

**NGO WEBSITE**

|  |  |
| --- | --- |
| **Authors** | **Group 1** |
| **Dang Dinh Thuan**  **Tran Ha Anh**  **Nguyen Giang Thanh**  **Le Thanh Dat**  **Pham Doan Toan** |
| **Date** | **28/03/2024** |
| **Instructor** | **PhD Kieu Van Khai** |

<Ha Noi, 2024>

**Index**

[**1.**](#_30j0zll) **Problem Definition 3**

[1.1.](#_1fob9te) Problem Abstraction 3

[1.2.](#_3znysh7) The exist System 4

[1.3.](#_2et92p0) The Proposed System 5

[**2.**](#_1t3h5sf) **System Requirements Specification 5**

[2.1.](#_4d34og8) Approach Method 5

[2.2.](#_2s8eyo1) Requirement Analysis Method 9

[2.3.](#_2s8eyo1) System Requirement 13

[**3.**](#_17dp8vu) **System Designs 14**

[3.1.](#_3rdcrjn) Process Diagram 14

[3.2.](#_3rdcrjn) Entity Relationship Diagram 17

[3.3.](#_26in1rg) Database Design 17

[3.4.](#_lnxbz9) Sitemap 23

[3.5.](#_35nkun2) System fucntions design 23

[3.5.1.](#_tyjcwt) Boundaries of the System 90

[3.5.2.](#_3dy6vkm) Hardware and Software Requirements 91

[**4.**](#_1ksv4uv) **Task sheet 91**

[**5.**](#_44sinio) **Validation Checklists 94**

# Problem Definition

## Problem Abstraction

* Project Information :
* Project name : NGO Website
* Project team : Group 1
* Instructor : PhD Kieu Van Khai
* Project Topic: Non-governmetal Organization Website

Non-governmental fund mobilization projects are efforts to mobilize financial resources from different sources outside the government budget. This is an important means for non-profit organizations or NGOs to carry out their activities and projects.

Non-governmental fundraising projects require a careful strategy and initiative in finding and maintaining financial resources from a variety of sources. This process is also often accompanied by relationship building with partners and sponsors to ensure ongoing support for the project.

NGOs often engage in humanitarian, relief, and community development activities. They play an important role in reducing hunger and poverty, providing education, health care and other basic services to communities.

Non-governmental organizations often work to protect the environment, maintain biodiversity, and promote sustainable development measures;

promote civic education, strengthen debate and community participation, and promote education, etc.

Information technology (IT) and communications plays an important role in the operations of Non-Governmental Organizations (NGOs), providing effective tools for management, community interaction, and data collection. materials, and enhance social impact.

By applying these technologies, NGOs can enhance management capabilities, increase transparency, and create positive interactions with communities and supporters.

## The Exist System

- Similar projects as:

+ crowdfrica (<https://www.crowdfrica.org/>) :

* The website focuses on educational and medical support for organizations or individuals in Africa.
* Advantages: Easy to use, all secured finances can be retained.
* Disadvantages: Campaign creation is limited in some countries, PayPal is not supported.

+ youcareds (<https://www.youcareds.com/>) :

* Volunteer website for people with disabilities includes equipment support, physical therapy instructions, health advice, etc.
* Advantages: YouCared does not charge commission on donations, Many social activities supported.
* Disadvantages: the cost of creating a website is very expensive and difficult to customize for customer requirements.

+ giveforward (<https://www.giveforward.com/>) :

* - Community fundraising website for all fields and situations that need help.
* Advantages: Strong community support for many social causes, No goals or deadlines required (mandatory)
* Disadvantages: the cost of creating a website is very expensive and difficult to customize for customer requirements - limited to the United States only.

## The Proposed System

* Because of the above advantages and disadvantages, we developed a new NGO website to overcome the remaining limitations. A flexible, cost-effective NGO Website that is easily accessible to users in matters of communication, organization and payment, accessing user opinions and making changes to suit the current situation and customization requirements of customer.

# System Requirements Specification

## Approach Method

* Why apply a software construction approach?

Nowadays, the need to use software to optimize management and operations activities in businesses is becoming more and more popular and is also a mandatory requirement to not fall behind and gain the opportunity to get ahead in this era. 4.0 has a dizzying pace of development today.

Software such as production software, accounting software, human resource management software, etc. have become effective support tools for businesses to improve labor productivity, minimize costs and increase speed. growth as well as enhancing competitiveness in the market.

A few popular approaches:

* Agile: is a technique used to design a disciplined software management process that typically allows for frequent, flexible development projects. The way this method works is by establishing a systematic structure for conducting various projects in software engineering.
* Advantages: highly adaptable, limited risks, instant, clear communication...
* Disadvantages: Not ideal for beginners, ineffective in terms of documentation, requires in-depth understanding of the project process and application,...
* DevOps: is a series of activities that maintain the development style of a development team. DevOps implementations focus on operational reforms that enhance coordination between departments responsible for different parts of the software development lifecycle, such as Development, Quality Assurance, and Quality Assurance. and Operations.
* Advantages: rapid and continuous delivery of working software, easy deployment of new processes, systems and applications, improved customer experience, continuous adaptive updates,...
* Disadvantages: must accept changes in working style, requires a lot of interaction within the team and project stakeholders,...
* Waterfall: is a rigid linear model consisting of sequential phases based on multiple objectives (requirements, design, implementation, verification, maintenance). Each step must be 100 percent completed before the next step can begin. It means, no reversible modifications can be made during development for any requirement.
* Advantages: Easy to understand and simple, good for small businesses, simple enough to manage project processes, saves a lot of time,...
* Disadvantages: Cannot be changed or modified, cannot handle major risks, not suitable for large software projects,...
* RAD: is an effective technique that promises much faster development and better product quality results than other software development methods. It is designed in such a way that a software developer can make use of it conveniently. The main goal of this method is to speed up the entire software development process.
* Advantages: Easy to understand, reduces risk, less effort, minimizes errors during development, easily collects feedback from customers for improvement, provides high functionality, etc.
* Disadvantages: Not ideal for low budgets, requires high skills, not suitable for large projects, difficult to track progress and issues, etc.
* Scrum: As a phase of software development, it begins with a brief plan for each sprint, followed by regular scrum meetings to verify the results after each sprint of the project, and ends with a breakdown. final volume. Scrum techniques are suitable for carrying out tasks with customer input and technical specifications that are not clearly described. Using this method to create a series of product prototypes in one go, software development companies can speed up application development.
* Advantages: Can release frequently, flexible approach, process of collecting feedback and customer reviews is easy and fast, bugs can be easily identified and changes can be responded to.
* Disadvantages: Needs constant communication, poor coverage with a team of too many developers, frequent team changes make it difficult to determine productivity.
* Spiral Model: Spiral Model is a combination of Agile and Waterfall methods, mainly focusing on detecting and mitigating risks earlier. Developers start small, then explore project risks, devise a risk management strategy, and finally determine whether to take the next phase of the project for iteration. Next twist or not. The performance of any project applying this model depends on the effective management, diligence and efforts of the project team. This spiral model approach helps developers make improvements in the code or design even during the testing phase, compared to the waterfall model. It is suitable for most types of projects, but requires effective management.
* Advantages: Less risks, easy operations for large projects, easy cost estimation, easy access to additional functions
* Disadvantages: The model is expensive, without proper risk analysis, the project may fail.
* Which method should be used? Why?

With limited manpower, resources, costs and uneven qualifications among members, plus a completion time of 1 month. Group 1 uses the Agile method because of its flexibility and ability to change frequently, while also being flexible in updating information for users.

As a project about a non-governmental organization and charity, the project requires flexibility in changing to suit customers, and requires continuous updating of relevant information.

## Requirement Analysis Method

To start working on a project in a certain field, the indispensable thing is to grasp the requirements of our customers and stakeholders, to know what they need and want. is on their website. And this is not simple, it requires us to have the necessary skills to analyze and understand those requirements in the most accurate and complete way.

And most of the time, our customers sometimes do not cover all the options needed when developing a particular product, their vision tends to be limited and not rich when making suggestions. initiative. It is difficult for users to imagine possibilities that are significantly different from what they have now.

Furthermore, no single requirement can help us come up with a complete set of requirements in all respects, it will still have shortcomings, gaps and cannot pass a thorough validation process.

To overcome those weaknesses, we need to apply a multi-faceted approach to income requirements. It is good advice that when you practice requirements engineering, you should use a variety of methods to suit different stages of the process.

There are many methods of requirements analysis engineering, but we have found 5 methods that we consider to be the most effective and popular. A healthy combination of these techniques will be tailored to different projects, helping to improve requirements related issues during software development to be handled better.

* **Interview method:** Interviews are a great way to elicit requirements from customers. The results of interviews are useful in gathering basic information about business needs, customer and user problems, and other stakeholders.

Interviews should be diverse and representative of stakeholders. We need complete customer and user profiles. This is necessary to get a proper view of competing needs, so that your system requirements are not biased towards one group.

During the interview process, it is important to ask open-ended questions. The questions that respondents must answer in detail and specifically, from there we get valuable information for the software development process. We ask interviewees to explain their thinking and give reasons, thereby providing context for evaluating and validating requirements.

Good follow-up questions can go into more detail or pull up to get an overview of the context. Some people will tend to be specific and exceptional. With them, we will need to try. Others will talk about context without going into specifics. For those, we will need to dig deeper.

* **Survey method using questions and surveys:** Personal interviews present some challenges. They can be difficult to arrange and time-consuming for the interviewer. Additionally, the requirements we collect may only be superficial, not every interviewer has the skills to ask follow-up questions in real time. Questionnaires may provide an effective alternative. They enable tracking with multiple stakeholders at once. The questionnaire needs to be carefully considered, asking probing questions is a good way to achieve basic requirements that the customer may not be aware of but are essential for a successful project.
* **Observation method:** One of the best ways to understand what users really need is to observe them performing their daily tasks. User observation can be passive or active. Observing and actively asking questions of users is the best approach to understanding that process. When observing users, record the actions and activities that take place. And ask questions like: What worked well? What makes it difficult for users? Note the obstacles that users must regularly overcome. By observing, you will better understand what they are experiencing and what can be improved. We will then be better able to specify a system that can successfully improve our users' processes and bring them much greater productivity and usability than simply providing give them an incremental improvement.
* **Document analysis method:** Viewing documents helps you perform process analysis and avoid gaps that need to be filled. Thanks to document analysis, it helps us collect data information that other techniques have not resolved. Okay. Little information about why the current system works the way it does is often buried in technical specifications and design documents. Insights gained from document analysis can help you further formulate questions and evaluate the completeness of your set of requirements.
* **Interface analysis method:** Analyzing the interface of the system, both human and machine, is extremely important to ensure requirements are fulfilled and the system will be usable. Interfaces for a software product will include interfaces that include: End users, system components with which the software interacts, external systems with which the software interacts. Analyzing the interface thoroughly, understanding the system's interaction context will uncover requirements that are not easily visible to the user.

Analyzing and specifying requirements is a step in the software project development process. The job in this step is to figure out what we have to develop. Requirements analysis involves regular communication with stakeholders and end users of the product to identify expectations, resolve conflicts, and document all key requirements.

- For the first phase of the project, we use document analysis method with customer requirements to have an overview of the project such as field, scale, and object required by the customer. We then use an interview approach with the customer and all key stakeholders to reach a common understanding of what the product should do, which will ensure that the final product is compliant requirements. After conducting interviews and knowing the requirements from customers and end users, we will use the technique of observing users, analyzing and asking questions to users to clearly understand the process. process and will help us visualize how the product will work. Next, using the interface analysis method, the interface is an extremely important step to ensure the requirements are completed and the system is usable. We can provide customers or users with a prototype to use. Having a look and feel for the final product, engineering helps us have a clearer view of the product and can solve problems ahead of time. Those are the requirements analysis methods we used, each method is suitable for different stages of the project process.

## System Requirement

1. General Requirement

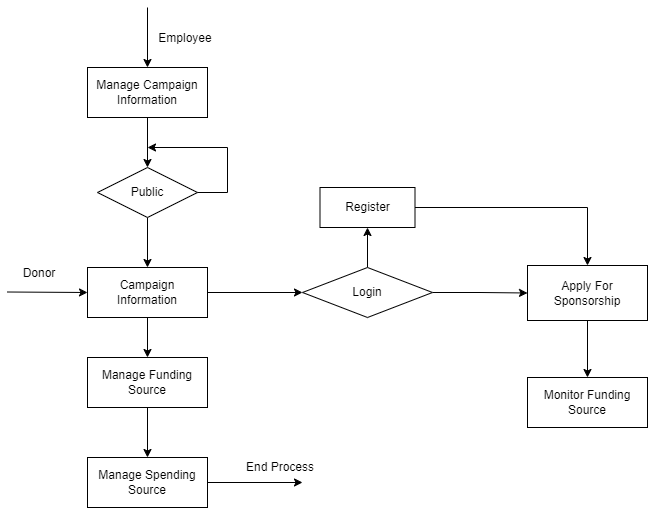
* Users can view and select the campaign they want to sponsor. When sponsoring, users must log in to the system.
* Managers can log in to the system to manage campaign information

1. Detail Requirement

* Users:
* Users can register an account to log into the system
* User can change account information
* Users can view information about campaigns, funds, and reviews of implemented campaigns.
* Users can choose a campaign to sponsor, payment method and track the progress of that campaign
* Managers:
* View a list of user accounts
* Manage campaign details
* Add, edit, delete campaign information
* Manage blog details
* Add, edit, delete campaign information
* Manage members
* Add, edit, delete member information
* Manage information about sponsor partners
* Add, edit, delete partner information
* View funding for each fund and each campaign

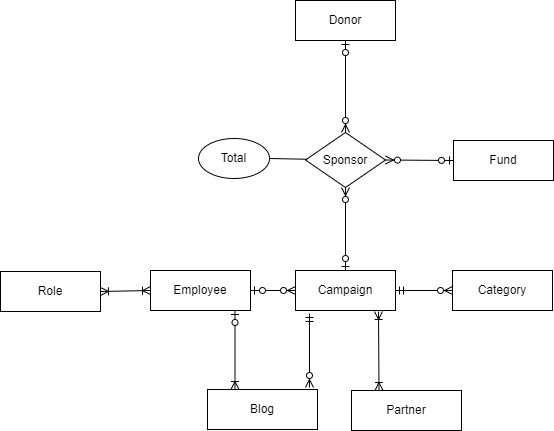
# System Designs

## Process Diagram

******

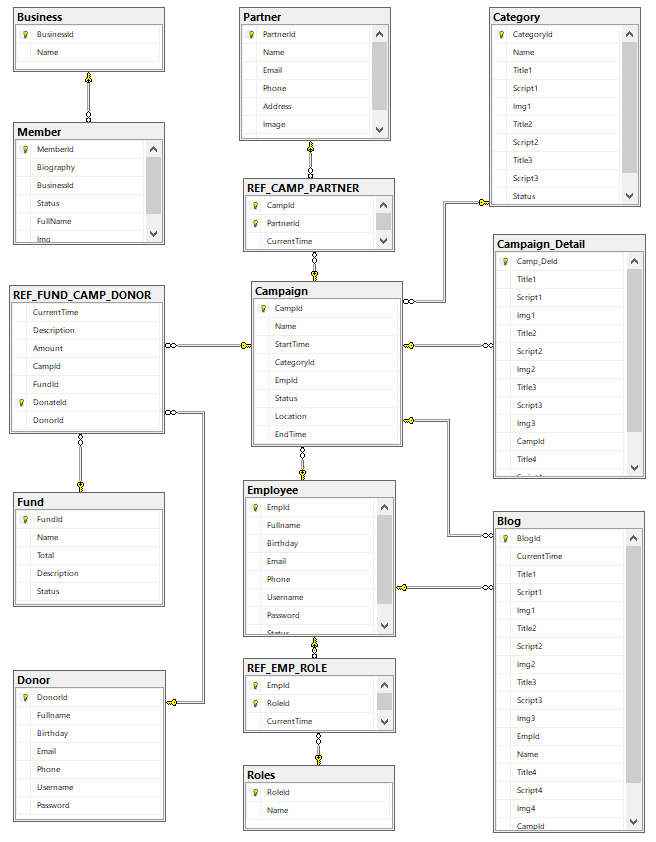
* **Manage campaign information:** Units involved in the fundraising campaign add detailed information about the project and put it on public standby.
* **View campaign information:**
* Manager: Public campaign information
* Donor: Visit the website and learn about current fundraising campaigns
* **Manage funding sources:** Units involved in financial work compile statistics and classify funding sources for each campaign.
* **Manage spending sources:** Units involved in financial work prepare budget estimates, invoice statements, and report on each source of expenditure for each campaign.
* **Login & Register:** Donors who want to register for sponsorship need to log in to the website.
* Login: Check user information. If information matches, login is allowed; If the information is incorrect, a message will appear
* Register: Require users to enter information to register for an account
* **Apply for sponsorship:** Donors choose the campaign they want to sponsor, sponsorship amount, message and payment method. The system requires users to enter contact information to send thanks and invoice statements.
* **Monitor funding sources:** Donors track the progress of a campaign and donations from other donors to that campaign.

## Entity Relationship Diagram



## Database Design

* Table Relationship



* Table Detail
* Donor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| DonorId | int | NOT NULL | Yes | No |
| Fullname | varchar(50) | NULL | No | No |
| Birthday | Date | NULL | No | No |
| Email | varchar(50) | NULL | No | No |
| Phone | varchar(10) | NULL | No | No |
| Username | varchar(18) | NULL | No | No |
| Password | varchar(50) | NULL | No | No |

* Fund

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| FundId | int | NOT NULL | Yes | No |
| Name | varchar(100) | NULL | No | No |
| Total | float | NULL | No | No |
| Description | text | NULL | No | No |
| Status | int | NULL | No | No |

* Business

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| BusinessId | int | NOT NULL | Yes | No |
| Name | varchar(50) | NULL | No | No |

* Member

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| MemberId | int | NOT NULL | Yes | No |
| Biography | text | NULL | No | No |
| Fullname | varchar(50) | NULL | No | No |
| Img | varbinary(max) | NULL | No | No |
| Status | int | NULL | No | No |
| BusinessId | int | NULL | No | Yes |

* Category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| CategoryId | int | NOT NULL | Yes | No |
| Name | varchar(100) | NULL | No | No |
| Title1 | varchar(100) | NULL | No | No |
| Script1 | text | NULL | No | No |
| Img1 | varbinary(max) | NULL | No | No |
| Title2 | varchar(100) | NULL | No | No |
| Script2 | text | NULL | No | No |
| Title3 | varchar(100) | NULL | No | No |
| Script3 | text | NULL | No | No |
| Status | int | NULL | No | No |

* Partner

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| PartnerId | int | NOT NULL | Yes | No |
| Name | varchar(50) | NULL | No | No |
| Email | varchar(30) | NULL | No | No |
| Phone | varchar(10) | NULL | No | No |
| Address | varchar(200) | NULL | No | No |
| Image | varbinary(max) | NULL | No | No |
| Status | int | NULL | No | No |

* Employee

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| EmpId | int | NOT NULL | Yes | No |
| Fullname | varchar(50) | NULL | No | No |
| Birthday | date | NULL | No | No |
| Email | varchar(50) | NULL | No | No |
| Phone | varchar(10) | NULL | No | No |
| Username | varchar(18) | NULL | No | No |
| Password | varchar(50) | NULL | No | No |
| Status | int | NULL | No | No |

* Role

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| RoleId | int | NOT NULL | Yes | No |
| Name | varchar(50) | NULL | No | No |

* Campaign

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| CampId | int | NOT NULL | Yes | No |
| Name | varchar(100) | NULL | No | No |
| StartTime | datetime | NULL | No | No |
| EndTime | datetime | NULL | No | No |
| Location | varchar(100) | NULL | No | No |
| EmpId | int | NULL | No | Yes |
| CategoryId | int | NULL | No | Yes |
| Status | int | NULL | No | No |

* Campaign\_Detail

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| Camp\_DeId | int | NOT NULL | Yes | No |
| Title1 | varchar(100) | NULL | No | No |
| Script1 | text | NULL | No | No |
| Img1 | varbinary(max) | NULL | No | No |
| Title2 | varchar(100) | NULL | No | No |
| Script2 | text | NULL | No | No |
| Img2 | varbinary(max) | NULL | No | No |
| Title3 | varchar(100) | NULL | No | No |
| Script3 | text | NULL | No | No |
| Img3 | varbinary(max) | NULL | No | No |
| Title4 | varchar(100) | NULL | No | No |
| Script4 | text | NULL | No | No |
| Img4 | varbinary(max) | NULL | No | No |
| Status | int | NULL | No | No |
| CampId | int | NULL | No | Yes |

* Blog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| BlogId | int | NOT NULL | Yes | No |
| Name | varchar(100) | NULL | No | No |
| Title1 | varchar(100) | NULL | No | No |
| Script1 | text | NULL | No | No |
| Img1 | varbinary(max) | NULL | No | No |
| Title2 | varchar(100) | NULL | No | No |
| Script2 | text | NULL | No | No |
| Img2 | varbinary(max) | NULL | No | No |
| Title3 | varchar(100) | NULL | No | No |
| Script3 | text | NULL | No | No |
| Img3 | varbinary(max) | NULL | No | No |
| Title4 | varchar(100) | NULL | No | No |
| Script4 | text | NULL | No | No |
| Img4 | varbinary(max) | NULL | No | No |
| Status | int | NULL | No | No |
| CampId | int | NULL | No | Yes |
| EmpId | int | NULL | No | Yes |

* REF\_CAMP\_PARTNER

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| CampId | int | NOT NULL | Yes | Yes |
| PartnerId | int | NOT NULL | Yes | Yes |
| CurrentTime | datetime | NULL | No | No |

* REF\_EMP\_ROLE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| EmpId | int | NOT NULL | Yes | Yes |
| RoleId | int | NOT NULL | Yes | Yes |
| CurrentTime | datetime | NULL | No | No |

* REF\_FUND\_CAMP\_DONOR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Datatype** | **Null Option** | **Is PK** | **Is FK** |
| DonateId | int | NOT NULL | Yes | No |
| CampId | int | NULL | No | Yes |
| DonorId | int | NULL | No | Yes |
| CampId | int | NULL | No | No |
| Amount | float | NULL | No | No |
| Description | text | NULL | No | No |
| CurrentTime | datetime | NULL | No | No |

## Sitemap

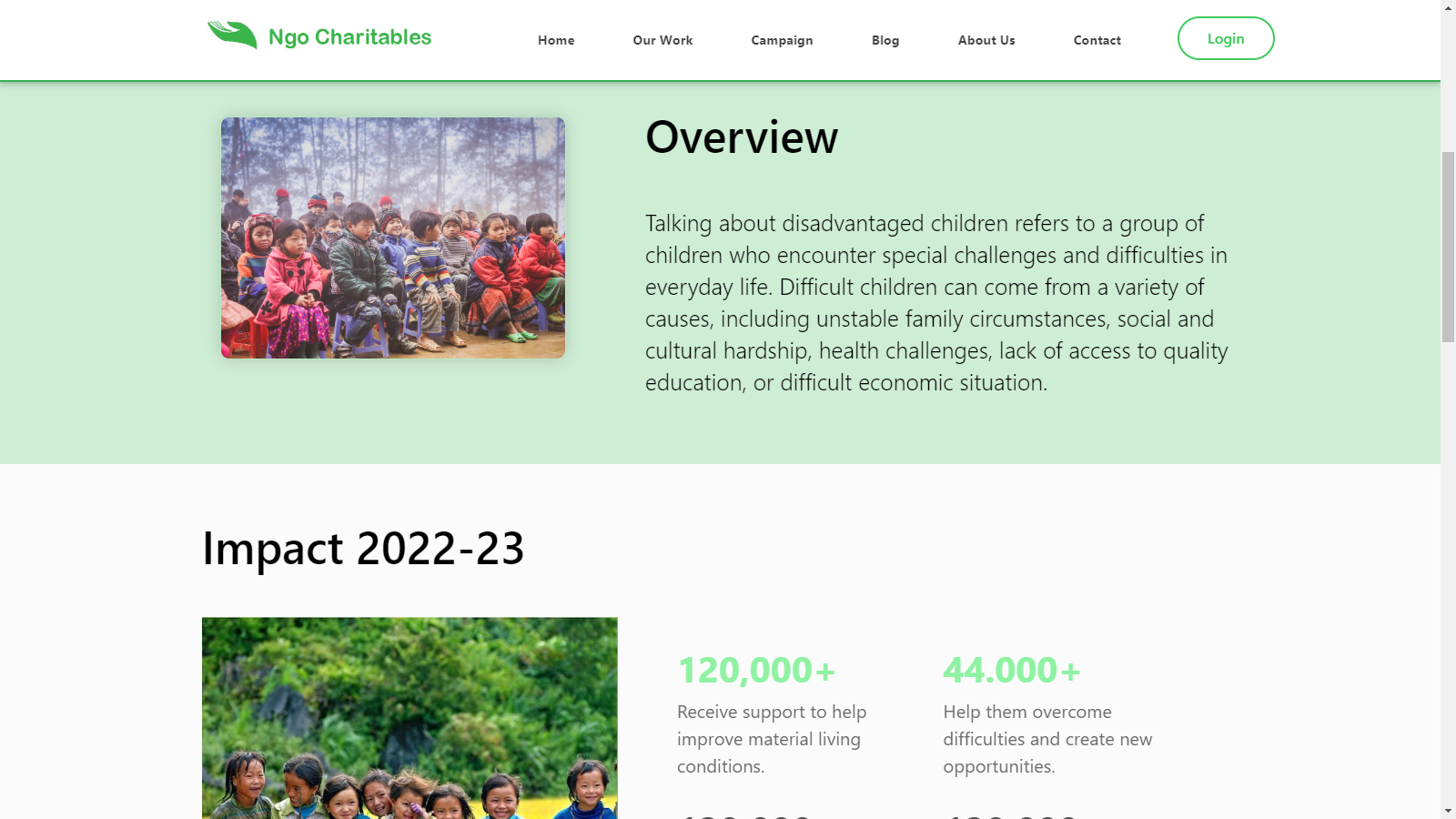
## 

## System fucntions design

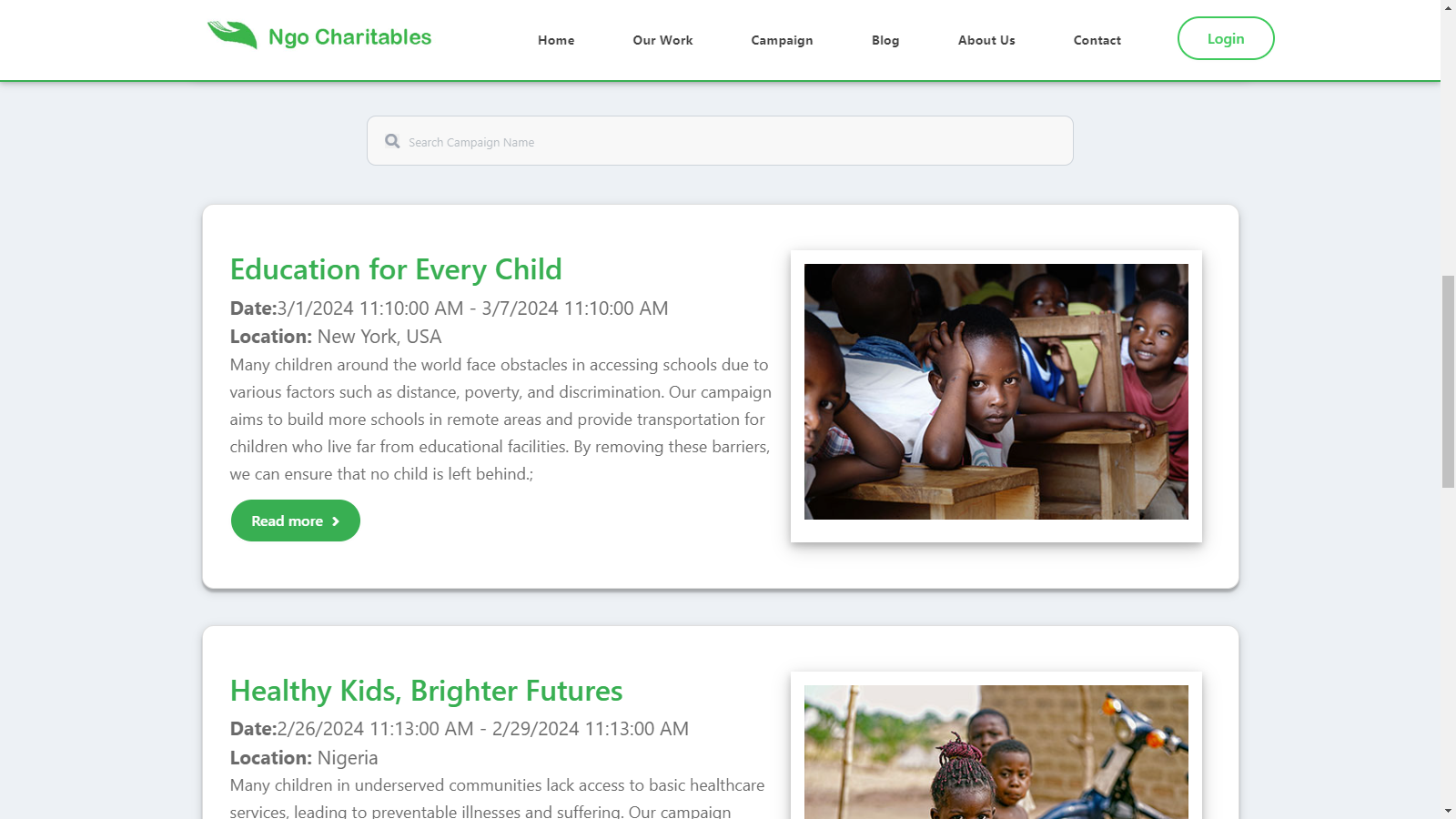
* User Interface
* Home



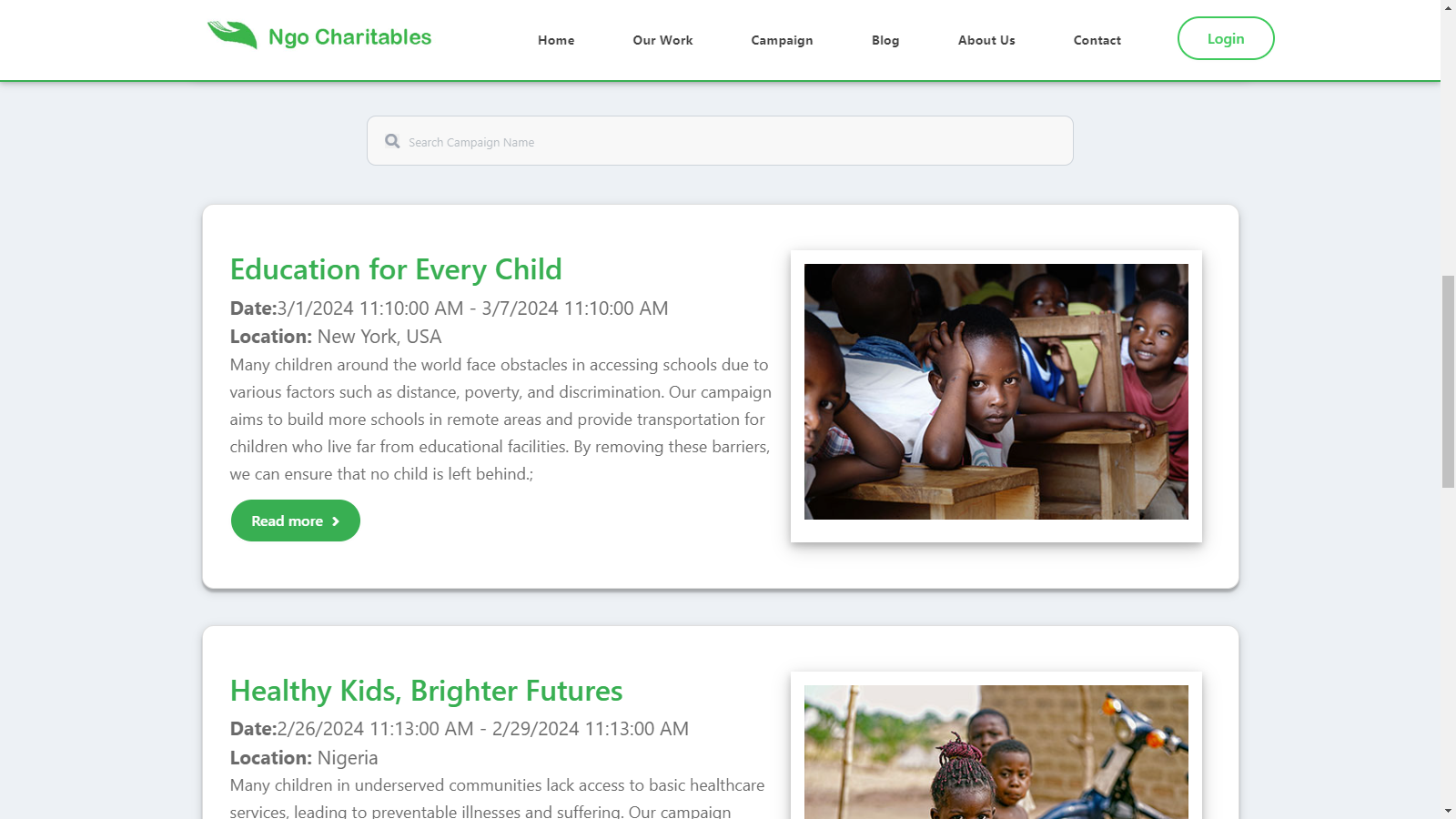
* Our Work



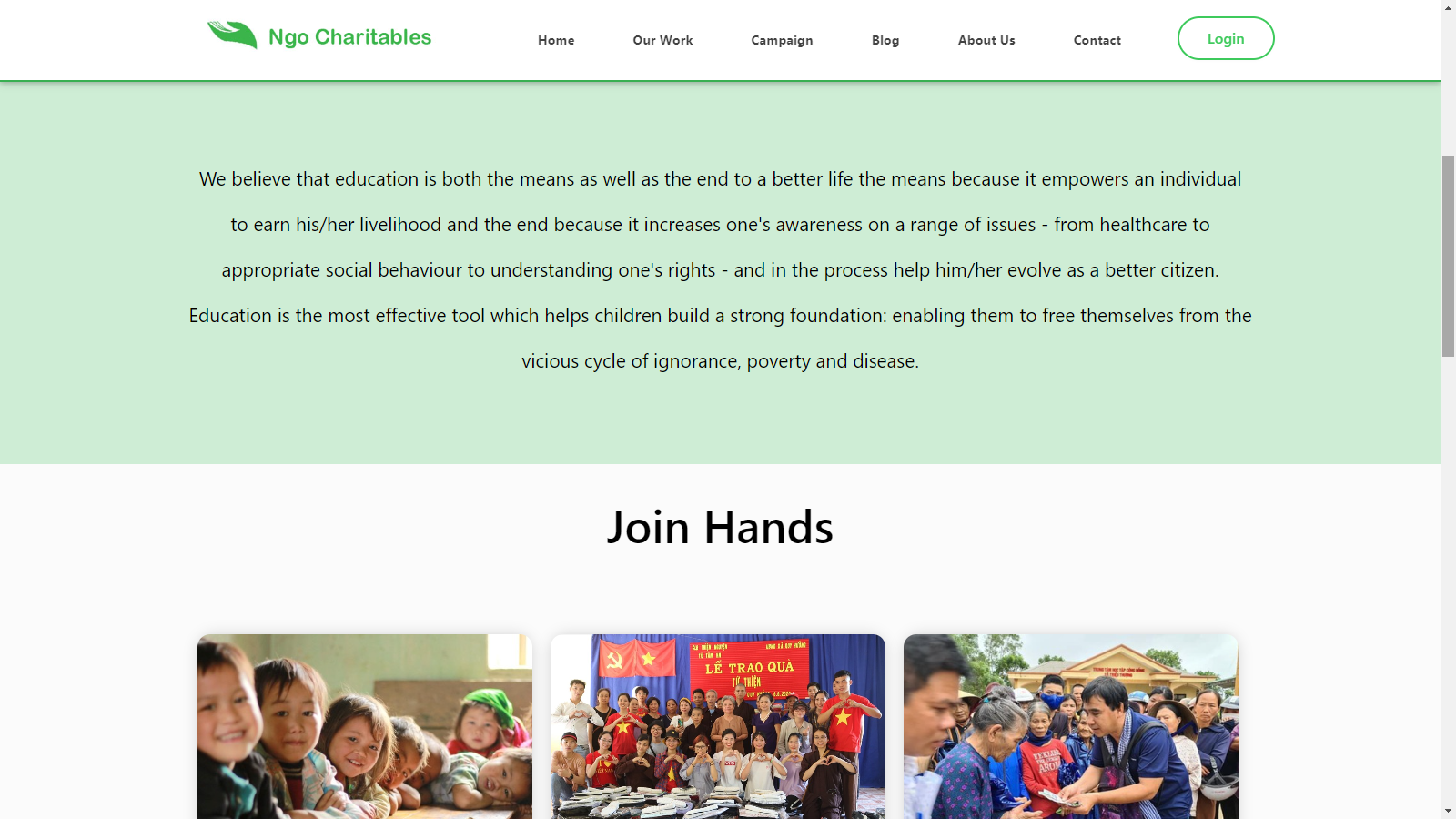
* Campaign



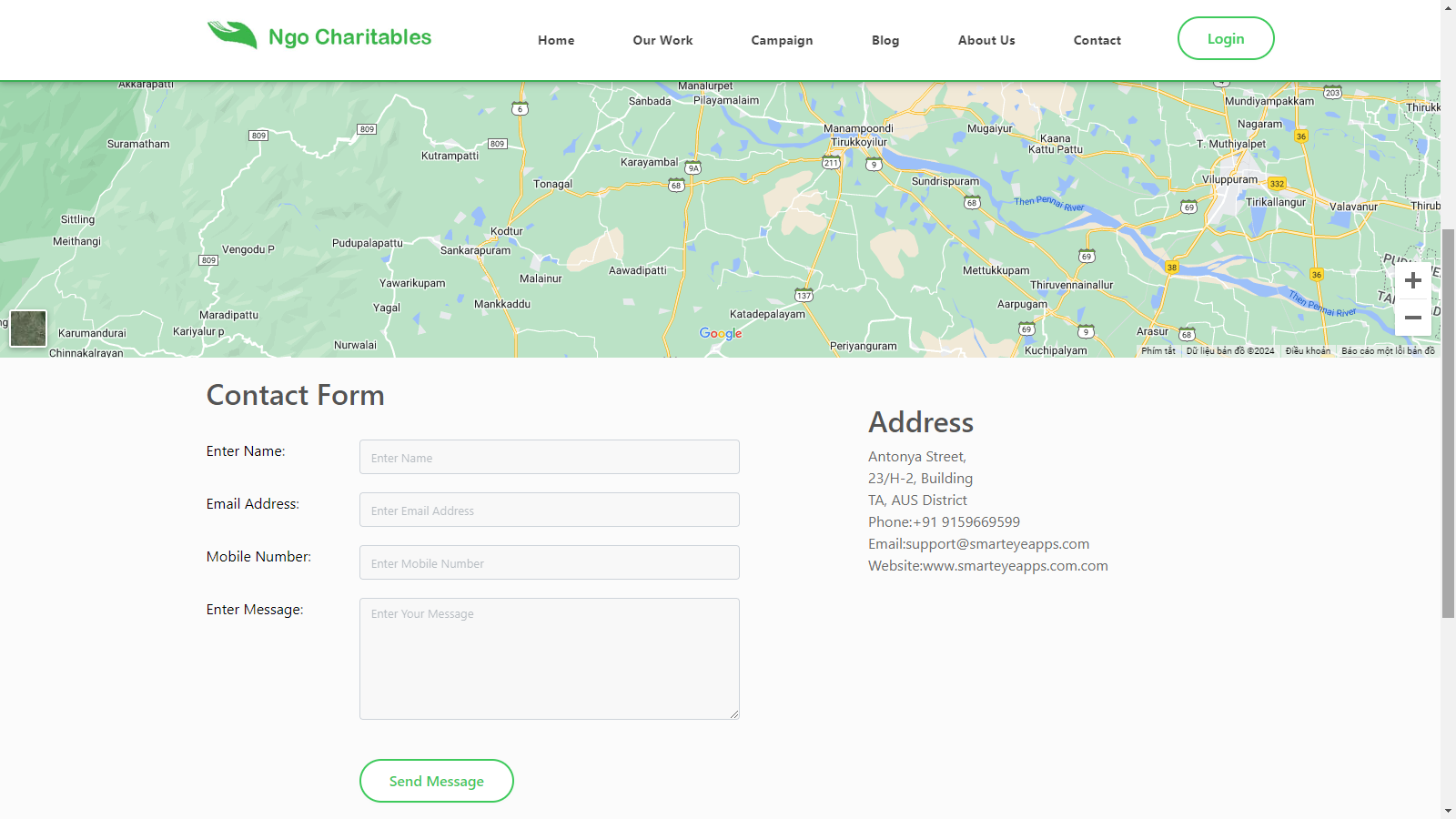
* Blog



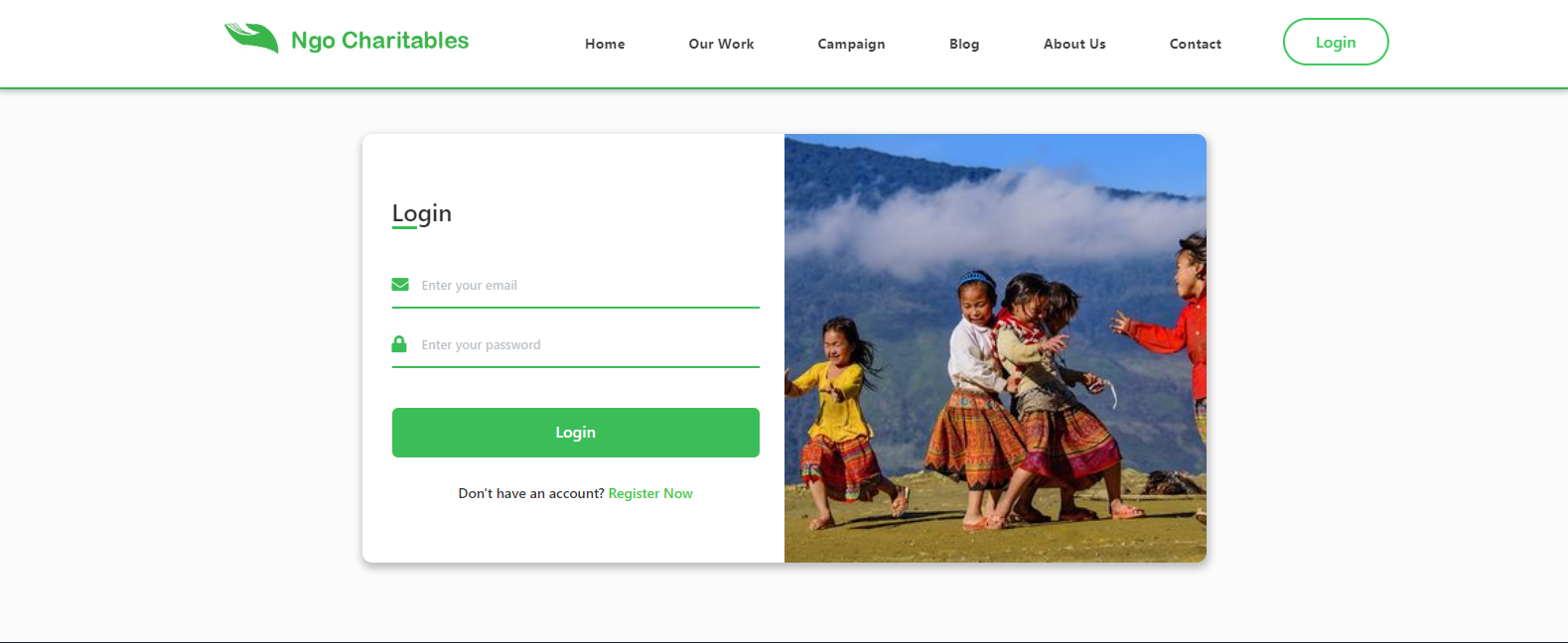
* About Us

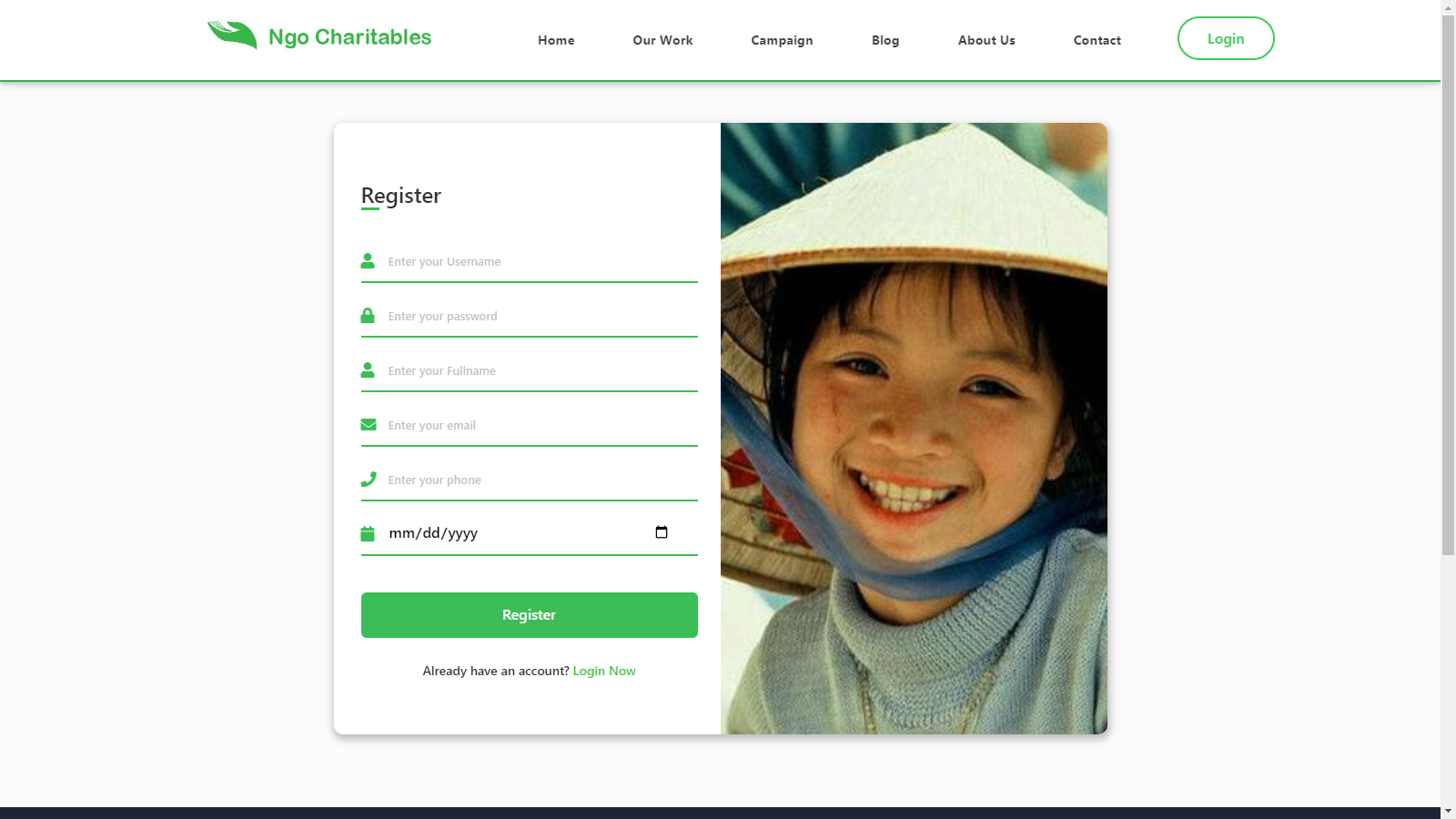


* Contact Us

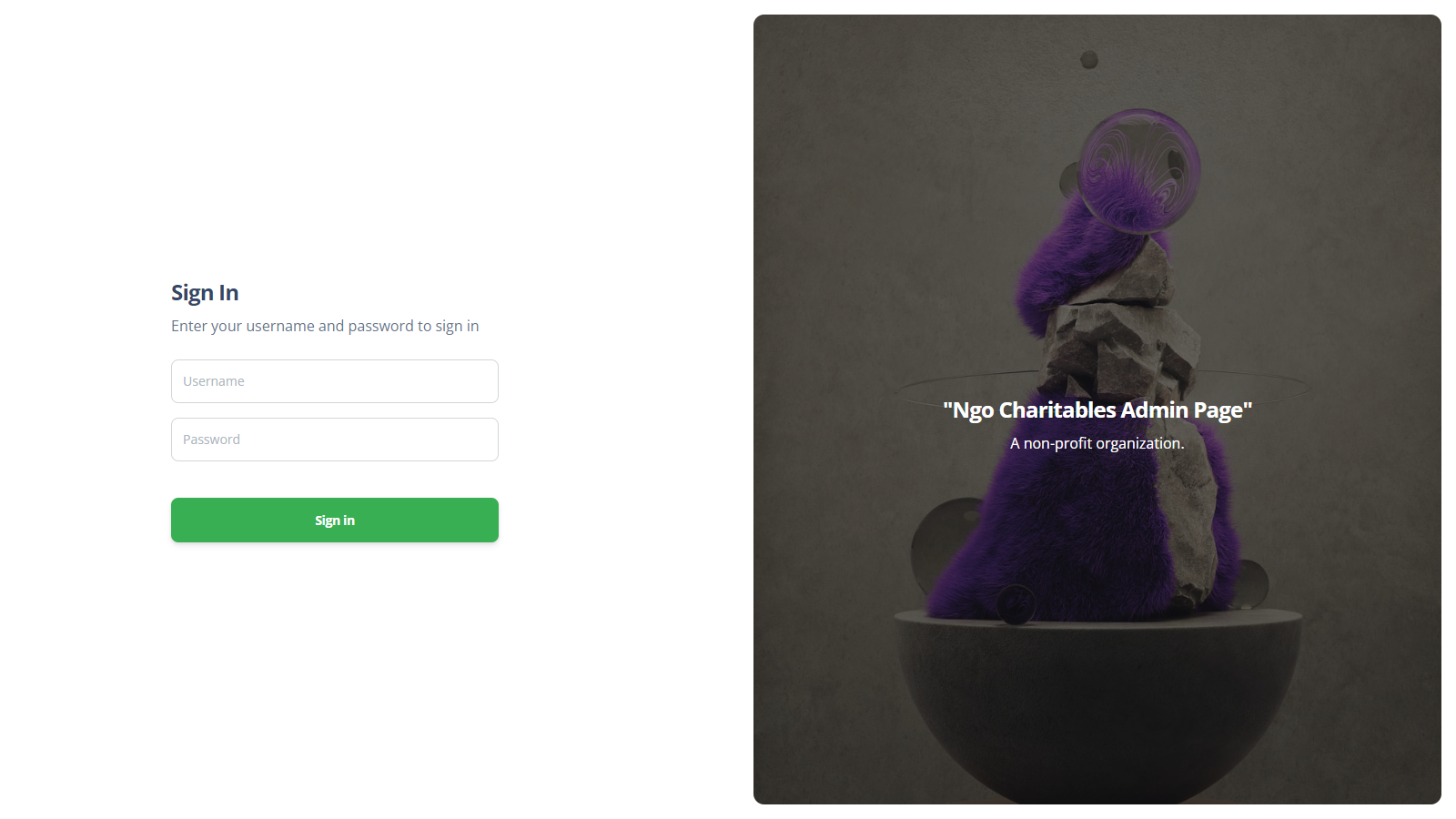


* Login & Register

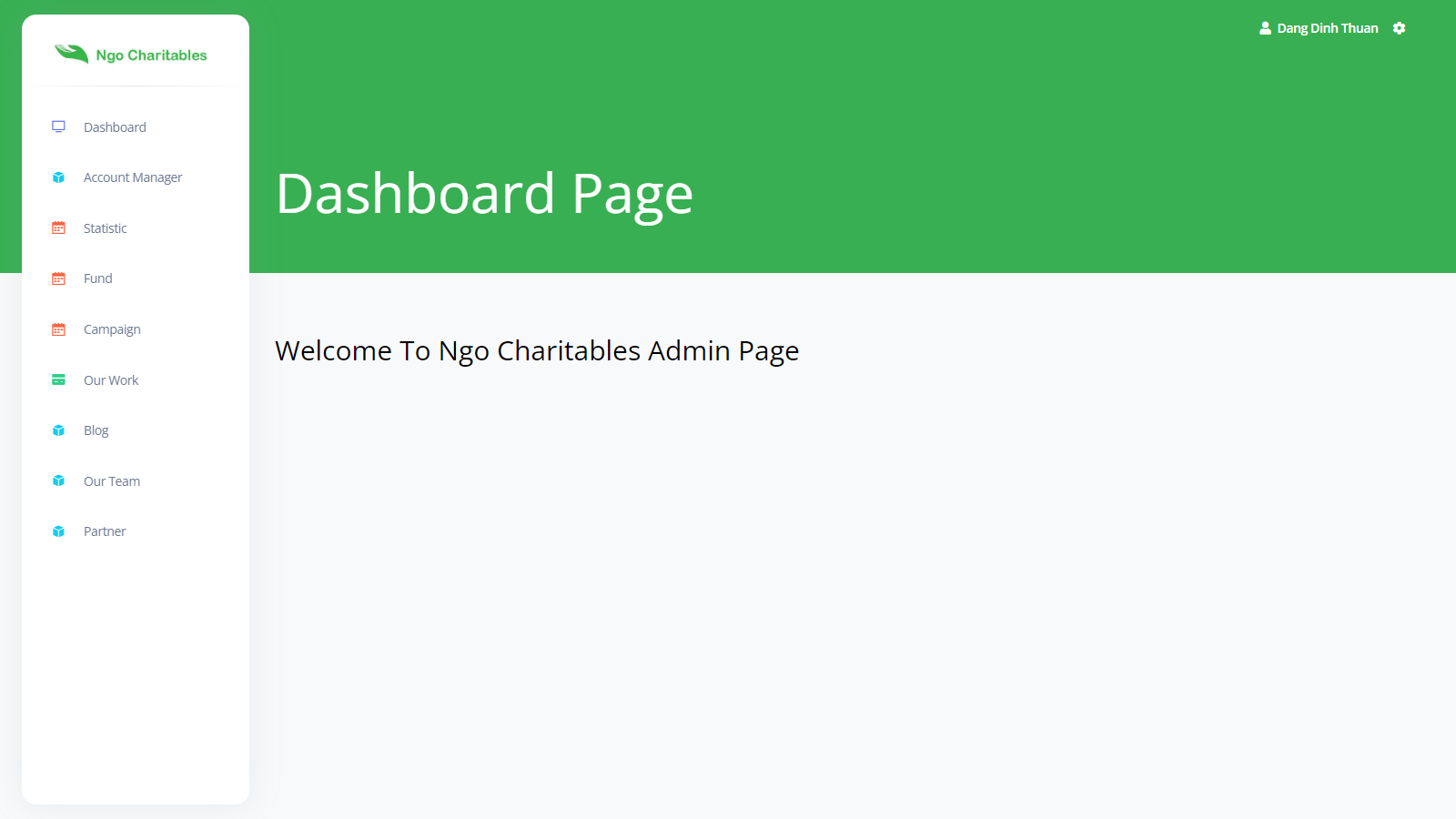




* Admin Interface
* Login



* Home



* Algorithm
* Rule for drawing diagram handing of business procedures

|  |  |
| --- | --- |
|  | Step to take |
|  | Check/Approve/Decide |
|  | Begin |
|  | End |

* User side function
* Login

|  |  |  |
| --- | --- | --- |
| User Login | | |
|  | FrontEnd | BackEnd |
| Fail  Correct  Home Page  Error message  Valid login information  Fill in your login information  Login | Create new session  Check login information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | User chooses to log in | User | When the user click on login page |
| 2 | Fill in your account information | User | The account information includes:   * Username * Password |
| 3 | Error message | System | Display an error message to the user when one of the two pieces of information is missing |
| 4 | Check login information | System | The system checks the validity of the information |
| 5 | Error message | System | Display an error message the user when the login credentials check fails |
| 6 | Create new session | System | Create new session for user |

* Register

|  |  |  |
| --- | --- | --- |
| User Register | | |
|  | FrontEnd | BackEnd |
| Register  Fail  Correct  Login  Error message  Valid login information  Fill in your login information | Insert information  Check login information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | User chooses to register | User | When the user click on register page |
| 2 | Fill in your account information | User | The account information includes:   * Username * Password * Full name * Phone * Birthday * Email |
| 3 | Error message | System | Display an error message to the user when one of the six pieces of information is missing |
| 4 | Check register information | System | The system checks the validity of the information |
| 5 | Error message | System | Display an error message the user when the login credentials check fails |
| 6 | Insert information | System | Insert information into database |

* List Our Work

|  |  |  |
| --- | --- | --- |
| List Our Work | | |
|  | FrontEnd | BackEnd |
| Click on type of our work  Display Our Work | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Hover on Our Work | User | User hover on our work |
| 2 | Select type of our work | User | User click on type of our work |
| 3 | Display Our Work Detail | User | See information of item |

* List Campaign

|  |  |  |
| --- | --- | --- |
| List Campaign | | |
|  | FrontEnd | BackEnd |
| Display campaign detail  Click on campaign detail  Display campaign list  Click on campaign | Select information  Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on Campaign | User | User click on campaign |
| 2 | Select information | System | Select information from campaign |
| 3 | Display campaign list | User | See information of item |
| 4 | Click on campaign detail | User | User click on campaign detail |
| 5 | Select information | System | Select information from campaign detail |
| 6 | Display Campaign Detail | User | See information of item |

* List Blog

|  |  |  |
| --- | --- | --- |
| List Blog | | |
|  | FrontEnd | BackEnd |
| Display blog detail  Click on blog detail  Display blog list  Click on blog | Select information  Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on blog | User | User click on campaign |
| 2 | Select information | System | Select information from blog |
| 3 | Display blog list | User | See information of item |
| 4 | Click on blog detail | User | User click on blog detail |
| 5 | Select information | System | Select information from blog detail |
| 6 | Display blog detail | User | See information of item |

* List member

|  |  |  |
| --- | --- | --- |
| List Member | | |
|  | FrontEnd | BackEnd |
| Display member list  Click on member | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on member | User | User click on member |
| 2 | Select information | System | Select information from member |
| 3 | Display member list | User | See information of item |

* List partner

|  |  |  |
| --- | --- | --- |
| List Partner | | |
|  | FrontEnd | BackEnd |
| Display partner list  Click on partner | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on partner | User | User click on partner |
| 2 | Select information | System | Select information from member |
| 3 | Display partner list | User | See information of item |

* Donate

|  |  |  |
| --- | --- | --- |
| Donate | | |
|  | FrontEnd | BackEnd |
| Correct  Fail  Correct  Error message  Success message  Correct  Check login  Validate information  Fill donate information  Login page  Have logged in yet?  Click on donate  Fail  Correct | Insert information  Check donate information  Check login information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on donate | User | User click on donate |
| 2 | Check login information | System | Check if the user is logged in |
|  | Have logged in yet | User | If user is logged in , redirect to donate page. Otherwise, redirect to login page |
| 3 | Login page | User | Fill login information |
| 4 | Check login | User | User login success, redirect to donate page |
| 5 | Fill donate information | User | The donate information include:   * Campaign * Fund * Amount * Payment method |
| 6 | Validate information | System | Display an error message to the user when pieces of information is missing |
| 7 | Checking donate information | System | The system checks the validate of the information |
| 8 | Error message | System | Display an error message the user when the donate credentials check fails |
| 9 | Success message | System | Display success message the user when the donate credentials check correct |
| 10 | Insert information | System | Insert information into database |

* List campaign which user had donated (for logged in user)

|  |  |  |
| --- | --- | --- |
| List Campaign Donated | | |
|  | FrontEnd | BackEnd |
| Display your campaign  Click your campaign | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on your campaign | User | User click on your campaign |
| 2 | Select information | System | Select information from database |
| 3 | Display list of campaign which user had donated | User | See information of item |

* Change account information (for logged in user)

|  |  |  |
| --- | --- | --- |
| User Edit Account | | |
|  | FrontEnd | BackEnd |
| Edit account  Fail  Correct  Success message  Error message  Valid edit information  Fill in your edit form information | Update information  Check edit information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | User chooses to edit account | User | When the user click on edit account page |
| 2 | Fill in new your account information | User | The account information includes:   * Full name * Phone * Birthday * Email |
| 3 | Error message | System | Display an error message to the user when pieces of information is missing |
| 4 | Check edit information | System | The system checks the validate of the information |
| 5 | Error message | System | Display an error message the user when the edit credentials check fails |
| 6 | Update information | System | Update information into database |

tim

* Admin side function
* Login

|  |  |  |
| --- | --- | --- |
| User Login | | |
|  | FrontEnd | BackEnd |
| Fail  Correct  Home Page  Error message  Valid login information  Fill in your login information  Login | Check login information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | User chooses to log in | User | When the user click on login page |
| 2 | Fill in your account information | User | The account information includes:   * Username * Password |
| 3 | Error message | System | Display an error message to the user when one of the two pieces of information is missing |
| 4 | Check login information | System | The system checks the validity of the information |
| 5 | Error message | System | Display an error message the user when the login credentials check fails |
| 6 | Follow the next procedure | System |  |

* List our work

|  |  |  |
| --- | --- | --- |
| List Our Work | | |
|  | FrontEnd | BackEnd |
| Display our work list  Click our work | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on our work | User | User click on our work |
| 2 | Select information | System | Select information from database |
| 3 | Display list of our work | User | See information of item |

* Add our work

|  |  |  |
| --- | --- | --- |
| Add Our Work | | |
|  | FrontEnd | BackEnd |
| Click on add our work  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Check form information  Insert information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | User click on add our work | User | When the user click on add our work page |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display an error message the user when the form credentials check fails |
| 6 | Insert information | System | Insert information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Edit our work

|  |  |  |
| --- | --- | --- |
| Edit Our Work | | |
|  | FrontEnd | BackEnd |
| Click on edit our work  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Select our work information  Check form information  Update information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step  code | Step name | Group of people | Describe |
| 1 | User click on edit our work | User | When the user click on edit our work page |
| 2 | Select our work information | System | Select our work information from database |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display error message the user when the form credentials check fails |
| 6 | Update information | System | Update information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Delete our work

|  |  |
| --- | --- |
| Delete Our Work | |
| FrontEnd | BackEnd |
| Click our work record  Remove successful | Update status and hidden into database |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Select our work want to delete | User |  |
| 2 | Remove record | System | Change status in database (Update status and hide record, not remove) |
| 3 | Remove successfully | System | Display success message |

* List fund

|  |  |  |
| --- | --- | --- |
| List Fund | | |
|  | FrontEnd | BackEnd |
| Display fund list  Click fund | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on fund | User | User click on fund |
| 2 | Select information | System | Select information from database |
| 3 | Display list of fund | User | See information of item |

* Add fund

|  |  |  |
| --- | --- | --- |
| Add Fund | | |
|  | FrontEnd | BackEnd |
| Click on add fund  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Check form information  Insert information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | User click on add fund | User | When the user click on add fund page |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display an error message the user when the form credentials check fails |
| 6 | Insert information | System | Insert information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Edit fund

|  |  |  |
| --- | --- | --- |
| Edit Fund | | |
|  | FrontEnd | BackEnd |
| Click on edit fund  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Select fund information  Check form information  Update information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step  code | Step name | Group of people | Describe |
| 1 | User click on edit fund | User | When the user click on edit fund page |
| 2 | Select fund information | System | Select fund information from database |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display error message the user when the form credentials check fails |
| 6 | Update information | System | Update information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Delete fund

|  |  |
| --- | --- |
| Delete Fund | |
| FrontEnd | BackEnd |
| Click case file record  Remove successful | Update status and hidden into database  Update status and hidden into database |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Select fund want to delete | User |  |
| 2 | Remove record | System | Change status in database (Update status and hide record, not remove) |
| 3 | Remove successfully | System | Display success message |

* List campaign

|  |  |  |
| --- | --- | --- |
| List Campaign | | |
|  | FrontEnd | BackEnd |
| Display campaign list  Click campaign | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on campaign | User | User click on campaign |
| 2 | Select information | System | Select information from database |
| 3 | Display list of campaign | User | See information of item |

* Add campaign

|  |  |  |
| --- | --- | --- |
| Add Campaign | | |
|  | FrontEnd | BackEnd |
| Click on add campaign  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Check form information  Insert information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | User click on add campaign | User | When the user click on add campaign page |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display an error message the user when the form credentials check fails |
| 6 | Insert information | System | Insert information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Edit campaign

|  |  |  |
| --- | --- | --- |
| Edit Campaign | | |
|  | FrontEnd | BackEnd |
| Click on edit campaign  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Select campaign information  Check form information  Update information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step  code | Step name | Group of people | Describe |
| 1 | User click on edit campaign | User | When the user click on edit campaign page |
| 2 | Select fund information | System | Select campaign information from database |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display error message the user when the form credentials check fails |
| 6 | Update information | System | Update information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Delete campaign

|  |  |
| --- | --- |
| Delete Campaign | |
| FrontEnd | BackEnd |
| Click campaign record  Remove successful | Update status and hidden into database |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Select campaign want to delete | User |  |
| 2 | Remove record | System | Change status in database (Update status and hide record, not remove) |
| 3 | Remove successfully | System | Display success message |

* List blog

|  |  |  |
| --- | --- | --- |
| List Blog | | |
|  | FrontEnd | BackEnd |
| Display blog list  Click blog | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on blog | User | User click on blog |
| 2 | Select information | System | Select information from database |
| 3 | Display list of blog | User | See information of item |

* Add blog

|  |  |  |
| --- | --- | --- |
| Add Blog | | |
|  | FrontEnd | BackEnd |
| Click on add blog  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Check form information  Insert information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | User click on add blog | User | When the user click on add blog page |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display an error message the user when the form credentials check fails |
| 6 | Insert information | System | Insert information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Edit blog

|  |  |  |
| --- | --- | --- |
| Edit Blog | | |
|  | FrontEnd | BackEnd |
| Click on edit blog  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Select blog information  Check form information  Update information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step  code | Step name | Group of people | Describe |
| 1 | User click on edit blog | User | When the user click on edit blog page |
| 2 | Select blog information | System | Select blog information from database |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display error message the user when the form credentials check fails |
| 6 | Update information | System | Update information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Delete blog

|  |  |
| --- | --- |
| Delete Blog | |
| FrontEnd | BackEnd |
| Click blog record  Remove successful | Update status and hidden into database |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Select blog want to delete | User |  |
| 2 | Remove record | System | Change status in database (Update status and hide record, not remove) |
| 3 | Remove successfully | System | Display success message |

* List member

|  |  |  |
| --- | --- | --- |
| List Member | | |
|  | FrontEnd | BackEnd |
| Display member list  Click member | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on member | User | User click on member |
| 2 | Select information | System | Select information from database |
| 3 | Display list of member | User | See information of item |

* Add member

|  |  |  |
| --- | --- | --- |
| Add Member | | |
|  | FrontEnd | BackEnd |
| Click on add member  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Check form information  Insert information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | User click on add member | User | When the user click on add member page |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display an error message the user when the form credentials check fails |
| 6 | Insert information | System | Insert information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Edit member

|  |  |  |
| --- | --- | --- |
| Edit Member | | |
|  | FrontEnd | BackEnd |
| Click on edit member  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Select member information  Check form information  Update information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step  code | Step name | Group of people | Describe |
| 1 | User click on edit member | User | When the user click on edit member page |
| 2 | Select member information | System | Select member information from database |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display error message the user when the form credentials check fails |
| 6 | Update information | System | Update information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Delete member

|  |  |
| --- | --- |
| Delete Member | |
| FrontEnd | BackEnd |
| Click member record  Remove successful | Update status and hidden into database |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Select member want to delete | User |  |
| 2 | Remove record | System | Change status in database (Update status and hide record, not remove) |
| 3 | Remove successfully | System | Display success message |

* List partner

|  |  |  |
| --- | --- | --- |
| List Partner | | |
|  | FrontEnd | BackEnd |
| Display partner list  Click partner | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on partner | User | User click on partner |
| 2 | Select information | System | Select information from database |
| 3 | Display list of partner | User | See information of item |

* Add partner

|  |  |  |
| --- | --- | --- |
| Add Partner | | |
|  | FrontEnd | BackEnd |
| Click on add partner  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Check form information  Insert information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | User click on add partner | User | When the user click on add partner page |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display an error message the user when the form credentials check fails |
| 6 | Insert information | System | Insert information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Edit partner

|  |  |  |
| --- | --- | --- |
| Edit Partner | | |
|  | FrontEnd | BackEnd |
| Click on edit partner  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Select partner information  Check form information  Update information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step  code | Step name | Group of people | Describe |
| 1 | User click on edit partner | User | When the user click on edit partner page |
| 2 | Select partner information | System | Select partner information from database |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display error message the user when the form credentials check fails |
| 6 | Update information | System | Update information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Delete partner

|  |  |
| --- | --- |
| Delete Partner | |
| FrontEnd | BackEnd |
| Click partner record  Remove successful | Update status and hidden into database |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Select partner want to delete | User |  |
| 2 | Remove record | System | Change status in database (Update status and hide record, not remove) |
| 3 | Remove successfully | System | Display success message |

* List account (for host)

|  |  |  |
| --- | --- | --- |
| List Account | | |
|  | FrontEnd | BackEnd |
| Display account list  Click account | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on account | User | User click on account |
| 2 | Select information | System | Select information from database |
| 3 | Display list of account | User | See information of item |

* Add account (for host)

|  |  |  |
| --- | --- | --- |
| Add Account | | |
|  | FrontEnd | BackEnd |
| Click on add account  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Check form information  Insert information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | User click on add account | User | When the user click on add account page |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display an error message the user when the form credentials check fails |
| 6 | Insert information | System | Insert information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Edit account (for host)

|  |  |  |
| --- | --- | --- |
| Edit Account | | |
|  | FrontEnd | BackEnd |
| Click on edit account  Fail  Correct  Success message  Error message  Valid form information  Fill in form information | Select account information  Check form information  Update information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step  code | Step name | Group of people | Describe |
| 1 | User click on edit account | User | When the user click on edit account page |
| 2 | Select account information | System | Select account information from database |
| 2 | Fill in form information | User | User fill in form |
| 3 | Valid form information | System | Display an error message to the user when pieces of information is missing |
| 4 | Check form information | System | The system checks the validate of the information |
| 5 | Error message | System | Display error message the user when the form credentials check fails |
| 6 | Update information | System | Update information into database |
| 7 | Success message | System | Display success message the user when the form credentials check correct |

* Delete account (for host)

|  |  |
| --- | --- |
| Delete account | |
| FrontEnd | BackEnd |
| Click account record  Remove successful | Update status and hidden into database |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Select account want to delete | User |  |
| 2 | Remove record | System | Change status in database (Update status and hide record, not remove) |
| 3 | Remove successfully | System | Display success message |

* List statistic of fund

|  |  |  |
| --- | --- | --- |
| List Statistic Of Fund | | |
|  | FrontEnd | BackEnd |
| Display fund list with total money  Click statistic of fund | Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on statistic of fund | User | User click on statistic of fund |
| 2 | Select information | System | Select information from database |
| 3 | Display list of statistic | User | See information of item |

* List Statistic Of Donor

|  |  |  |
| --- | --- | --- |
| List Campaign | | |
|  | FrontEnd | BackEnd |
| Display donor detail  Click on donor detail  Display donor list  Click on donor | Select information  Select information |

Describe the steps in the business process

|  |  |  |  |
| --- | --- | --- | --- |
| Step code | Step name | Group of people | Describe |
| 1 | Click on statistic of donor | User | User click on statistic of donor |
| 2 | Select information | System | Select information from database |
| 3 | Display donor list | User | See information of item |
| 4 | Click on donor detail | User | User click on donor detail |
| 5 | Select information | System | Select information from donor and statistic money of campaign donated |
| 6 | Display Donor Detail | User | See information of item |

* + 1. **Boundaries of the System**
* In the context of a non-governmental organization's website, the system's limitations can be defined as follows:
* **Customers:** The customers of the non-governmental organization's website may be citizens, businesses, non-profit organizations, or anyone interested in the information, services, or products provided by the organization. The limitation of this system is the end users and the community that the organization aims to serve and interact with.
* **User groups:** The user groups of the non-governmental organization's website could include all users accessing the website to retrieve information, use online services, or participate in interactive activities such as discussions or feedback. These user groups may include individuals and organizations.
* **Scope of the website’s application:** The non-governmental organization's website may provide information and services related to a variety of fields and issues, but there are limits to the areas in which the organization operates and its goals. This scope can be defined by specific programs, projects, or activities that the organization is implementing.

These limitations help clearly define the website's objectives, the audience it aims to serve, and the scope of the activities and services it provides. This helps optimize user experience and the organization's performance.

* + 1. **Hardware and Software Requirements**
* Hardware:
* A minimum computer system that will help you access all the tools in the
* courses is a Pentium 166 or better
* 64 Megabytes of RAM or better
* Windows 98 (or higher if possible)
* Java Virtual Machine / Dot Net Framework
* Software:
* Notepad/HTML editor/Java editor
* Dream weaver/JavaScript
* j2sdk1.4.1\_02 (or later) / .Net / J2EE
* JSP / Servlets
* EJB / Struts
* JDBC
* SQL Server 2000 (or later) / Oracle 9i / MS Access
* JDBC Driver

# Task sheet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task | Start time | Finish time | Implemented by | Evaluation  (%) |
| ***Feasibility study phase*** | | | | |
| Create memo | 21/02/2024 | 28/02/2024 | Dang Dinh Thuan | 25 |
| Create opinions | 21/02/2024 | 28/02/2024 | Tran Ha Anh | 25 |
| Create opinions | 21/02/2024 | 28/02/2024 | Nguyen Giang Thanh | 20 |
| Create opinions | 21/02/2024 | 28/02/2024 | Le Thanh Dat | 15 |
| Create opinions | 21/02/2024 | 28/02/2024 | Pham Doan Toan | 15 |
| ***Requirement Analysis phase*** | | | | |
| Create memo | 01/03/2024 | 08/03/2024 | Dang Dinh Thuan | 25 |
| Create opinions | 01/03/2024 | 08/03/2024 | Tran Ha Anh | 25 |
| Create opinions | 01/03/2024 | 08/03/2024 | Nguyen Giang Thanh | 20 |
| Create opinions | 01/03/2024 | 08/03/2024 | Le Thanh Dat | 15 |
| Create opinions | 01/03/2024 | 08/03/2024 | Pham Doan Toan | 15 |
| ***Design phase - Template*** | | | | |
| FE - Create Home Template | 10/03/2024 | 17/03/2024 | Tran Ha Anh | 10 |
| FE - Create Our Work Template | 10/03/2024 | 17/03/2024 | Nguyen Giang Thanh | 10 |
| FE - Create Campaign Template | 10/03/2024 | 17/03/2024 | Tran Ha Anh | 10 |
| FE - Create Campaign-Detail Template | 10/03/2024 | 17/03/2024 | Tran Ha Anh | 5 |
| FE - Create Blog Template | 10/03/2024 | 17/03/2024 | Le Thanh Dat | 10 |
| FE - Create Blog-Detail Template | 10/03/2024 | 17/03/2024 | Dang Dinh Thuan | 5 |
| FE - Create About Us Template | 10/03/2024 | 17/03/2024 | Dang Dinh Thuan | 5 |
| FE - Create What We Do Template | 10/03/2024 | 17/03/2024 | Pham Doan Toan | 5 |
| FE - Create Our Mission Template | 10/03/2024 | 17/03/2024 | Pham Doan Toan | 5 |
| FE - Create Our Team Template | 10/03/2024 | 17/03/2024 | Nguyen Giang Thanh | 5 |
| FE - Create Our Partner Template | 10/03/2024 | 17/03/2024 | Le Thanh Dat | 5 |
| FE - Create Contact Us Template | 10/03/2024 | 17/03/2024 | Dang Dinh Thuan | 5 |
| FE - Create Login Template | 10/03/2024 | 17/03/2024 | Nguyen Giang Thanh | 10 |
| FE – Create Register Template | 10/03/2024 | 17/03/2024 | Nguyen Giang Thanh | 10 |
| ***Design phase - Front End*** | | | | |
| FE - Complete Home Page | 19/03/2024 | 23/03/2024 | Tran Ha Anh | 10 |
| FE - Complete Our Work Page | 19/03/2024 | 23/03/2024 | Nguyen Giang Thanh | 10 |
| FE - Complete Campaign Page | 19/03/2024 | 23/03/2024 | Tran Ha Anh | 5 |
| FE - Complete Campaign-Detail Page | 19/03/2024 | 23/03/2024 | Tran Ha Anh | 10 |
| FE - Complete Blog Page | 19/03/2024 | 23/03/2024 | Le Thanh Dat | 5 |
| FE - Complete Blog-Detail Page | 19/03/2024 | 23/03/2024 | Dang Dinh Thuan | 5 |
| FE - Complete About Us Page | 19/03/2024 | 23/03/2024 | Dang Dinh Thuan | 5 |
| FE - Complete What We Do Page | 19/03/2024 | 23/03/2024 | Pham Doan Toan | 5 |
| FE - Complete Our Mission Page | 19/03/2024 | 23/03/2024 | Pham Doan Toan | 5 |
| FE - Complete Our Team Page | 19/03/2024 | 23/03/2024 | Nguyen Giang Thanh | 5 |
| FE - Complete Our Partner Page | 19/03/2024 | 23/03/2024 | Le Thanh Dat | 5 |
| FE - Complete Contact Us Page | 19/03/2024 | 23/03/2024 | Dang Dinh Thuan | 10 |
| FE - Complete Login Page | 19/03/2024 | 23/03/2024 | Nguyen Giang Thanh | 10 |
| FE – Complete Register Page | 19/03/2024 | 23/03/2024 | Nguyen Giang Thanh | 10 |
| ***Design phase – Back End*** | | | | |
| BE – Complete Home | 19/03/2024 | 23/03/2024 | Tran Ha Anh | 10 |
| BE – Complete Our Work | 19/03/2024 | 23/03/2024 | Dang Dinh Thuan | 10 |
| BE - Complete Campaign | 19/03/2024 | 23/03/2024 | Dang Dinh Thuan | 10 |
| BE - Complete Campaign-Detail | 19/03/2024 | 23/03/2024 | Nguyen Giang Thanh | 10 |
| BE - Complete Blog | 19/03/2024 | 23/03/2024 | Dang Dinh Thuan | 10 |
| BE - Complete Blog-Detail | 19/03/2024 | 23/03/2024 | Dang Dinh Thuan | 10 |
| BE - Complete About Us | 19/03/2024 | 23/03/2024 | Nguyen Giang Thanh | 10 |
| BE - Complete What We Do | 19/03/2024 | 23/03/2024 | Nguyen Giang Thanh | 10 |
| FE - Complete Our Mission | 19/03/2024 | 23/03/2024 | Pham Doan Toan | 10 |
| BE - Complete Our Team | 19/03/2024 | 23/03/2024 | Le Thanh Dat | 10 |
| BE - Complete Our Partner | 19/03/2024 | 23/03/2024 | Le Thanh Dat | 10 |
| BE – Complete Login-Register | 19/03/2024 | 23/03/2024 | Pham Doan Toan | 10 |
| ***Testing phase*** | | | | |
| FE - Fix User Page | 24/03/2024 | 29/03/2024 | Tran Ha Anh  Nguyen Giang Thanh | 25 |
| BE - Fix User Logic | 24/03/2024 | 29/03/2024 | Dang Dinh Thuan  Le Thanh Dat  Pham Doan Toan | 25 |
| FE - Fix Admin Page | 24/03/2024 | 29/03/2024 | Tran Ha Anh  Nguyen Giang Thanh | 25 |
| BE - Fix Admin Logic | 24/03/2024 | 29/03/2024 | Dang Dinh Thuan  Le Thanh Dat  Pham Doan Toan | 25 |
| ***Deployment phase*** | | | | |
| Submit Source code | 29/03/2024 | 29/03/2024 | Dang Dinh Thuan | 15 |
| Submit Project Doc | 29/03/2024 | 29/03/2024 | Tran Ha Anh  Nguyen Giang Thanh  Pham Doan Toan | 70 |
| Submit Task Sheet | 29/03/2024 | 29/03/2024 | Dang Dinh Thuan | 15 |

# Validation Checklists

|  |  |
| --- | --- |
| Option | Validated |
| Do all image links lead to the appropriate App? | Yes |
| Are all the images and link visible on the page? | Yes |
| Is the app page load quickly? | Yes |
| All form has been validated? | Yes |