Subject Name: Enterprise Web Software Development

Subject Code: COMP 1640

Tutor: Nguyen Tran Dinh Long

Student name: Nguyen Minh Son

Student ID: 001181266

Email: sonnmgch18055@fpt.edu.vn

Group 4		
Role	**Task**	**Member**
Product owner	Evaluate requirement, rate requirement	Pham Quang Minh
Database designer	Design database	Nguyen Minh Son Pham Quang Minh
Interface designer	Design Interface	Nguyen Van Tuan
Tester	Test	Bui Chi Huong Nguyen Trung Doan Khang
Scrum master	Make a plan, Track project progress, Division of work.	Nguyen Minh Son
Developer	Front-end	Nguyen Van Tuan
	Back-end	Nguyen Minh Son Pham Quang Minh

- URL Functional demo video: https://drive.google.com/file/d/1rKz8q-A3tbGWUBSLmm-2AFnk3hs1g_KD/view?usp=sharing
- URL Group repository: https://github.com/minhpham8502/Final?fbclid=IwAR3KCGLlbP_lpc-DKd5ebWbB_rAiCgnabMfzvmoUBAEHGQS_9DkQR7k7-vQ
- URL Documentation: https://docs.google.com/spreadsheets/d/1M7S_dYFzRcJOwniKhvdANdqrphx7SSXnKjeOSuHiG04/edit?usp=sharing
- URL Site: https://appcuatoi123.herokuapp.com/

Account for website to check:

List account				
Role	Email	Password	Faculty	FacultyID
Guest	guest@gmail.com	12345678	graphic design	GCHGD
Student	nguyenminhson09112000@gmail.com	12345678	graphic design	GCHGD
Marketing Coordinator	minhpqgch18572@fpt.edu.vn	12345678	graphic design	GCHGD
Marketing Manager	manager@gmail.com	12345678		
Admin	admin@gmail.com	12345678		

Figure 1: List of account

Table of Contents

Table of Figures

I. Introduction

This report will evaluate the project's overall product that we have completed and assess the team members according to each specific criteria. Evaluate the project implementation process when applying the Scrum method; besides, evaluate yourself and give lessons learned during project implementation.

II. Evaluation of product

1. Functionalities

The system's functions have been carefully tested and planned, so the website has a good performance, no problems when running the program. The website's functions are easy to use, focus on essential needs, and do not have cumbersome steps, causing annoyance and impatience for users. All parts work according to the requirements of the coursework:

ID	Role	Function
1.	All User	Login and Log Out
2.	Guest	View the selected reports.
3.	Student	Submit one or more articles as Word documents to the magazine.
		Upload high-quality images,e.g., Photographs.
		Agree to Terms and Conditions before they can submit.
4.	Marketing Coordinator	Receive email notification after student posting articles to the system
		Comment student's magazine within 14 days.
		Interact with students in their Faculty to edit and select those for publication.
5.	Marketing Manager	View all the selected contributions but cannot edit any
		Download all the selected contributions after the final closure date in a ZIP file.
		Set time for upload magazine
		Statistical analysis number of contributions per Faculty
6.	Administrator	CRUD faculty, Account (Guest, Student, Marketing Coordinator, Marketing Manager)
		Add Marketing Coordinator and Student to Faculty

Figure 2: List of function

2. Interface

The user interface is easy to use and can be suitable for devices with different screen sizes such as phones, laptops, and tablets. It is compatible with varying platforms of browser such as Microsoft Edge, google chrome, etc. The web interface is designed quite simply because all interfaces are written in HTML and do not use any bootstrap framework. So the design also takes quite a long time to complete.

2.1 Responsive

The system can be compatible with different devices such as phones, tablets, laptops. Besides, because the front-end code is written entirely manually and does not use the support library, bootstrap, the interfaces have some incomplete areas, and the phone interface still somewhat interfaces. The interface of the PC. Here are some illustrations:

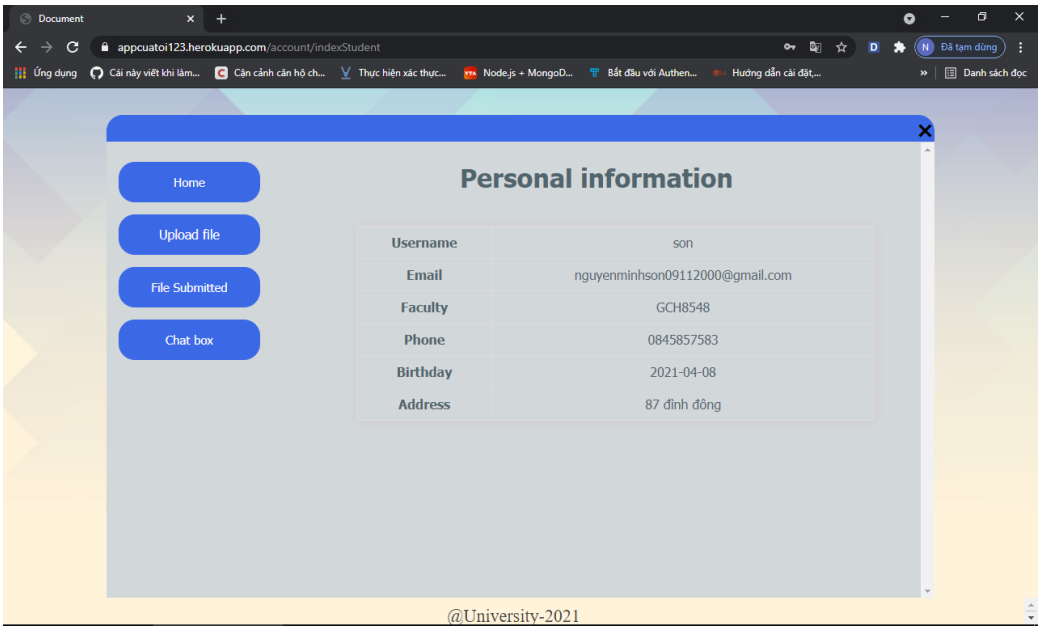

Figure 3: Interface on Laptop and PC

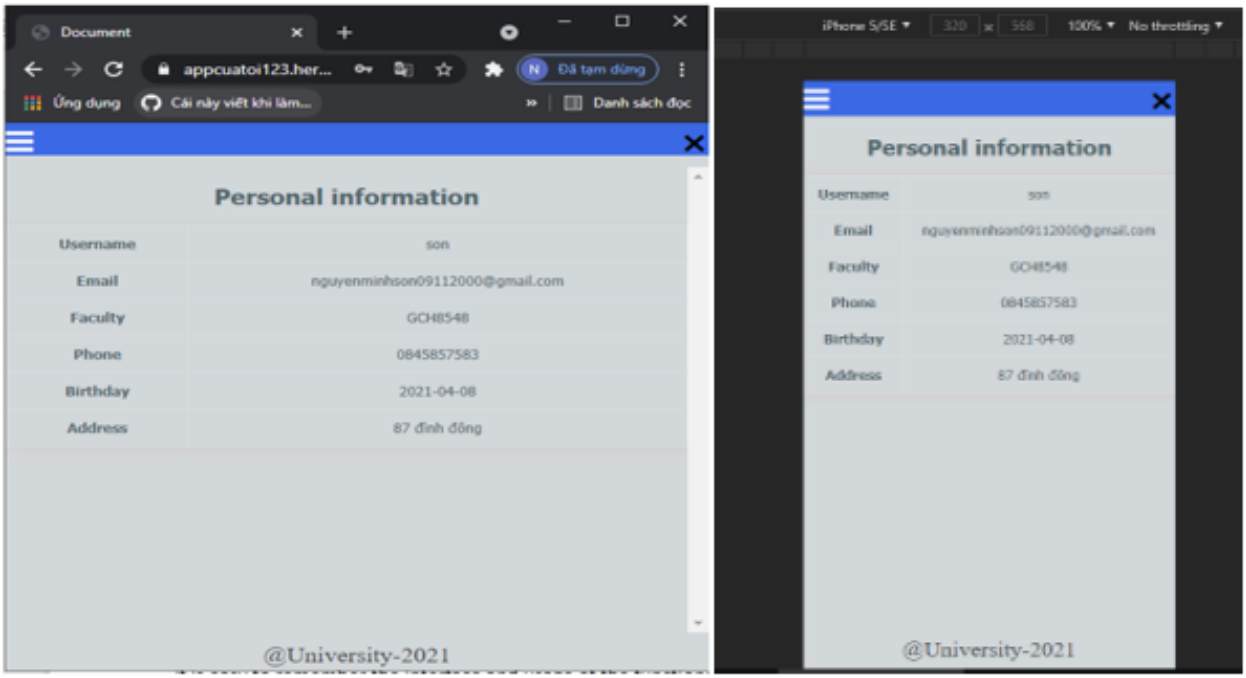

Figure 4: Interface on tablet and phone.

Some pictures of Responsive error:

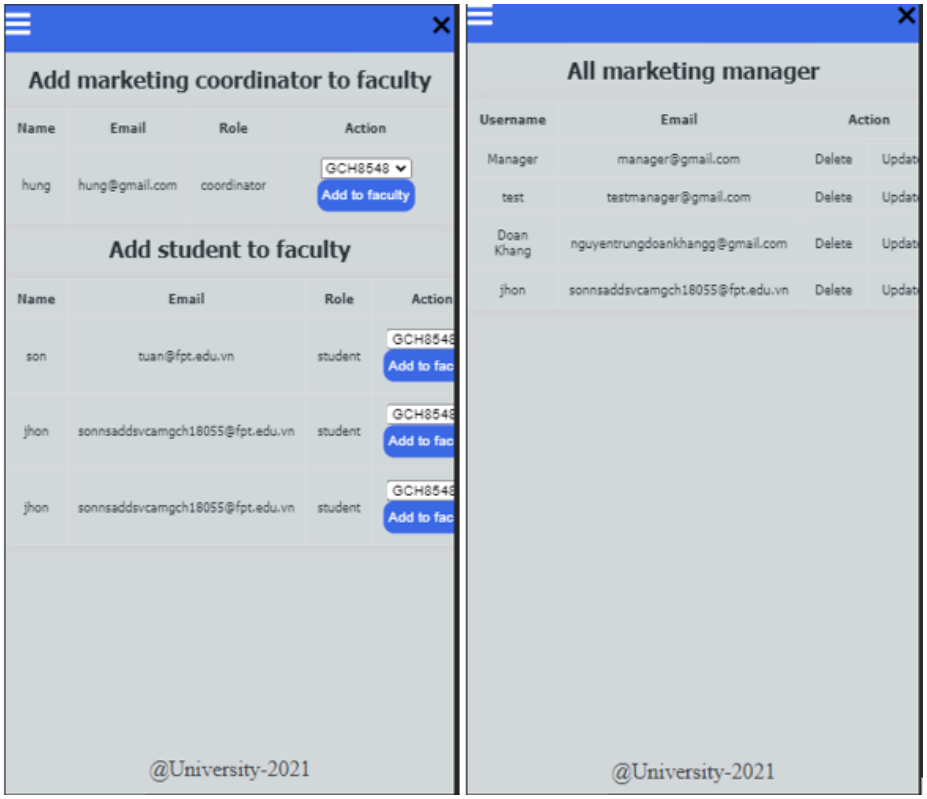

Figure 5: Interface on Phone

2.2 Usability

The system's interface is simple but highly usable. The system has a simple interface design, but this will help users quickly identify the functions and use them on the phone, PC. When using the system, users will find it very easy to use the functions of the system. The menu of functions for each role will be displayed on the left side of the PC and will be hidden again for the phone (click to bring up a menu). The name of the function recorded on the menu is clearly shown that function. As a result, users can quickly identify and use the functions. In addition, users can easily remember the steps performed because the function is optimized and avoid unnecessary steps that make users feel confused and confused. In the content imported and uploaded to the system, the system will notify the user if the content is used or the file is incorrectly formatted or has problems. Based on the hidden menu bar placed to the right (phone form), the functions can be easily recognized and used. The following are illustrations of student and manager roles.

Figure 6:Usability of student and manager

Besides, when a user encounters an error during use, the system will notice and guide the user. The following is an example of a student uploading to a file system other than the required system format

(Docx). After downloading, students will receive a notification about the type of file the system accepts and other conditions for uploading the file to the system.

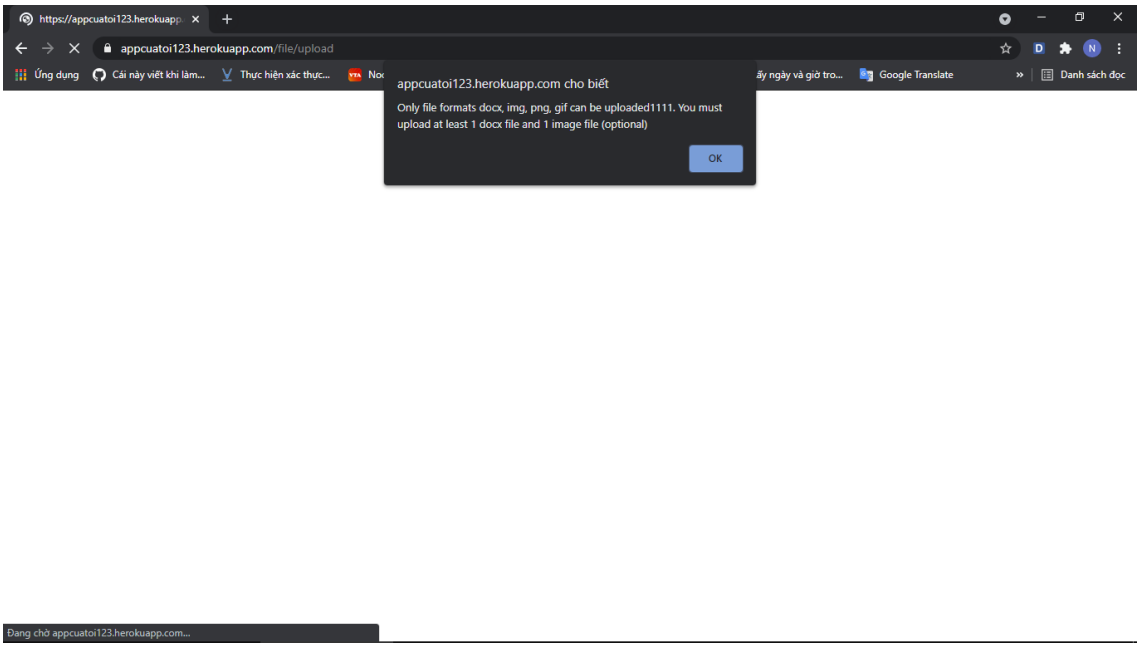

Figure 7: Check type of file

2.3 Accessibility

- The website is designed to help people with low vision. The fonts are big and clear for the user to see.
- Important information will be in bold for users to pay more attention.
- In addition, the website also integrates the use of the keyboard to use in case the user cannot use the computer mouse.

3. Programming model and database

During project construction, we chose the programming model Model - View - Controller (MVC). Developed on this model, we were able to divide the layout of the system clearly. Thanks to this feature of the MVC model, the system's construction is very convenient and flexible. During the implementation, we can quickly review and upgrade the system thanks to the MVC model's clarity. For the database, we chose the database as MongoDB (NoSql) because this is a database that has diverse data query capabilities suitable for the project and has performance (query speed problem) good (Pedamkar, 2020). Applying MongoDB database size will help your website increase performance thanks to the fast querying capabilities of MongoDB.

4. Security

The website has good security. When creating an account, important information such as passwords will be encrypted and then saved to the database. This will keep the user account information secure and prevent it from being stolen. To encrypt passwords, the system uses Bcrypt to encrypt data and uses Salt to authenticate data.

```
let signUpController = async(req,res)=>{
    try {
        let username = req.body.username;
        let password = req.body.password;
        let email = req.body.email;
        if(username && password){
            const salt = bcrypt.genSaltSync(saltRounds);
            const hash = bcrypt.hashSync(password, salt);
            AccountModel.create({
                username,
                password :hash,
                email
            });
        }
```

Figure 8: Sign-up an account

To increase protection for the website, the system has used JSON Web Token to decentralize. When logging in, the system will authenticate the login information. If the login is successful, the system will return an encrypted token containing the data _id of that account in the database. After the token is registered and stored in a cookie, the limit is one day to avoid attacks.

```
let token = jwt.sign({_id: req.body._id},'minh');
res.cookie("token",token,{maxAge: 60*60*10000});
```

Figure 9: Create a token and save on cookies

To decentralize, the server-side will take the token in the cookie and encrypt the token to get the _id of the account and check the system's rights. In the system, we use intermediary functions to check the user information when logging in. By re-encrypting the token code we have registered when the user logs in, the system will get the id of that account and check. If that account exists, the system will go to the next step. For example, the checkAuth () function is used to check the user account information.

```
let checkAuth = async (req,res,next)=>{
    try {
        var token = req.cookies.token || req.body.token
        let decodeAccount = jwt.verify(token,'minh')
        let user = await getUserById(decodeAccount._id)
        if(user){
            req.userLocal = user;
            next();
        }else{
            return res.status(400).json({
                message : "tk k ton tai",
                status: 400,
                error : true,
            })
        }
    } catch (error) {
        res.status(500).json({
            message : "hay dang nhap",
            status: 500,
            error : true
        },
        res.redirect('/'))
    }
}
```

Figure 10: Function check account information

To prevent users from exceeding their authority when using the system, the system also has functions to check the user's role against the system. Based on the token stored in the cookie, the system will reverse-encode the token to see the user id information and check the person's permissions on the database. If right, the system will let go. If the permissions are not correct, the system will issue a notice.

```
let checkAdmin = (req,res,next)=>{
    if (req.userLocal.role === "admin"){
        next()
    }else{
        return res.status(400).json({
            message : "no permission",
            status: 400,
            error : true,
        })
    }
}
```

Figure 11: Check role and permission

Example: A user wants to enter and perform the admin functions, that user will have to go through 2 stages of testing. It's about checking account information and checking permissions. If not, the system will report an error and not allow access to the site.

```
adminRoute.use(checkAuth);
adminRoute.use(checkAdmin);
//create account
adminRoute.get('/createAccount', AdminController.createAccount)
adminRoute.post('/docreateAccount', AdminController.docreateAccount)

//Add to faculty
adminRoute.get('/addtoFaculty', AdminController.addtoFaculty)
adminRoute.post('/doadddtoFaculty:id', AdminController.doadddtoFaculty)
```

Figure 12: Check permissions for each page

III. Evaluation of process

Firstly, when implementing the project, we held a meeting to agree on issues such as technology development, member capacity assessment, and customer requirements (creating a product back-log.). The creation of a product back-log is intended to define user functions and to evaluate difficulty for those requirements. This step is very important because it helps us to plan and divide the work properly. Assessing the problem of functions will help with meticulous and rational planning. After creating the product back-log, we proceed to make a mini-plan called sprints. Based on the back-log, we divided the plan into four sprints (each sprint lasted nine days). This Division of this plan will help us grasp the progress of project capture and easy management. Based on the difficulty and importance of the requirements, we have appropriately divided the work for each sprint.

We divide the critical work first and have moderate difficulty - easy for early sprints. Later sprints will be divided into jobs of the severe problem and low importance. As a result, we will always ensure that the website has the necessary functions and the program can function correctly. Besides, in every sprint, we often have to open regular meetings (2 meetings). Meetings are usually held for 2-3 hours. Because of this Scrum ethos, we always keep track of project progress and address the risks we may face regularly. The project was successful as we expected. We often discuss the work as done (evaluation of the interface, how the function works, whether to improve those functions, etc.) and discuss the tools—preparing to do next. During the meetings, all group members have to give opinions about the product, the problem the group has encountered, and everyone gathers together to come up with the best solution to the problem. After each meeting, the whole group will agree on goals, the following work to be achieved until the next meeting is held. Thanks to that, the project progress always goes according to the plan.

Thanks to the continuously held meetings and the constant information exchange we ask with our instructors, the product is constantly improved to suit users' best. During our discussions with the instructor, we encountered several change requests, such as the manager having to set submission deadlines for all faculties and automatic time additions for editing deadlines and timetable comments. Graph (moving from a bar graph to a circle for easy expression) but thanks to regularly held meetings, we successfully organized and handled the work.

IV. Evaluation of myself

During project implementation with the team, I hold the roles of Scrum master, design database, developer (back-end) of the team. Taking on three roles has helped me interact with my group members a lot, especially from the position of the Scrum Master. I created a session to review the members' capabilities and then divide the tasks appropriately. During project implementation, for the project to always be on schedule and no problems arise, I always have to create meetings to do work statistics and review project progress. During meetings, I always have to prepare issues around the product (design, functionality optimization, etc.) to make content for the meetings and gather opinions from team members. Each other agreed to come up with the best solution to the problem to be solved. In order not to pressure during the meeting, I created a comfortable atmosphere during the meeting. During the meeting, I always write down the necessary notes (tasks to do, change products, whoever is busy, what date, etc.) during the meeting and summarize with each member at the end of the meeting. This will make the work more efficient and avoid missing out on the job. As a result, our team is always on track for each sprint.

As a designer database and developer for the back-end, I worked with Pham Quang Minh. During our work, we exchanged information and helped each other a lot. Whenever the other party has difficulty in the code back end, I open conversations through google meetings to exchange and help. In database design, I helped Minh to understand the MongoDB database and NoSQL relationship better.

The back-end code division work is divided voluntarily. This helps to make the coding work more comfortable and without the extrusion of responsibility and mutual incompatibility.

Technically, I have learned and practiced the technologies from the previous projects, so the initial development of the project is quite fast. After doing the basic building steps of the project (login, create an account), I ran into some difficulty in the process of building the client requirements. To solve the problem, I had to learn about a number of libraries on the Internet

to refer and adapt them to suit the system's circumstances. Learning the libraries has helped me a lot with the development of the system. In asking students and coordinators to receive emails when students upload articles to the system, I encountered difficulty and disorientation. This is my first time meeting this automated mail request build, but by doing some research through the Internet, I got to know the nodemailer library, and I tried to apply it and adapt it to the system context and was successful. Besides, while performing the statistical view function for the project's Marketing Manager role, I learned about MongoDB's query statements for the most accurate data statistics. Through learning and optimizing how to collect the data, I have completed the data statistics of each Faculty and the representation on the column chart; after talking with Mentor, I changed the column chart to a chart. Circular so that system users can better represent the significance of the statistics.

V. Evaluation of each team member

5=perfection (100%) 4=almost always (95-99%) 3=frequently (85-94%) 2=sometimes (75-84%) 1= seldom (65-74%) 0= almost never (0-64%)

Name	Minh	Tuan	Huong	Khang
Behaviors	Place a 1-5 score in each box			
Arrives promptly to group meetings	5	5	3	3
Attends group meetings	4	4	3	3
Completes assigned work on time	4	3	4	4
Is fully prepared for each group meeting	5	4	2	2
Notifies in advance of late/missed meetings	5	4	3	3
Subtotal	4.4	4	3	3
Teamwork				
Participates in group discussions	5	5	4	3
Offers ideas/suggestions	5	4	3	4
Is open to criticism/questioning	5	4	3	3
Acknowledges expertise of others	4	4	4	4

Figure 13: Evaluation of each team member

Name	Minh	Tuan	Huong	Khang
Technical Competency				
Can explain own work to others	4	4	3	4
Corrects own technical problems	4	4	3	4
Conducts research as needed	4	4	4	4
Accurately generates statistical data	4	4	4	4
Participates in "number crunching"	4	4	4	4
Subtotal	4	4	3.6	4
Work Product				
Writes in agreed upon style	4	4	3	4
Submitted work is complete	4	4	4	4
Written work is professional	4	4	3	3
Oral presentation is well rehearsed	4	4	4	4
Output consistent with expectations	4	4	4	4
Subtotal	4	4	3.6	3.8

Figure 14: Evaluation of each team member

Pham Quang Minh (Back-end Developer, Database designer, Product Owner)

- Behavior: He always attends his appointments on time and always raises ideas and problems he encounters during project implementation. The work he was assigned was completed and submitted on time.
- Teamwork: He is ready to help his teammates in times of difficulty. During the back-end coding process, he helped me a lot when I was having trouble. During the meetings, he regularly gives ideas to solve the problems faced by the group.
- Technical Competency: His skills in technologies are pretty good due to his research and practice. In writing back-end code to build functions for the project, he encountered a difficult problem but did research, learn, and solve the problem effectively.
- Work Product: He works and completes work on time.
- Overall Evaluation: He is an important member of our team. He has made a great contribution to the success of the project.

Nguyen Van Tuan (Front-end Developer, Interface Designer)

- Behavior: She always attends appointments and doesn't miss a session. He is always ready to help people and take a job from the scrum master. His work is always completed on time.
- Teamwork: He has good teamwork skills. He is constantly learning and listening to others. He always comments on his part and learns from those comments.
- Technical Competency: He has a fair knowledge of the front-end code section. During the construction of the interface for the project, he learned some of the features that the Scrum Master assigned to improve the website's usability and completed it very well.
- Work Product: He always gives the product on time for the project.
- Overall Evaluation: During the project implementation, he gave a lot of solutions to the team's problems.

Bui Chi Huong (Tester)

- Behavior: He participates in most group meetings. He sometimes has a few comments about getting a job from a scrum master. His work is always completed on time.
- Teamwork: He has good teamwork skills. He often gives his opinion on the problems the group is having.
- Technical Competency: He is knowledgeable in the Testing section. During the test, we learned more information and applied that information for a complete test.
- Work Product: He always has a plan to test and deliver the results on time.
- Overall Evaluation: As a tester, he plays a major role in perfecting the product.

Nguyen Trung Doan Khang (Tester)

- Behavior: He participates in most group meetings. He is always ready to take on the assigned jobs. He always gives his opinion in the sessions in which he participates.
- Teamwork: He has good teamwork skills. Be willing to perform and accept the ideas of others. He has the ability to coordinate well with others to get the best test done.
- Technical Competency: He has a fair knowledge of test planning and execution. During the testing process, he planned the test quite reasonably and made the product perfect.
- Work Product: He always has a reasonable test plan and gives the results on time.
- Overall Evaluation: As a tester, he plays a big role in making the product perfect and avoiding unnecessary risks.

VI. Lesson learns

After applying agile to the project and becoming a scrum master, I learned a lot during the project building process. As a scrum master, I learned how to manage and develop a project. I had to work with all the team members and find out about each other's abilities. Since then, I have arranged and assigned work for each member in a reasonable way based on each person's capacity. To always keep the project schedule and things under control, I had to create many meetings. During the implementation of the agile approach, agile strengths have been found in developing projects. It's easy to catch project progress, clear work division, etc. With the agile method, I have successfully managed the project, and everything is always right as we hope. These role experiences help me gain experience, improve skills such as teamwork, project management, problem-solving, etc., and this will help me become more stable and experienced. More when joining new projects in the future.

VII. Conclusion

In the report, I gave my personal opinion about the project's products. Besides, I evaluated each team member according to teamwork, behavioral expression, technical competence, and work productivity. This gives me a better understanding of how Scrum works and how people get involved in the project. Through implementing projects with the Scrum model, I learned many lessons for myself, which will be a great advantage when I participate in big projects in the future.

VIII. References

Pedamkar, P., 2020. *educba.* [Online]

Available at: https://www.educba.com/mongodb-vs-sql-server/