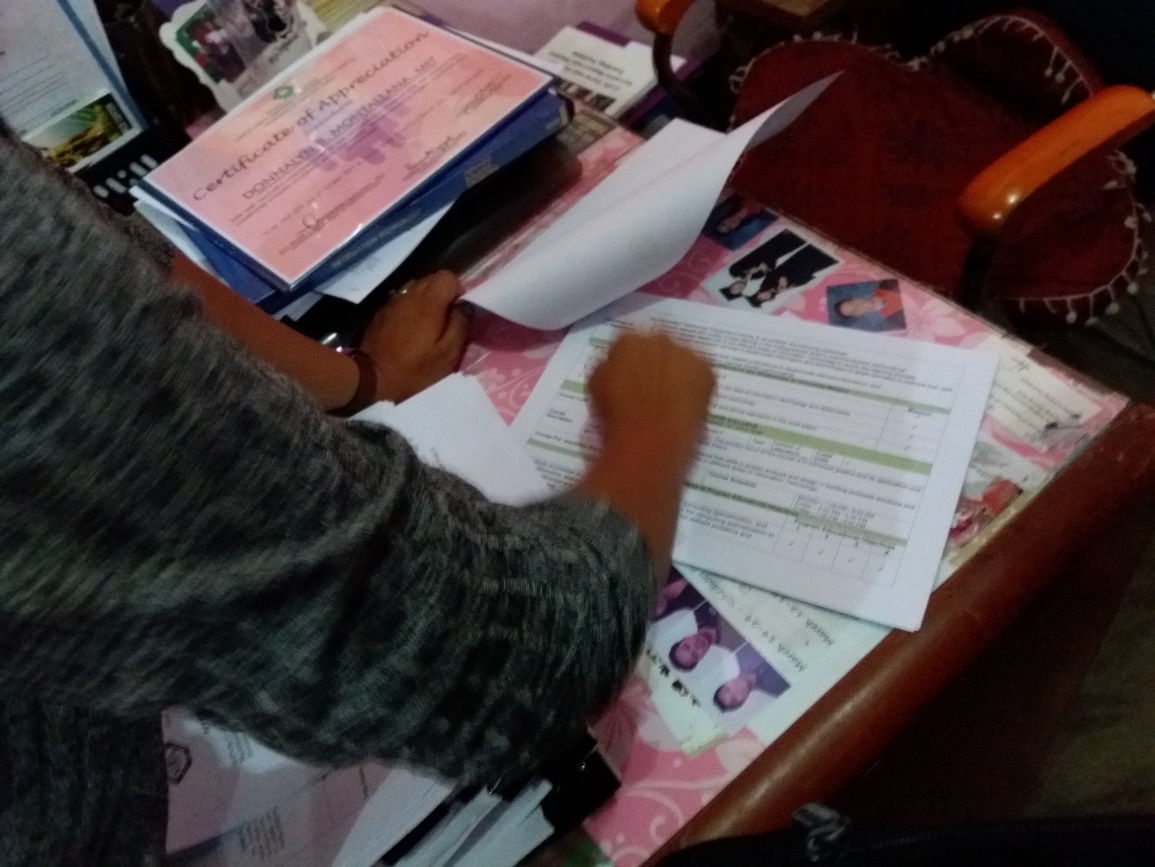
****

**Figure 13.** The Interview with Mr. Roneson Angelo Abug with regards to the Categories of the Tasks that they give to the Students in CvSU Bacoor Campus

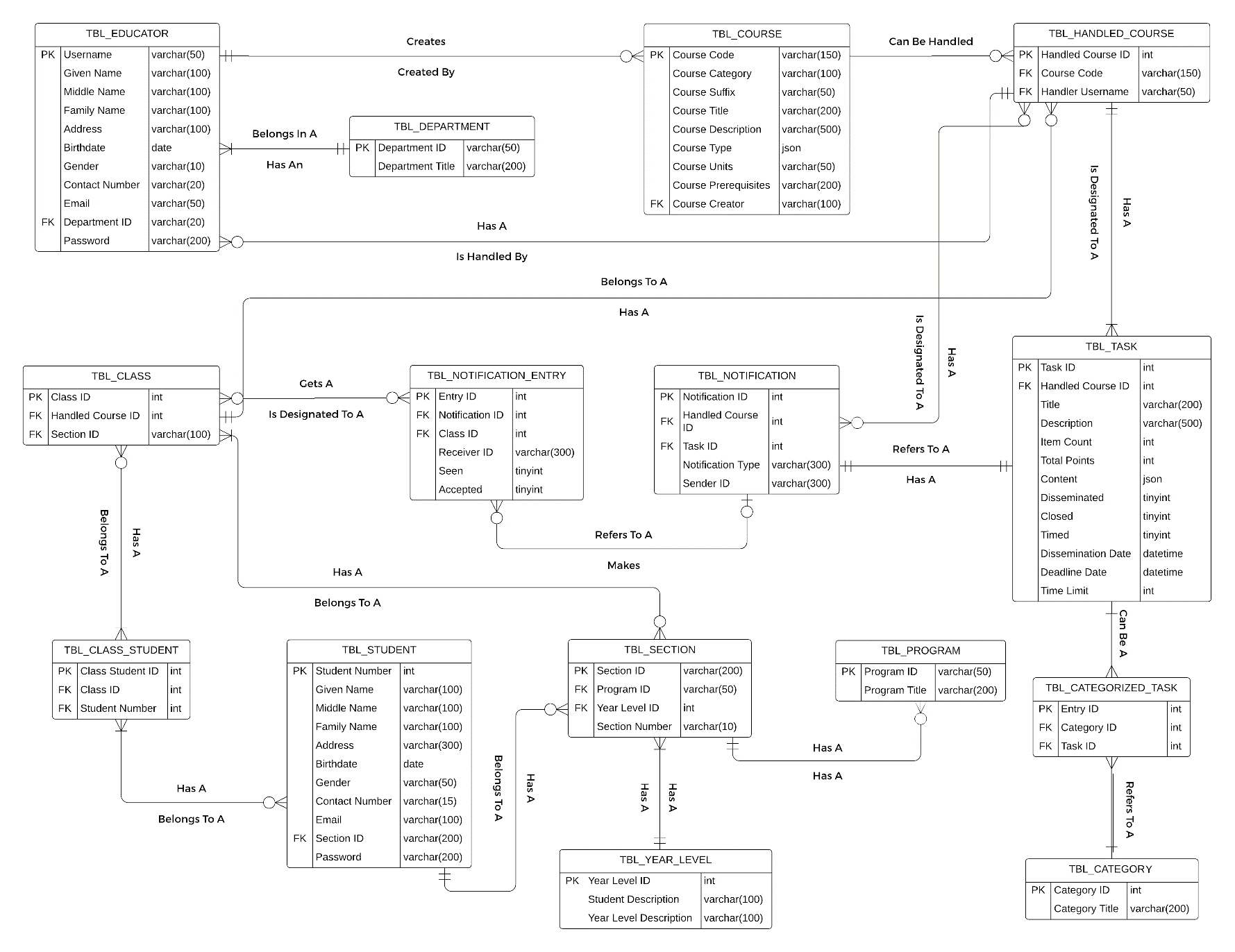




**Figure 14.** The Interview with Mr. Jim Dacanay with regards to distribution of Courses to the instructors of CvSU - Bacoor City Campus

****

**Figure 15.** The Interview with Ms. Donnalyn B. Montallana with regards to the proper naming of terminologies involves in the System

****

**Figure 16.** The Entity Relationship Diagram of the System using Reingruber’s approach

|  |
| --- |
| **TBL\_EDUCATOR** |
| **COLUMNS** |
| Username |
| Given Name |
| Middle Name |
| Family Name |
| Block Number |
| Lot Number |
| Subdivision |
| Barangay |
| City |
| Province |
| Birthdate |
| Gender |
| Contact Number |
| Email |
| Department |
| Password |

**Figure 17.** Unnormalized Table of Educators

|  |
| --- |
| **TBL\_EDUCATOR** |
| **COLUMNS** |
| Username |
| Given Name |
| Middle Name |
| Family Name |
| Address |
| Birthdate |
| Gender |
| Contact Number |
| Email |
| Department |
| Password |

**Figure 18.** 2NF of Figure 17

|  |
| --- |
| **TBL\_DEPARTMENT** |
| **COLUMNS** |
| Department |
| Department Title |

**Figure 19.** 2NF of Figure 18

|  |
| --- |
| **TBL\_STUDENT** |
| **COLUMNS** |
| Student Number |
| Given Name |
| Middle Name |
| Family Name |
| Block Number |
| Lot Number |
| Subdivision |
| Barangay |
| City |
| Province |
| Birthdate |
| Gender |
| Contact Number |
| Email |
| Program ID |
| Year Level |
| Block Section |
| Password |

**Figure 20.** Unnormalized Table of Students

|  |
| --- |
| **TBL\_STUDENT** |
| **COLUMNS** |
| Student Number |
| Given Name |
| Middle Name |
| Family Name |
| Address |
| Birthdate |
| Gender |
| Contact Number |
| Email |
| Program ID |
| Year Level |
| Block Section |
| Password |

**Figure 21.** 2NF of Figure 20

|  |
| --- |
| **TBL\_PROGRAM** |
| **COLUMNS** |
| Program ID |
| Program Title |

**Figure 22.** 2NF of Figure 21

|  |  |  |  |
| --- | --- | --- | --- |
| TBL\_STUDENT | TBL\_SECTION | TBL\_PROGRAM | TBL\_YEAR\_LEVEL |
| **COLUMNS** | **COLUMNS** | **COLUMNS** | **COLUMNS** |
| **Student Number** | **Section ID** | **Program ID** | **Year Level ID** |
| **Given Name** | **Program ID** | **Program Title** | **Student Description** |
| **Middle Name** | **Year Level ID** |  | **Year Level Description** |
| **Family Name** | **Section Number** |  | **Description** |
| **Address** |  |  |  |
| **Birthdate** |  |  |  |
| **Gender** |  |  |  |
| **Contact Number** |  |  |  |
| **Email** |  |  |  |
| **Section ID** |  |  |  |
| **Password** |  |  |  |

**Figure 23.** 3NF of Table of Students

|  |
| --- |
| **TBL\_TASK** |
| **COLUMNS** |
| Task ID |
| Class ID |
| Content |
| Status |
| Date of Dissemination |
| Time Limit |

**Figure 24.** Unnormalized Table of Tasks

|  |
| --- |
| **TBL\_CLASS** |
| **COLUMNS** |
| Class ID |
| Handled Course ID |
| Section ID |

**Figure 25.** 2NF of Figure 24

|  |  |  |
| --- | --- | --- |
| **TBL\_CLASS** | **TBL\_HANDLED\_COURSE** | **TBL\_COURSE** |
| **COLUMNS** | **COLUMNS** | **COLUMNS** |
| **Class ID** | **Handled Course ID** | **Course Code** |
| **Handled Course ID** | **Course Code** | **Course Category** |
| **Section ID** | **Handler Username** | **Course Suffix** |
|  |  | **Course Title** |
|  |  | **Course Description** |
|  |  | **Course Type** |
|  |  | **Course Units** |
|  |  | **Course Prerequisites** |
|  |  | **Course Creator** |

**Figure 26.** 3NF of Figure 25

1. **REFERENCES**

**Online Sources**

What is E-learning? Definition of E-learning, E-learning Meaning. (n.d.). Retrieved from https://economictimes.indiatimes.com/definition/e-Learning.

What is elearning? (2018, August 18). Retrieved from https://www.allencomm.com/resource/what-is-elearning/.

**Submitted by:**

**LORENZO G. VERAN**

**NOMMEL ISANAR L. AMOLAT**

**Submitted to:**

**KHWEEN PRINCES H. MONCAYO**

Instructor