# Database

1. **Site Design**
   1. **Usability Heuristics**

We followed Nielsen Heuristics throughout the design process to ensure the system met industry-standard usability criteria. Stick to what users will be used to and don't complicate the user interface, thus maintaining the design is simple, consistent, and easy to use.

* + 1. **Nielsen’s heuristics, (Jakob Nielsen 1995)**

According to (Nielsen, 1994), 10 Usability Heuristics for User Interface Design:

* **Visibility of System Status**

It is crucial to maintain system simplicity by making sure users are constantly aware of the section of the system they are in and the website that is presently being used by them. Therefore, each page should have a title that appropriately conveys to consumers. In addition, utilizing notifications efficiently to verify the accomplishment of certain tasks.

* **Match Between the System and the Real World**

The design needs to be user-friendly. Employ recognizable language, phrases, and ideas instead of internal use jargon with the user. Make information seem in a logical and natural sequence by adhering to real-world norms. The contents of a page should be arranged logically. For example, when a user fills out the submission form to add their contribution to the system, the form's elements should be displayed in a way that makes sense to them. This means that the form should begin with the contribution title at the top and end with the user clicking "Submit" after verifying that they have read the terms and conditions. It could be confusing to place the upload button before the user has chosen which file to upload.

* **User Control and Freedom**

Users may make a mistake clicking on a button or link and they want to undo it, they can always navigate back to the home page or undo it no matter where they are in the app, by clicking the Home button on the header of the system or the Cancel button. When users can easily back out of a process or undo an action, it gives the user a sense of freedom and ease in using the system. Exiting allows the user to maintain control of the system and avoid getting stuck or confused.

* **Consistency and Standards**

All items on the pages will always be in the same place, regardless of the part of the application the user is in, thanks to the application's consistent design, which is enforced by the usage of Bootstrap. As a result, the user will always be aware of what will happen when they click the button.

* **Error Prevention**

When feasible, validation will be used to show the user which pieces are missing and will result in an error. For instance, when a user tries to log in, mandatory fields on the login screen that are not filled out should be highlighted. ensuring that users are aware of their errors and will improve their system experience as a result.

* **Recognition Rather than Recall**

Reduce the amount of memory used by the user by making options, actions, and items visible. Information shouldn't need to be remembered while navigating between different areas of the UI. When needed, information that is necessary to use the design (such as menu items or field labels) should be clearly visible or accessible. This will be especially helpful when uploading an article because it will cover every aspect of the system's operation and users aren’t therefore required to guess how to carry out certain tasks.

* **Flexibility and Efficiency of Use**

To ensure that the design can accommodate both beginner and expert users, shortcuts that are concealed from novice users may expedite the interaction for the expert user. Allow users to tailor frequent actions. Flexible processes can be implemented in a variety of ways, allowing users to select the one that best suits their needs.

* **Aesthetic and Minimalist Design**

Maintaining a system as user-friendly as feasible will be made easier by sticking with a clean, minimalist, and straightforward user interface design. By keeping the screen from becoming too crowded, it will be possible to complete activities and explore pages without overwhelming the user with unnecessary information and graphics. Because of this, our system will take a minimalistic stance, showing just the information that is necessary to complete the current task. Additionally, a color scheme will be carefully considered to make sure the selected colors work well together to create an aesthetically beautiful design.

* **Help Users Recognize, Diagnose, and Recover from Errors**

Error messages should accurately describe the issue and offer a workable remedy (no error codes). To aid users in seeing and identifying these error signals, precisely indicate the problem, and constructively suggest a solution. These error messages should also be presented with visual treatments that will help users notice and recognize them.

* **Help and Documentation**

Little textual pieces of help material will be supplied when needed to avoid overwhelming the user with extraneous information, particularly if it has nothing to do with the activity at hand. Consequently, the page where the user is doing the task that is specified in the text fragment will show instructions.

* 1. **Usability**

High system usability is the primary emphasis of the development of our website's front-end design, back-end code structure, and other components. The website is both platform-neutral and user-friendly thanks to its responsive design. Users may utilize the system on any device because the website style works with all browsers and screen sizes. We test our website across a range of devices, all of which have distinct screen sizes.

* 1. **Accessibility**

The website's color design was chosen with human-computer interface (HCI) elements in mind. The following are some of the most important elements that enable users to accomplish tasks quickly and make our system website accessible to a wide range of audiences.

* + 1. **Color factor**

According to recent studies, the hues purple, pink, and red are more calming. The many sections of the website, such as the buttons, tabs, links, etc., aid in helping people remember it (Elliot & Maier, 2013).

* + 1. **Color scheme for color blind people**

According to (CRAVIT, 2022),Color blindness, commonly referred to as color vision deficit (CVD), is the inability to discriminate between various hues. It happens when the retina, which is light-sensitive tissue at the back of the eye, is unable to react to changes in light wavelengths appropriately, which allows humans to see color differences. Purple is the color that works best for all kinds of eyesight because of this. Purple is still able to distinguish between several online page elements, including the color of the headline, current tab, and navigation bar. The below-mentioned picture below shows the result of there are three distinct types of color blindness (figure 1):

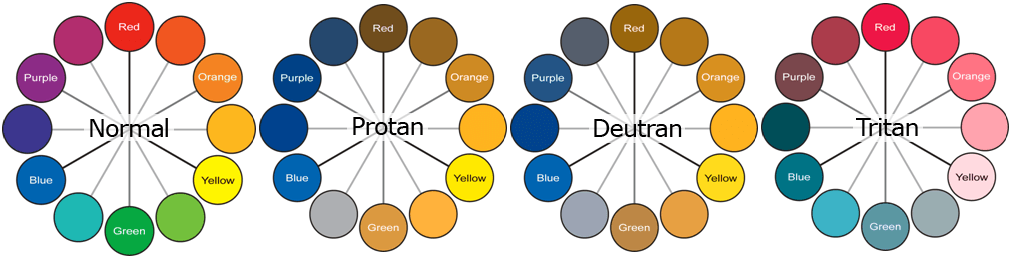


Figure 1: Three distinct types of color blindness

* 1. **House Style Consistency**

Our system was designed with Bootstrap, which not only enforces a uniform style but also provides a number of advantages, including ease of maintainability and reusability, as will be covered below.

* + 1. **Maintainability**

Should the stakeholders need to make changes, it would be simple to alter the website design rather than having to duplicate the modifications across every page that has to be changed. enabling effective site design modifications to be performed as needed and at the appropriate times. If a stakeholder wants to alter their logo, for instance, because the university's branding has changed, this may be done in one place and will update the logo on all the pages that utilize it.

* + 1. **Reusability**

It allowed for the construction of a lightweight application design that wouldn't be hampered by the end client's bandwidth limits. It also prevented code duplication and allowed the reuse of the same design for other website components, ensuring consistency throughout the website. Additionally, a mobile site was implemented thanks to the usage of Bootstrap.

* 1. **Responsive**

In contrast to more conventional platforms like desktops or laptops, people are increasingly using mobile devices to browse websites. Our system is designed to meet the needs of this particular market. As a result, Bootstrap has been used to provide compatibility for devices with smaller displays. The usage of a responsive layout consistently across our system—which enables a completely scaled design, irrespective of the user's screen size—will be covered in the section following.

* + 1. **Use of Bootstrap**

According to (Bootstrap, 2024),Bootstrap is a free and open-source framework for web development. Its purpose is to facilitate the creation of mobile-first, responsive websites by offering a set of template designs' syntax. Put another way, because web developers don't have to bother about basic commands and functions, Bootstrap makes website development faster. It is made up of scripts that are based on HTML, CSS, and JS that handle many aspects and features of web design. Bootstrap CSS served as the project's main hub for all user-visible design elements.

* 1. **Other technologies utilized**

In the execution of our system design, we also made use of Font Awesome, a collection of unique vector-based icons that are transformed into online fonts. With 675 icons, the most recent version (6.5.1) is available for use in any project under the terms of the GPL. The vector-based icons appear better on all devices since they are resolution independent.

* 1. **Other aspect considered**

When developing a project using ASP.NET API in combination with Angular, there are several other aspects to consider to ensure the best performance, security and user experience which are listed below:

* 1. **Security**

Authentication and authorization are two important factors to consider when developing a web application. Ensure that we apply security measures such as HTTPS, user authentication, and session management. ASP.NET API can integrate authentication mechanisms such as JWT (JSON Web Tokens) or OAuth to ensure that requests are served only to authenticated and authorized users.

* 1. **State Management**

In the Angular application, we need to consider how to manage the application state, especially when working with dynamic data from the API. Using tools like RxJS to manage data streams or state management libraries like NgRx can help you organize code and minimize issues related to state management.

* 1. **Performance**

Ensure that our application is optimized for fast loading and smooth operation. Consider using techniques such as lazy loading in Angular to load parts of the application on demand, and use caching techniques to minimize the number of requests to the API.

* 1. **Interaction with the API**

Define how the Angular application interacts with API. Use HTTP Interceptors in Angular to handle global HTTP requests such as authentication or error handling. Ensure that HTTP requests are handled efficiently and code is well organized.

* 1. **Session and Resource Management**

For applications with session requirements, we need to consider how to manage user sessions. ASP.NET provides session management mechanisms such as Session or session less work. Ensure that we choose the appropriate method according to the application's needs.

* 1. **Testing**

Working with ASP.NET API and Angular requires ensuring that both parts of the application are thoroughly tested. Use automated testing techniques such as unit tests, integration tests, and end-to-end tests to ensure the quality of code and application functionality.

* 1. **Information Architecture**

Information Architecture (IA) is an indispensable aspect of website design, playing a crucial role in crafting a seamless and user-friendly online experience. Within our information architecture framework, we have established a system comprising five primary roles: Guest, Student, Coordinator, Manager, and Admin, each meticulously designed with unique interfaces and functionalities (refer to figure 2). All users are required to log in through the login interface to ensure security and easy management. In case of unsuccessful login attempts, the system provides error notifications and redirects users back to the login page for retrying. However, upon successful authentication, each role is directed to a distinct interface, optimized to enhance their experience on the website. For Guests, their journey begins with the homepage, where they can easily navigate to other pages. Moving on to the Student role, they also start with the homepage, which is uniquely designed to optimize their experience compared to Guests, aiming for a more professional user experience. From this homepage, Students can navigate to other pages effortlessly. Next, Coordinators start with the Dashboard, where they can easily navigate to other pages to perform the desired functions. Following that, Managers begin with the homepage, which is tailored to suit their Managerial roles, allowing them to navigate to other pages within their role easily. Finally, Admins start with the Dashboard, where they can navigate to pages for managing and analyzing the website. Each role allows users to access their profile management page and log out of the system at any time. From the provided information, it's evident that our IA is meticulously designed to cater to the diverse needs of users, ensuring a smooth and efficient browsing experience tailored to each role.

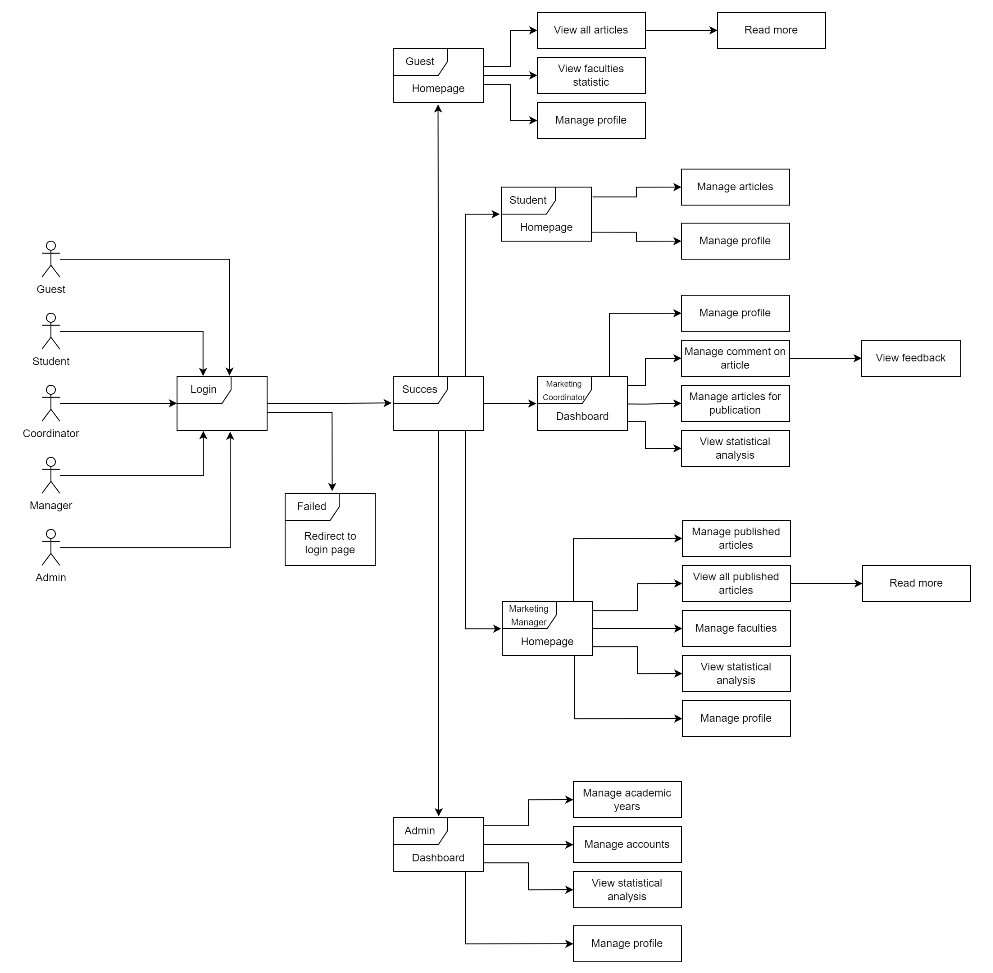


Figure 2: Sitemap

* + 1. **Mobile Design**

Below are some wireframes of our website designs for mobile devices.

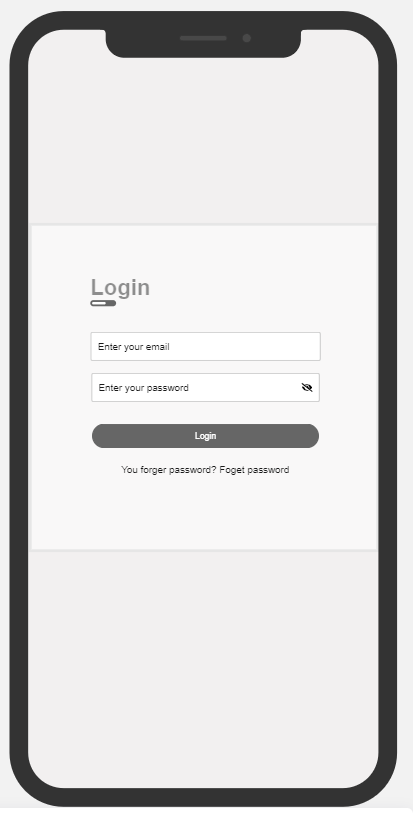


Figure 3: Login page

This is the login page interface designed for mobile devices, where users log in to continue accessing the website (figure 3).

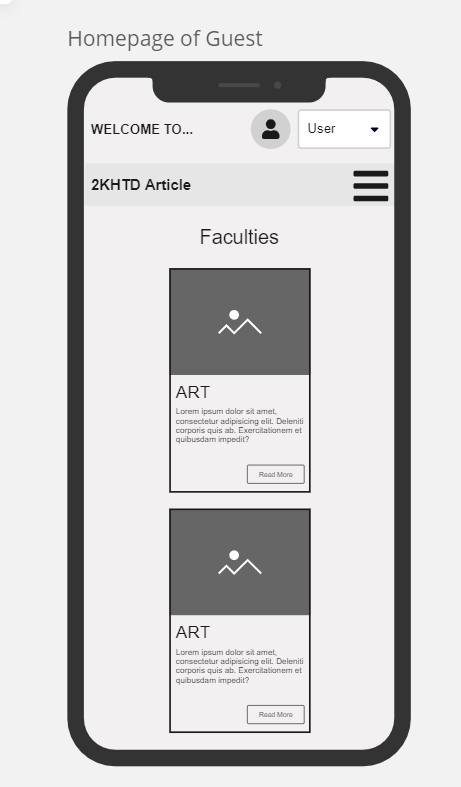


Figure 4: Homepage of Guest

This is the home page interface of the "Guest" role designed for mobile devices, where the user will be redirected to after successfully logging in (figure 4).

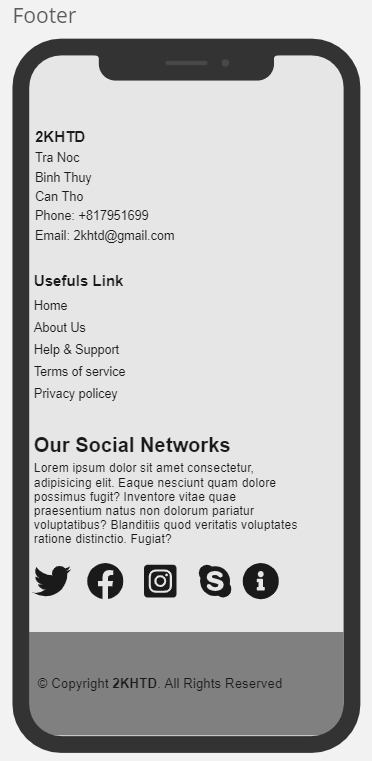


Figure 5: Footer

This is a footer interface designed for mobile devices (figure 5).

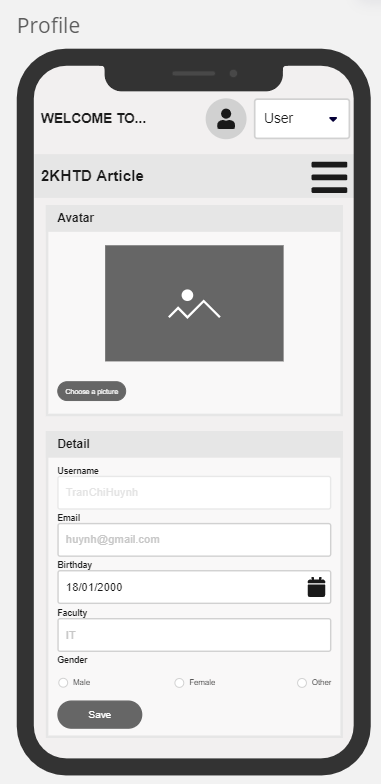


Figure 6: Profile page

This is the profile page interface designed for mobile devices, where users can manage their information (figure 6).

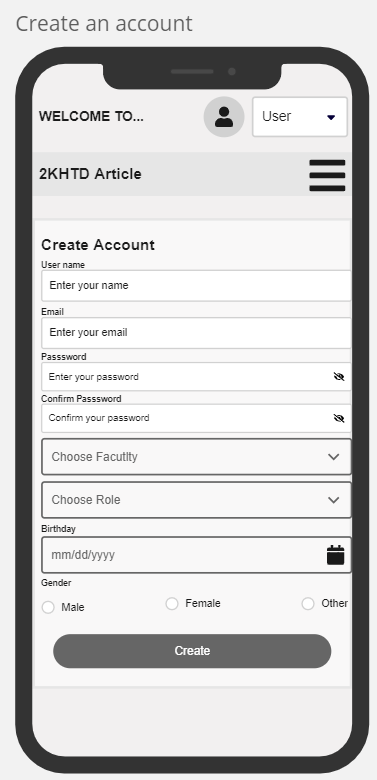


Figure 7: Account Creating page of Admin

This is the account creating page interface of the "Admin" role designed for mobile devices, where admins can create accounts for other users (figure 7).

* + 1. **Desktop Design**

Below are a few wireframes of the desktop interface

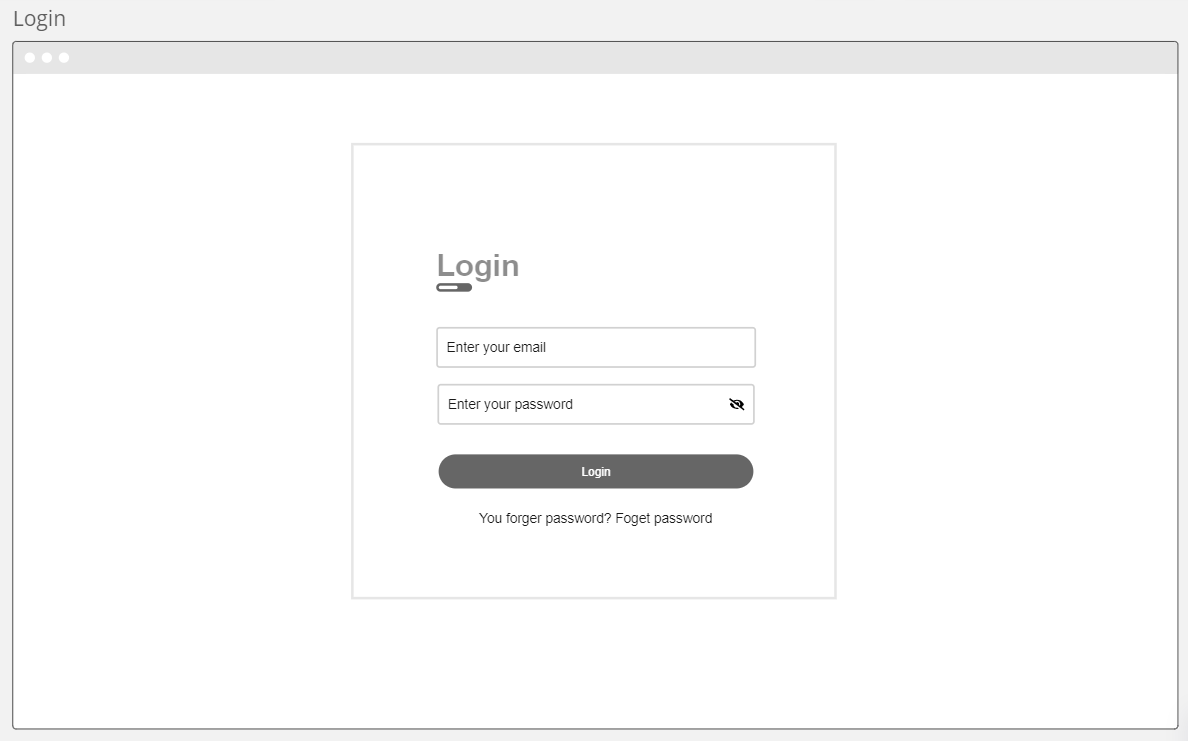


Figure 8: Login page for desktop

This is the login page interface designed for desktop devices, where users can log in (figure 8).

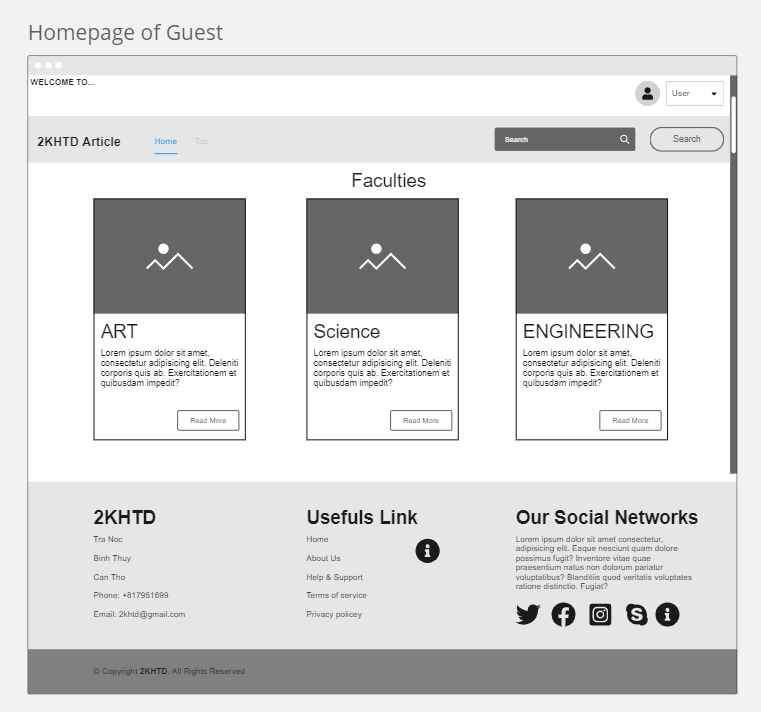


Figure 9: Homepage of Guest for desktop

This is the homepage interface of the "Guest" role designed for desktop devices, where users can view published articles of all faculty (figure 9).

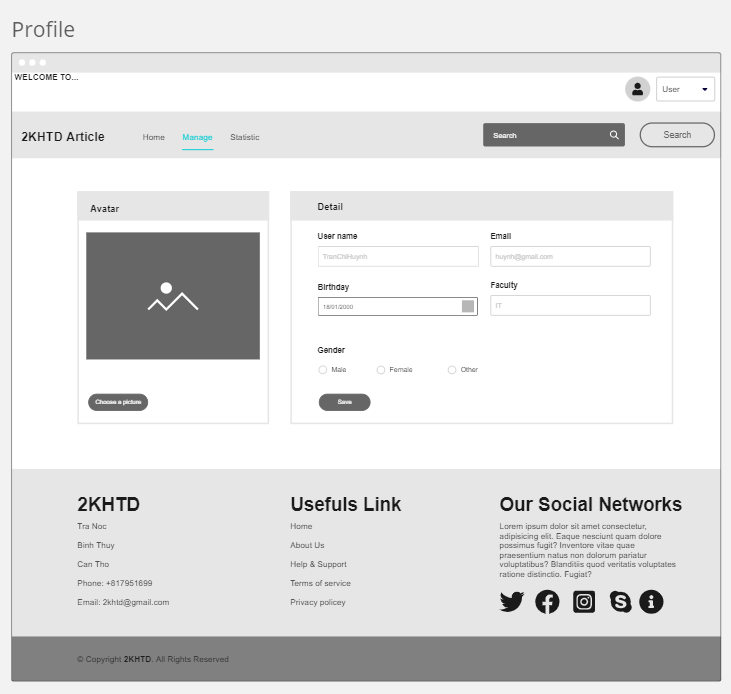


Figure 10: Profile page for desktop

This is a profile page interface designed for desktop devices, where users can view their registration information (figure 10).

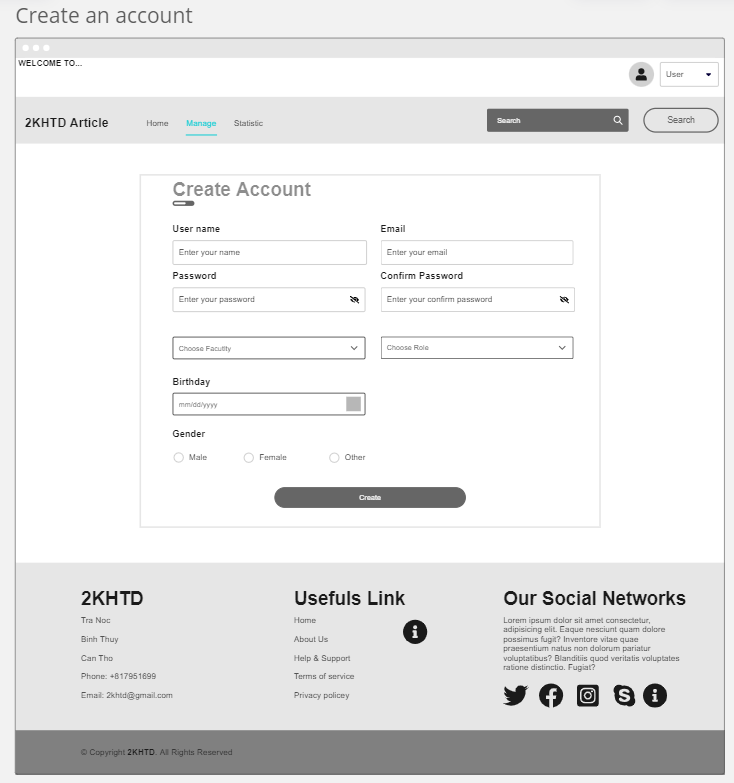


Figure 11: Account Creating of Admin for desktop

This is the account creating page interface of the "Admin" role designed for desktop devices, where admins can create accounts for other users (figure 11).

# Bibliography

Bootstrap, 2024. *getbootstrap.* [Online]   
Available at: https://getbootstrap.com/docs/5.3/getting-started/introduction/

CRAVIT, R., 2022. *venngage.* [Online]   
Available at: https://venngage.com/blog/color-blind-friendly-palette/

Elliot, A. J. & Maier, M. A., 2013. *Pubmed.* [Online].

Nielsen, J., 1994. *10 Usability Heuristics for User Interface Design.* [Online]   
Available at: https://www.nngroup.com/articles/ten-usability-heuristics/