

GROUP PROJECT FRONT SHEET

Qualification	BTEC Level 5 HND Diploma in Computing		
Unit number and title	WEBG301 - Project Web		
Submission date	1/6/2022	Date Received 1st submission	
Re-submission Date		Date Received 2nd submission	
Student Name	Nguyen Duc Anh	Student ID	GCH17051
Class	GCH0805	Assessor name	Nguyen Dinh Tran Long
Student declaration I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice.			
		Student's signature	ducanh

Grade

☐ **Summative Feedback:**

☐ **Resubmission Feedback:**

Grade:

Assessor Signature:

Date:

Signature & Date:

Contents

I. Introduction	4
II. User Requirement	4
III. System Design	5
1. Site map.....	5
2. Entity Relationship Diagram	6
IV. Implementation	7
1. Sample Source Code	7
a. View	8
b. Controller	11
c. Model.....	13
2. Web screenshots.....	15
V. Conclusion.....	27
1. Evaluate.....	27
2. Future Improvement.....	27
VI. Appendix	27

Project Web

Student Management System

I. Introduction

The article is a group report on the student management system, consisting of three parts. Part one introduces user requirements, ERD database design, and Site map. The second part shows the MVC pattern used in the project and screenshots of the website. The final part evaluates the site's strengths and weaknesses, and future improvements. In this project, the author only works with Student and Class function. Therefore, in this article, some part only introduces about things related to the above two functions.

II. User Requirement

As a/an	I want to...<perform some task>	So that I can...<achieve some goal>
Admin	Manage all account of all user	Create account for all Student, Teacher
	Manage all Course	Create New, Update information, delete, search
	Manage all Class	Create New, Update information, delete, search
	Manage all Student	Assign and remove student to class Update information, delete, search, create new
	Manage all Teacher	Create New, Update information, delete, search
	Manage all Room	Create New, Update information, delete, search
	Manage all Category	Create New, Update information, delete, search
Teacher	Check Course, Class, Room, Course Category, Student	See list course, class, student, Category and see detail course, class, Course Category Search Course, Class, Room, Course Category, Student, Teacher
Student	Check Course, Class, Room, Course Category,	See list course, class, Category and see detail course, class, Course Category

		Search Course, Class, Room, Course Category,
--	--	--

Figure 1: User requirement

III. System Design

1. Site map

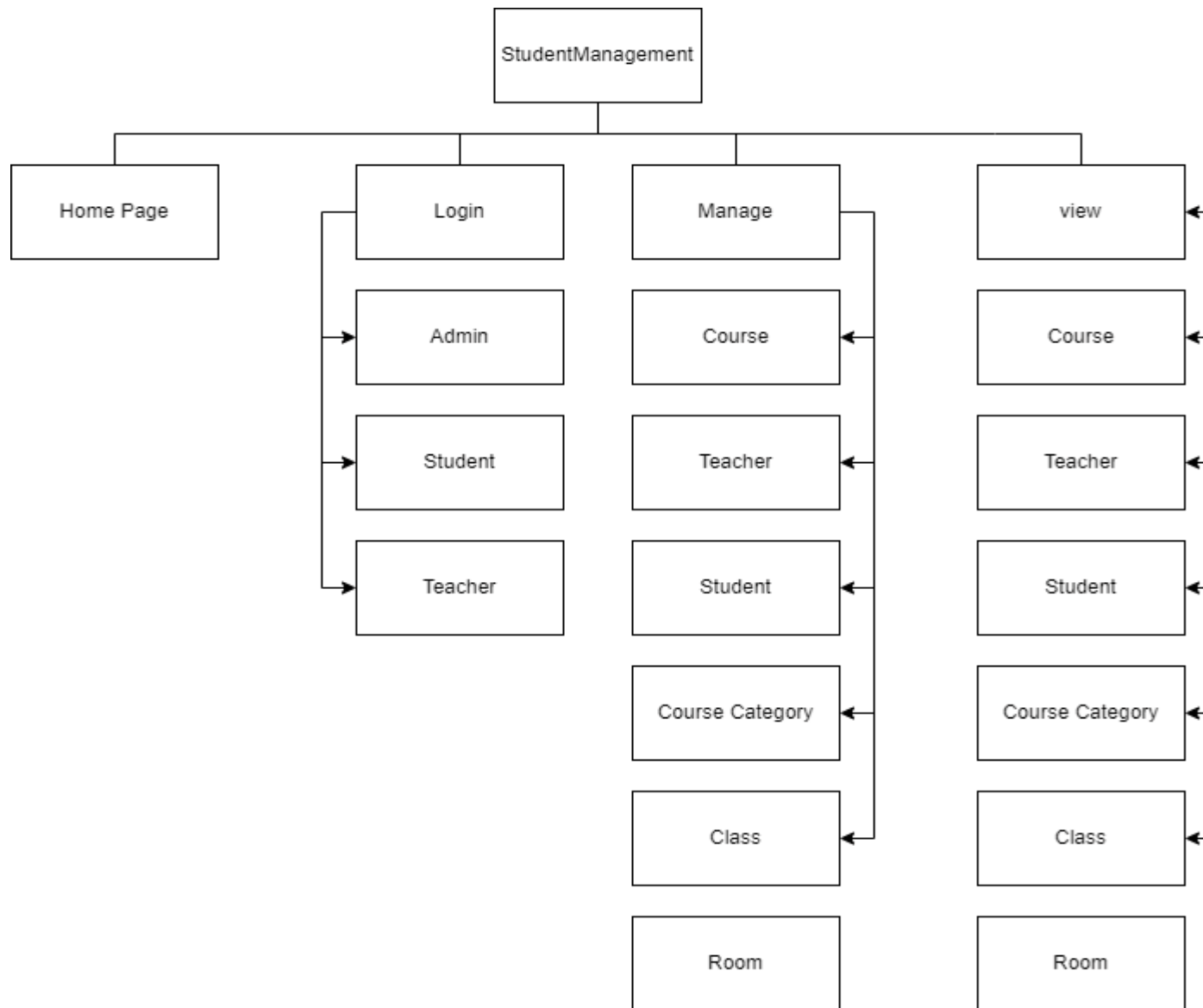


Figure 2: Site map

2. Entity Relationship Diagram

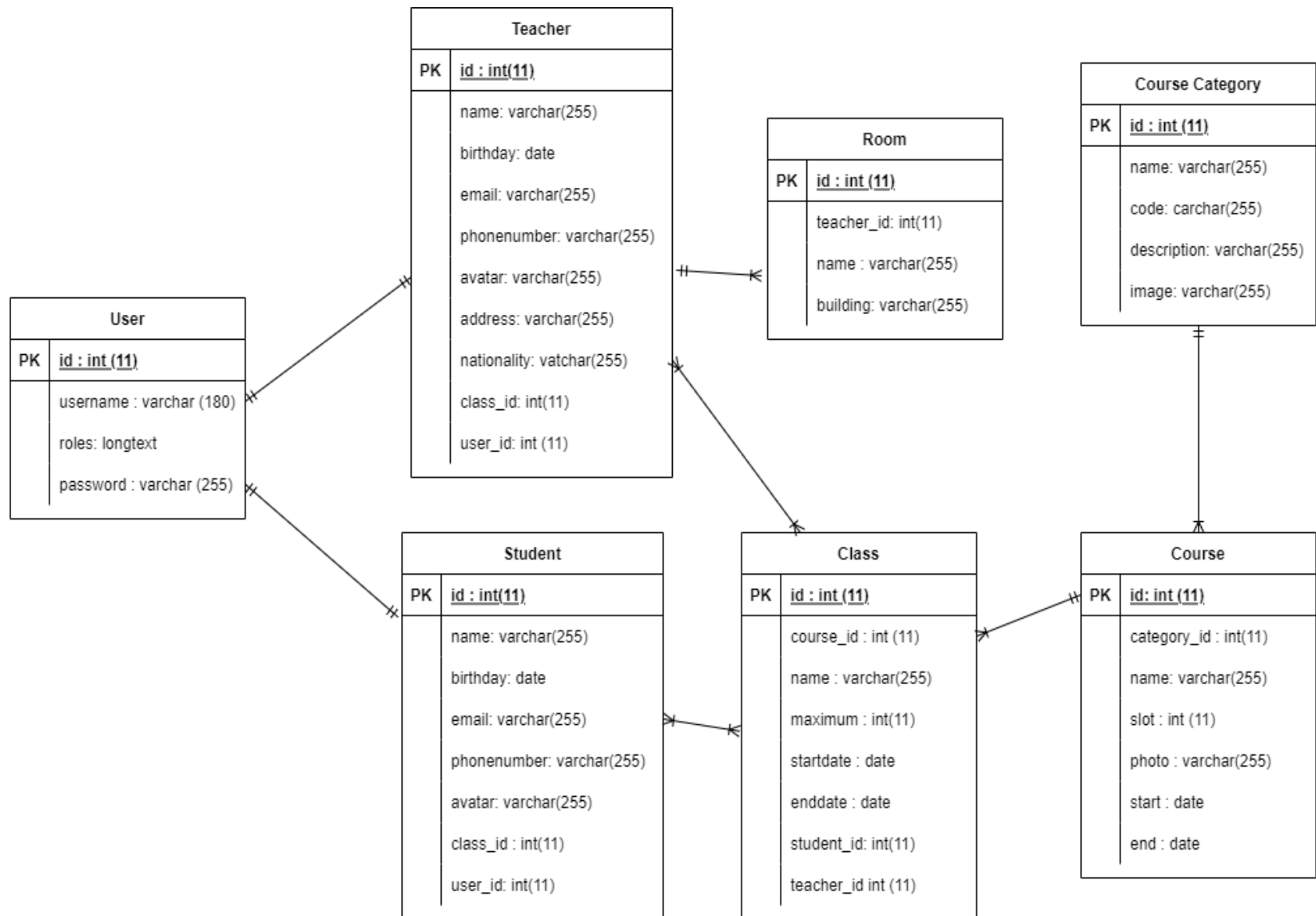


Figure 3: ERD diagram

IV. Implementation

1. Sample Source Code

The system uses the Symfony framework, so the image below shows the directory of the whole project that has been created with the Symfony command. Obviously, Symfony has divided folders according to the MVC pattern: model and controller in the src folder, and view in the templates folder.

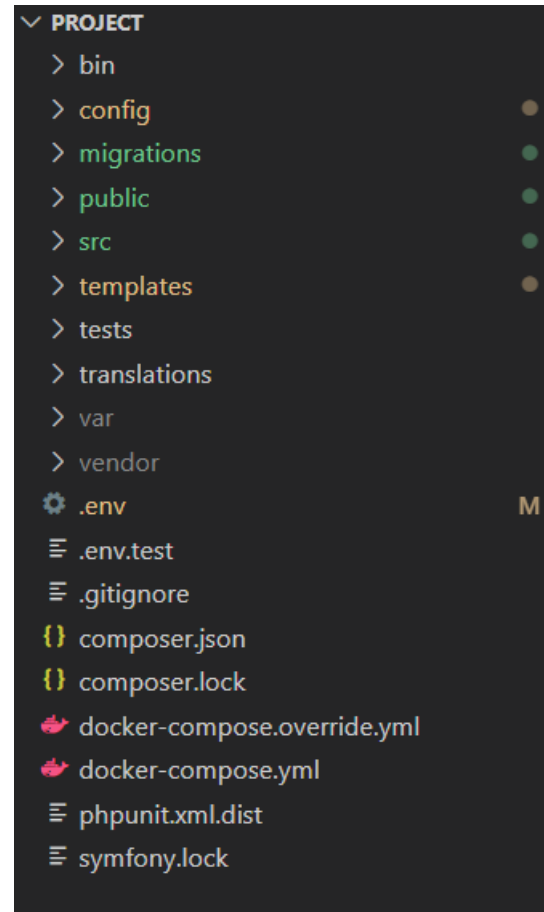


Figure 4: Source code folder

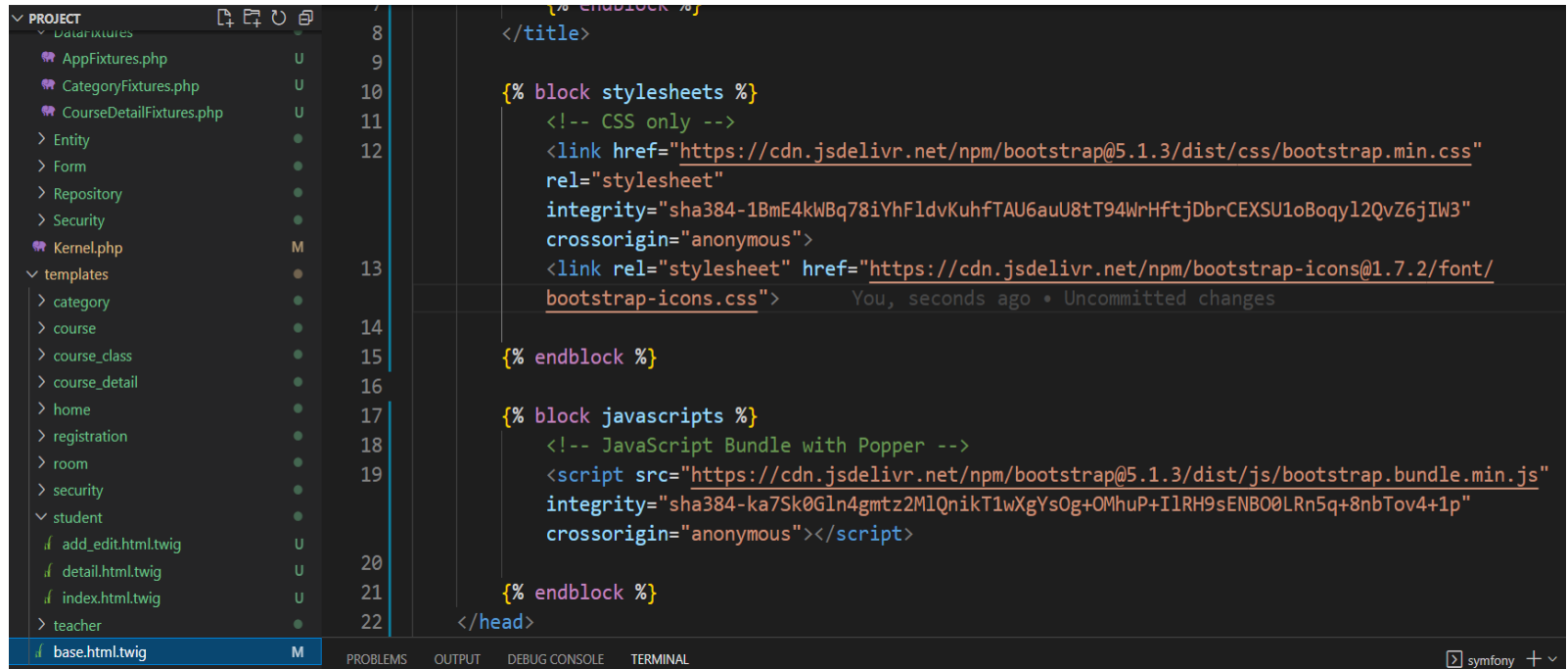
a. View

In the MVC model, the View represents the user interface. In symphony, this view is located in the "templates" folder.


```
▼ templates
  ▼ category
    if add.html.twig
    if detail.html.twig
    if edit.html.twig
    if index.html.twig
  ▼ course
    if add_edit.html.twig
    if detail.html.twig
    if index.html.twig
  ▼ course_class
    if add_edit.html.twig
    if detail.html.twig
    if index.html.twig
    if view_student.html.twig
    if view.html.twig
  > course_detail
  > home
  > registration
  > security
  > student
  if base.html.twig
```

Figure 5: View folder

The following are some example of view



```

7  {% endblock %}
8  </title>
9
10 {% block stylesheets %}
11     <!-- CSS only -->
12     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
13         rel="stylesheet"
14         integrity="sha384-1BmE4kWBq78iYhF1dvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
15         crossorigin="anonymous">
16     <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.7.2/font/
17         bootstrap-icons.css">
18
19 {% endblock %}
20
21 {% block javascripts %}
22     <!-- JavaScript Bundle with Popper -->
23     <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
24         integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
25         crossorigin="anonymous"></script>
26
27 {% endblock %}
28 </head>

```

base.html.twig M

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

symfony +

Figure 6: base.html.twig

PROJECT			
▼ DataFixtures		28	
AppFixtures.php	U	29	
CategoryFixtures.php	U	30	
CourseDetailFixtures.php	U	31	
> Entity	•	32	
> Form	•	33	
> Repository	•	34	
> Security	•	35	
Kernel.php	M		
▼ templates	•	36	
> category	•	37	
> course	•	38	
> course_class	•	39	
> course_detail	•	40	
> home	•	41	
> registration	•	42	
> room	•		
> security	•		
▼ student	•	43	
add_edit.html.twig	U	44	
detail.html.twig	U	45	
index.html.twig	U	46	

```

<tbody>
  <form action="{{ path('student_delete', {'id': 'checkbox'}) }}" method="post">
    {% for s in student %}
      <tr>
        <td><input type="checkbox" name="checkbox[]" id="" value="{{ s.id }}"></td>
        <td>
          <a class="text-decoration-none" href="{{ path('student_detail', {'id': s.id}) }}">{{ s.name }}</a>
        </td>
        <td>
          
        </td>
        <td>
          <button>
            <a class="text-decoration-none" href="{{ path('student_edit', {'id': s.id}) }}">Edit</a>
          </button>
        </td>
      </tr>
    {% endfor %}
  </tbody>

```

Figure 7: Student index

b. Controller

Basically, in the MVC model, the controller acts as a middle layer between the view and the model. It receives commands from the view and sends them to the model and vice versa.

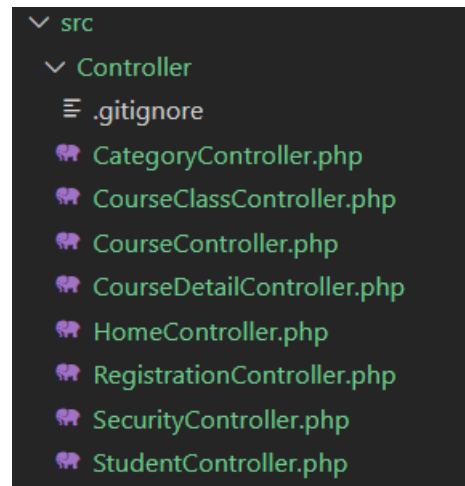


Figure 8: Controller folder

An example in the student's controller, two functions `studentIndex` and `studentDetail` are responsible for getting data from the `Student` table in the database and rendering it in the view.

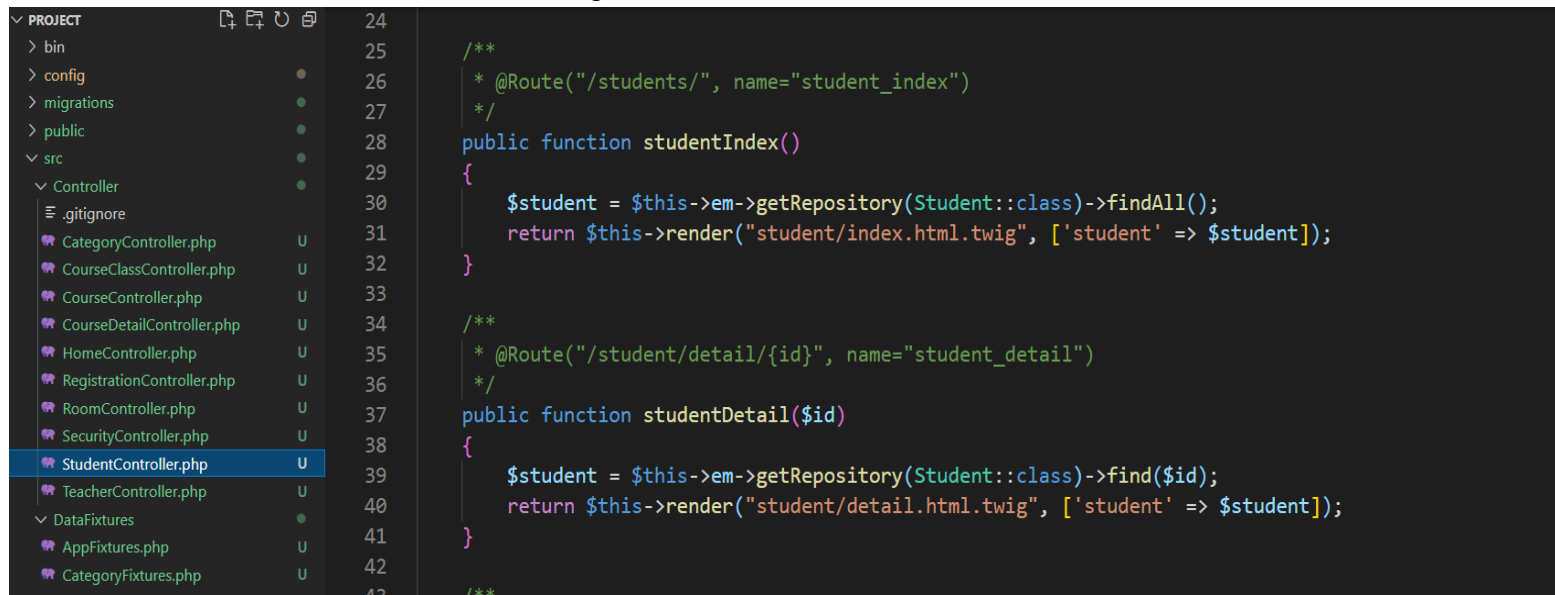


Figure 9: Student controller

c. Model

In simple terms, in the MVC pattern, this model handles and accesses operations in the database. In Symfony, the Model can be auto-generated with commands such as 'symfony make: entity'.

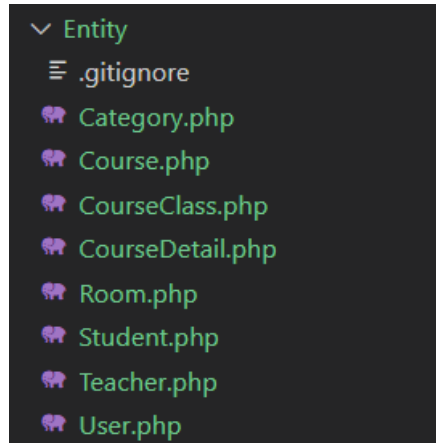


Figure 10.1: Entity folder

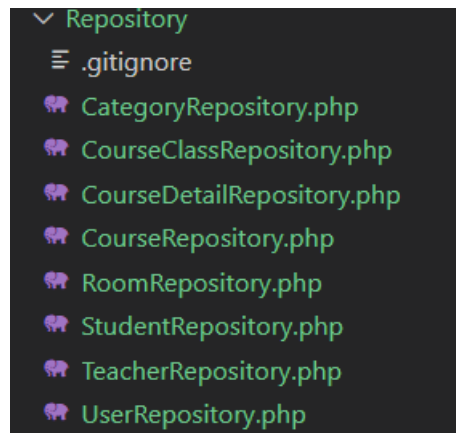
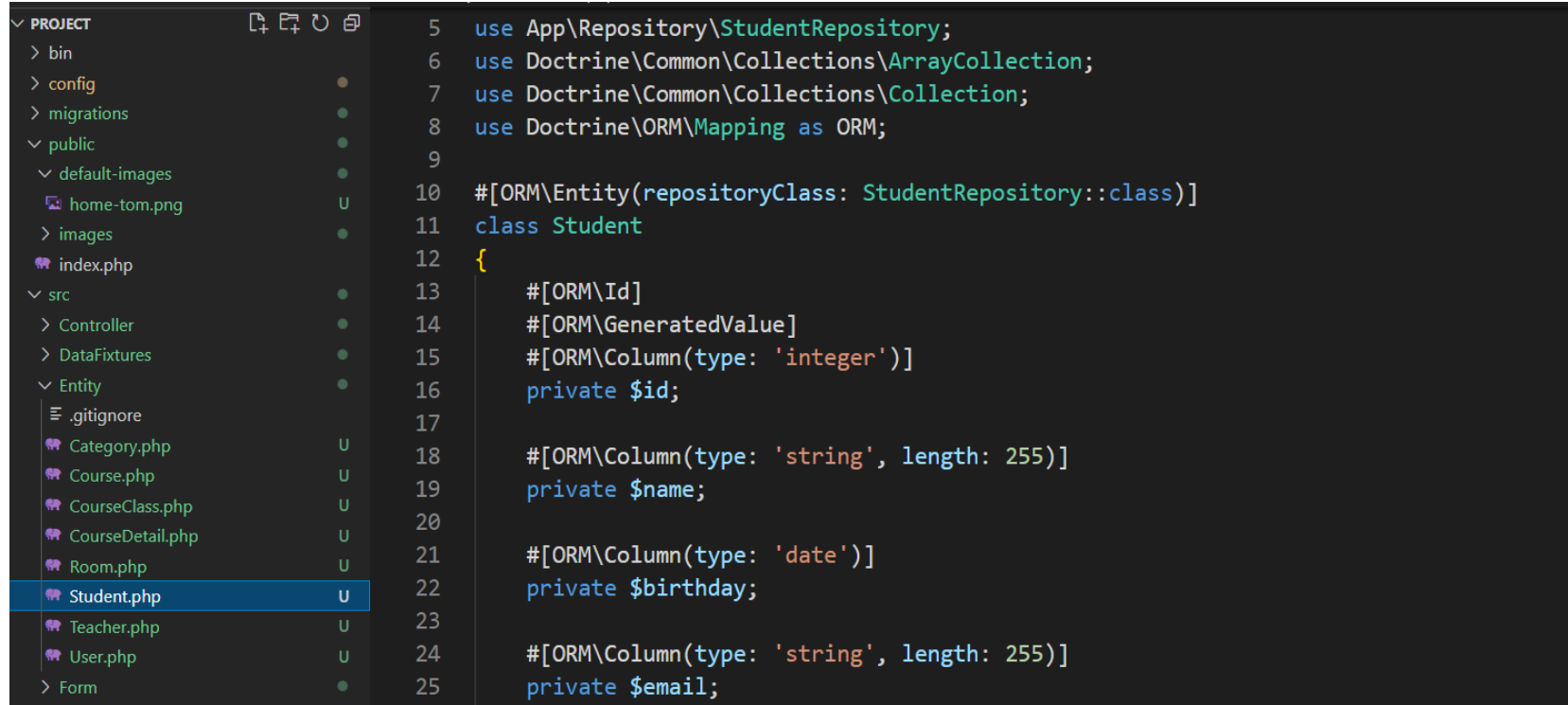


Figure 10.2: Repository folder



```
5 use App\Repository\StudentRepository;
6 use Doctrine\Common\Collections\ArrayCollection;
7 use Doctrine\Common\Collections\Collection;
8 use Doctrine\ORM\Mapping as ORM;
9
10 #[ORM\Entity(repositoryClass: StudentRepository::class)]
11 class Student
12 {
13     #[ORM\Id]
14     #[ORM\GeneratedValue]
15     #[ORM\Column(type: 'integer')]
16     private $id;
17
18     #[ORM\Column(type: 'string', length: 255)]
19     private $name;
20
21     #[ORM\Column(type: 'date')]
22     private $birthday;
23
24     #[ORM\Column(type: 'string', length: 255)]
25     private $email;
```

Figure 11: Student entity



```
47  
48  
49 */  
50  
51 /**  
52  * @return CourseClass[]  
53  */  
54 public function searchClass($value)  
55 {  
56     return $this->createQueryBuilder('entity')  
57         ->andWhere('entity.name LIKE :value')  
58         ->setParameter('value', '%' . $value . '%')  
59         ->getQuery()  
60         ->getResult();  
61 }  
62  
63 }
```

Figure 12: Student orm

2. Web screenshots



Student Managemet

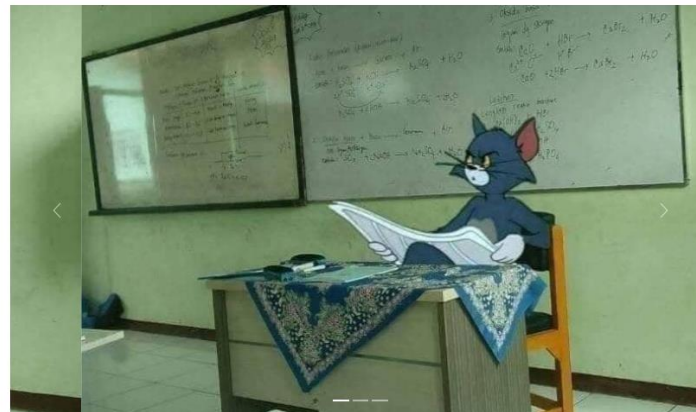
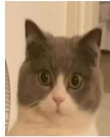





Figure 13: Home and navigation bar interface

+

search by name

<input type="checkbox"/> Check all	Student Name	Student Avatar	Actions
<input type="checkbox"/>	da12		Edit
<input type="checkbox"/>	dademo23	<div>Tôi nằm trên giường một nửa sau một ngày mệt mỏi vì nằm trên giường</div> 	Edit
<input type="checkbox"/>	dademo2345		Edit
<input type="checkbox"/>	da124545fd	<div>"Thập niên 15 phút của buổi trưa sẽ giúp bạn tỉnh táo"</div> <div>Tôi cần biết được một giấc ngủ 15 tiếng</div> 	Edit

Delete



Figure 14: View Student



ADD NEW STUDENT

Student Name

Birthday

Email

Phone Number

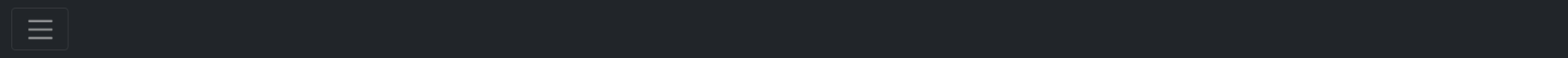
Student Image

 Không có tệp nào được chọn

Save

Figure 15: Add student





EDIT STUDENT

Student Name

da12

Birthday

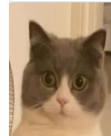
29 ▼ 11 ▼ 1998 ▼

Email

anhnd277@gmail.com

Phone Number

849383830




Student Image

Chọn tệp Không có tệp nào được chọn

Save

Figure 16: Update student



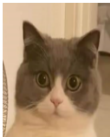


Student ID	Student Image	Student Name	Birthday	Email	Phone Number	Class	Actions
2		da12	29/11/2021	anhnd277@gmail.com	849383830	demo1 demo989 demomax2 demomax3 LOP MAU GIAO	 

Figure 17: Student detail

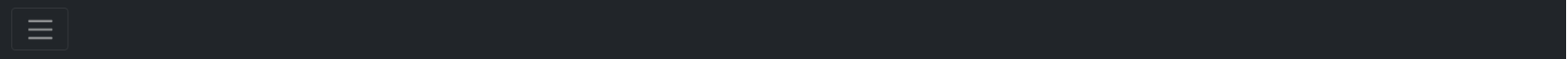


+

search by name

<input type="checkbox"/> Check all	Class Name	Course	Actions
<input type="checkbox"/>	demo1	GCH08079	✎
<input type="checkbox"/>	demo12	GCH0807fsdfds	✎
<input type="checkbox"/>	demo45	GCH0806	✎
<input type="checkbox"/>	demo989	GCH0806	✎
<input type="checkbox"/>	demo166666	GCH0807fsdfds	✎
<input type="checkbox"/>	demoMax	ACH0807	✎
<input type="checkbox"/>	demomax2	BCH0807	✎
<input type="checkbox"/>	demomax3	BCH0807	✎
<input type="checkbox"/>	LOP MAU GIAO	GCH0807fsdfds	✎
<input type="checkbox"/>	ljkjlklj	ACH0807	✎
Delete			


Figure 18: View class




ADD NEW CLASS

Class Name

Maximum Students

Start Date
 

Start Date
 

Course

Save



Figure 19: Add class



EDIT CLASS

Class Name

demo1

Maximum Students

30

Start Date

01/12/2021



Start Date

31/12/2021




Course

☐GCH0807fsdfs ☐GCH0806 ☒GCH08079 ☐GCH0807 ☐ACH0807 ☐BCH0807 ☐GCH08090909

Save



Figure 20: Update class






Class ID	Class Name	Course	Maximum Student	Number Students	Start Date	End Date	Students	Actions
4	demo1	GCH08079	30	3	01/12/2021	31/12/2021	view	 

Figure 21: Class detail



<input type="checkbox"/> Check all	Student ID	Student Name	Student Course
<input type="checkbox"/>	2	da12	demo1 demo989 demomax2 demomax3 LOP MAU GIAO
<input type="checkbox"/>	8	dademo23	demo1 demomax3 LOP MAU GIAO
<input type="checkbox"/>	9	dademo2345	demo1 LOP MAU GIAO
<input type="checkbox"/>	11	da124545fd	
Assign To Class			

Figure 22: Assign student to a class



[+ Assign Student To Class](#)

<input type="checkbox"/> Check all	Student ID	Student Name
<input type="checkbox"/>	2	da12
<input type="checkbox"/>	8	dademo23
<input type="checkbox"/>	9	dademo2345

Remove

[Click to Back](#)

Figure 23: Remove student from a class

V. Conclusion

1. Evaluate

Strength	Weaknesses
<ul style="list-style-type: none"> The website has met the minimum requirements of a student management system such as: adding, updating, editing, deleting searching. The system is built according to the MVC model (Model, View, and Controller). Website is easy to use, and all functions can be performed by no more than four operations. In the process of running the website, there are no errors, the functions work very efficiently. The website has authorization, access denial, and password encryption. Easy to modify and improve in the code. 	<ul style="list-style-type: none"> Low security. There are no special functions other than simple functions such as add, edit and delete. Simple and boring interface. Users other than admin can only view. Some tables in the database are not connected properly. Some features are not logical.

Figure 24: Pros and Cons

2. Future Improvement

In the future, the system will be added with more features such as grading function, student attendance, assignment submission, etc. In addition, the team will improve the user interface of the web to make them look better and more interactive. In the near future, the team will try to correct the shortcomings and irrationalities in the system to build a complete website.

VI. Appendix

Group member list	Role
Nguyen Duc Anh (GCH17051)	User Login, Student and Class function back-end and front-end
Pham Khue	Room and Teacher function back-end and front-end
Nghiem Cao Nhan	Course Category and Course function back-end and front-end

Figure 25: Member list

Link GitHub: <https://github.com/paulpham98/projectWeb>