Table of Contents

*[Introduction](#_Toc449450998)* [3](#_Toc449450998)

[Bios 3](#_Toc449450999)

[Alexander V. 3](#_Toc449451000)

[Bin 3](#_Toc449451001)

[Fatemeh 3](#_Toc449451002)

[Lee 3](#_Toc449451003)

[Susan 3](#_Toc449451004)

[*Rationale and project goals* 3](#_Toc449451005)

[*Third party libraries and code* 4](#_Toc449451006)

[1. jQuery version 1.10.2 4](#_Toc449451007)

[2. Bootstrap.js 4](#_Toc449451008)

[3.jQuery.unobtrusive-ajax.min.js 4](#_Toc449451009)

[4. SignalRChat 4](#_Toc449451010)

[5. PayPal 4](#_Toc449451011)

[*Personas (n/a)* 4](#_Toc449451012)

[*Database schema (n/a)* 7](#_Toc449451013)

[Content plan 7](#_Toc449451014)

[NotreDame Heart Navigation 7](#_Toc449451015)

[Associate Dean (Heather) 8](#_Toc449451016)

[Erica (staff introduced by Sean) 9](#_Toc449451018)

[Alphosus Eze (Classmate) CardSort 9](#_Toc449451019)

[Eric Westcott(Classmate) Card Sort 10](#_Toc449451020)

[*User roles (n/a)* 11](#_Toc449451021)

[*Code (n/a)* 11](#_Toc449451022)

[*Master page(n/a)* 11](#_Toc449451023)

[*CSS (n/a)* 12](#_Toc449451024)

[*Features (n/a)* 12](#_Toc449451025)

[Make an appointment 12](#_Toc449451026)

[Narratives 12](#_Toc449451027)

[Database 12](#_Toc449451028)

[Flow Chart 13](#_Toc449451029)

[Wireframes 15](#_Toc449451030)

[Navigation Management 18](#_Toc449451031)

[Narratives 18](#_Toc449451032)

[Database 18](#_Toc449451033)

[Flow Chart 19](#_Toc449451034)

[Wireframes 20](#_Toc449451035)

[Add a new page 21](#_Toc449451036)

[Narratives 21](#_Toc449451037)

[Database 21](#_Toc449451038)

[Flow Chart 22](#_Toc449451039)

[Wireframes 23](#_Toc449451040)

[Bin Liu 24](#_Toc449451041)

[Feedback Page 24](#_Toc449451042)

[Feed Back Database 24](#_Toc449451043)

[Job Posting Page 27](#_Toc449451044)

[Job Posting Database 27](#_Toc449451045)

[Parking payment Page 30](#_Toc449451046)

[Parking payment Database 30](#_Toc449451047)

[PHP Online exam Page 32](#_Toc449451048)

[Parking payment Database 32](#_Toc449451049)

[*Development schedule (n/a)* 5](#_Toc449451050)

[*Maintenance plan* 5](#_Toc449451051)

[*Handover plan* 5](#_Toc449451052)

# Introduction

# Bios

### Alexander V.

### Bin

### Fatemeh

Fatemeh Abdizadeh is a web developing student at Humber College who used to work as a problem solving IT supporter in Iran Railways and also junior web developer there. She has a B.A. in Software Engineering and also massive knowledge in backend developing especially with .NET.

### Lee

Lee is a full-stack web developer from Humber College. Having just completed a tenure as an English teacher in Korea and returned from a lengthy journey in Far East, he has a lot of knowledge in understanding other peoples’ wants and needs. Lee has a knack for database solutions when it comes to developing websites. He is based in Toronto and spends his free time with his disobedient shiba inu.

### Susan

Names – no pics

# Rationale and project goals

The goal of this project is to establish a web presence for NDH on all browsers and devices. Previous versions of the website were non-functioning on most web browsers. The new website is to also adhere to AODA standards. This means the web site is to be accessible to persons with disabilities. New developments previously unavailable are to implement mobile layouts of the website so NDH can be viewed comfortable on smaller screens. The redesign of NDH should also be able to attract new staff and donations to the organization in a hassle-free manner.

# *Third party libraries and code*

The following lists the third party libraries utilized in this project. They are available from Visual Studio’s nuget package manager.

## 1. jQuery version 1.10.2

File location: Scripts

Javascript framework that allows concise javascript code and provides a base for other libraries used on this website.

## 2. Bootstrap.js

File location: Scripts

Html framework that uses javascript to quickly scaffold responsive HTML designs.

## 3.jQuery.unobtrusive-ajax.min.js

File location: scripts

jQuery library that allows asynchronous client-side validation of the application without reloading the page. Validation messages appear next to input fields when user has made an error.

## 4. SignalRChat

File locations: References, Scripts

Creates a host that enables two or more users to engage in asynchronous chat. Obtained from NuGet package manager.

## 5. PayPal

File location:

Application that handles user monetary transactions securely. Redirects the user to a payment window.

# Personas (n/a)

**Name**: Jacques Berndadette

**Age**: 60

**Occupation**: Farmer

**Background**: Grew up in French rural Ontario all his life, his children are adults with their own families. He wakes up at 5:30 everyday and tends to his fields with his wife. If he ever uses a computer, it's in the evening to keep in touch with his family and business enquiries.

**Location:** Hearst, Ontario

**Devices**: Desktop computer

**Internet speed:** 1mb/s

**Internet usage:** Occasional email use

**Goals:** Have a bountiful harvest by autumn

**Primary language:** French

**Name**: Antoinette Smith

**Age**: 32

**Occupation**: Graphic designer

**Background**: Currently part time of her job because she is 6 months pregnant with her husband’s child. She goes to work 3 days a week and is very familiar with photoshop. To past the time she watches Netflix on the TV.

**Location:** Thunder Bay, Ontario

**Devices**: iphone, Macintoshcomputer

**Internet speed:** 1mb/s

**Internet usage:** WhatsApps her husband. Netflix. Email to send design files.

**Goals:** Looking forward to the birth of her first child

**Primary language:** English

**Name**: Kim chong

**Age**: 26

**Occupation**:Live streamer

**Background**: Corporate 9-5 in an insurance company turned live streamer, Kim eats meals in front of internet audiences all day. Her income is from viewers sending gifts of monetary value in the chat. Her new found profession requires her to be internet savvy and understand viewing trends. She studied English in high school but is still intimidated by it.

**Location:** Seoul, South Korea

**Devices**: Desktop computer with web camera. Samsung note 2 phone.

**Internet speed:** 50 mb/s

**Internet usage:** KakaoTalk, LINE, afreecaTV.

**Goals:** Break a million views in a single broadcast.

**Primary language:** Korean

**Name**: h@x0r

**Age**: 25

**Occupation**: Freelance hacker

**Background**: Studied computer science for his undergraduate degree but dropped out in his second year. Spends most of his day at home in front of the computer on internet relay chat rooms with other computer savvy netizens. Voted "biggest nerd" in his high school graduation yearbook. Very knowledgeable in web-related technologies. Decent hacker that hides behind multiple VPNs and has not been caught by the authorities for his actions.

**Location:** Seattle, Washington

**Devices**: Custom built desktop computer, rooted smartphone, raspberry PI, 2 rooted tablets

**Internet speed:** 5 mb/s

**Internet usage:** Frequent IRC use, email, online gaming

**Goals:** Hack CITI Bank

**Primary language:** English

**Name**: Oliver Reid

**Age**: 22

**Occupation**: Medical student

**Background**: Very studious, punctual, and highly praised by his superiors. He has a bright future ahead of him if he continues along the path he has right now. He knows medical school is very competitive and has been sending out letters and applications to hospitals all over the province. He wants to get in touch with Ontario’s hospitals to get familiar with his future colleagues but finds the state of hospital websites in Ontario to be all over the place.

**Location:** Toronto, Ontario

**Devices**: Desktop, iPad, Smart Phone

**Internet speed:** 5 mb/s

**Internet usage:** Frequent IRC use, email, online gaming

**Goals:** Find a residency to complete his studies

**Primary language:** English

# Database schema (n/a)

Overall

# Content plan

## NotreDame Heart Navigation



## Associate Dean (Heather)

# 



## Erica (staff introduced by Sean)



# 

## Alphosus Eze (Classmate) CardSort



## Eric Westcott(Classmate) Card Sort



# User roles (n/a)

Fatemeh – Backend Developer

# Code (n/a)

Refer to Appendix A

# Master page(n/a)

\_\_Layout – template wireframe

# CSS (n/a)

Refer to Appendix B

# Features (n/a)

Fatemeh Abdizadeh

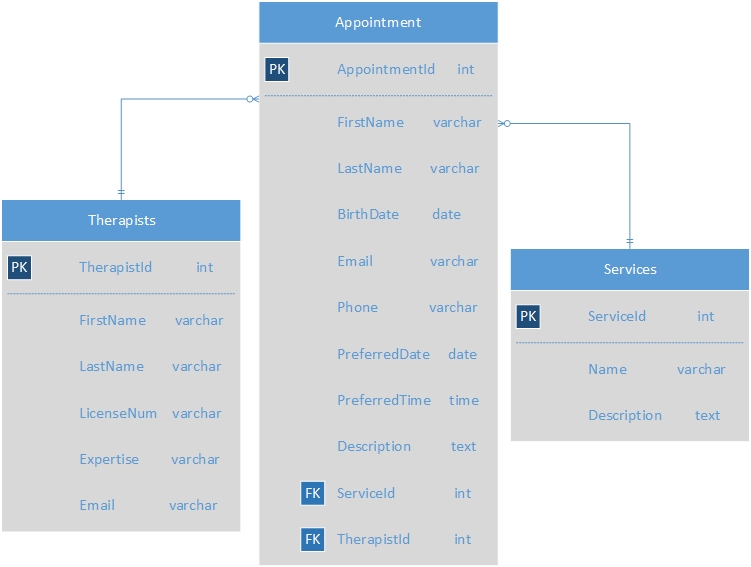
## Make an appointment

### Narratives

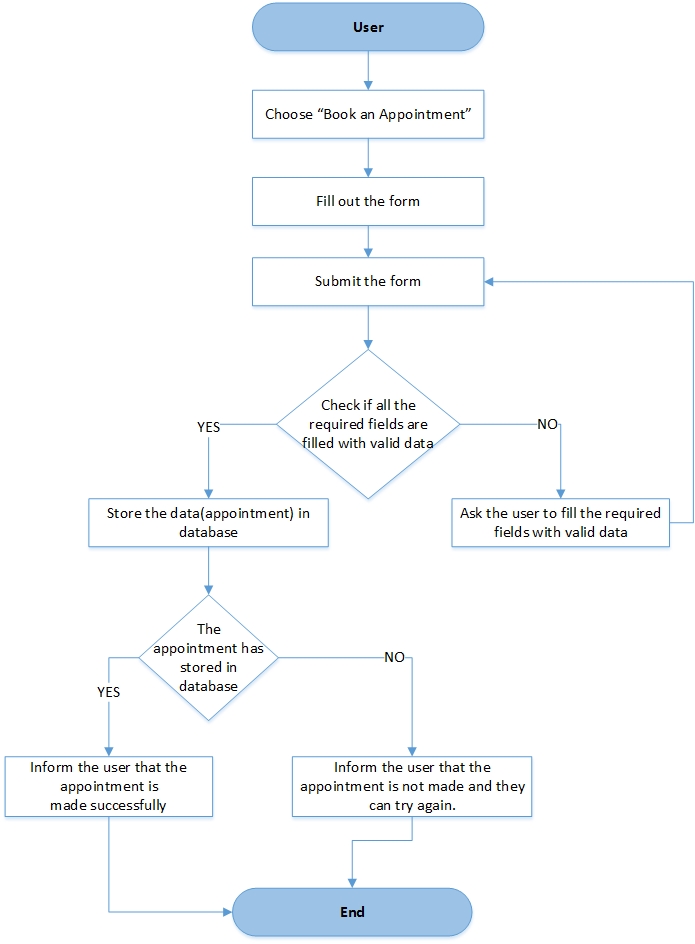
Users will be able to fill out a form including their personal information like first name, last name, birthday date, email address, phone number etc. Then, they are supposed to choose a service using a dropdown list such as physiotherapy. After that they can choose a preferred therapist from a dropdown list. They will be able to set their preferred date and time as well. At the end they should submit the form.

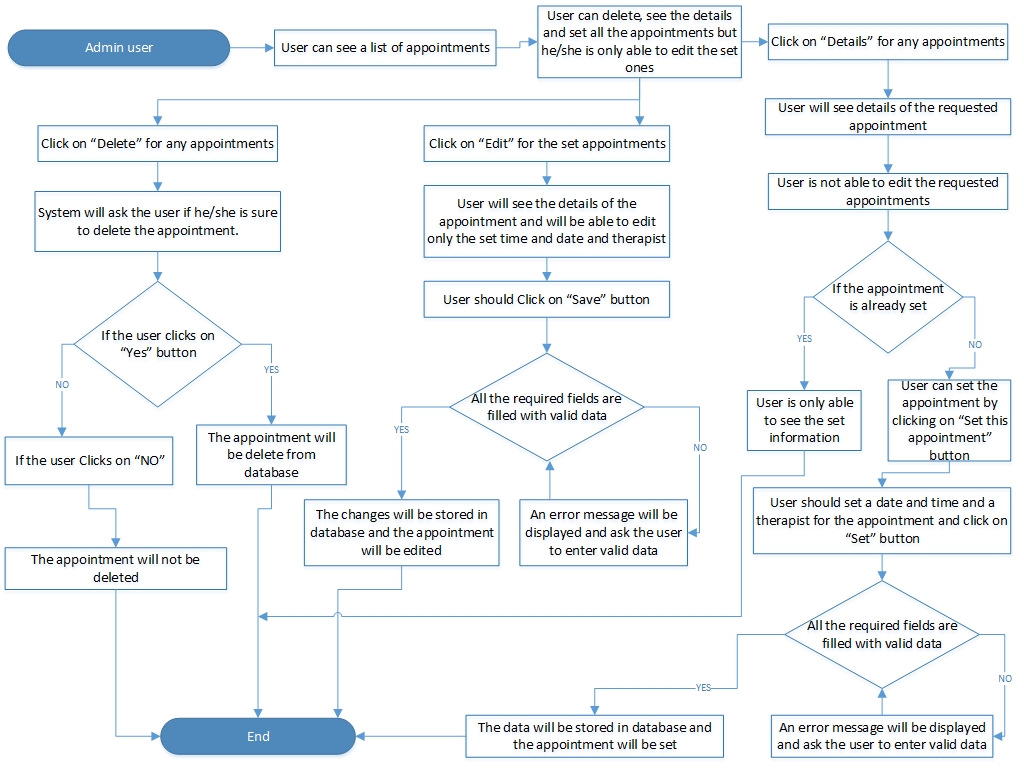
System will check if any required fields are left blank or contain invalid data, if so error messages will be displayed. Otherwise all the data for this appointment will be saved in the database and a message will be display to confirm that they have booked an appointment successfully. If for some reason the data does not save in the database, an error message will be displayed and ask them to try again.

### Database



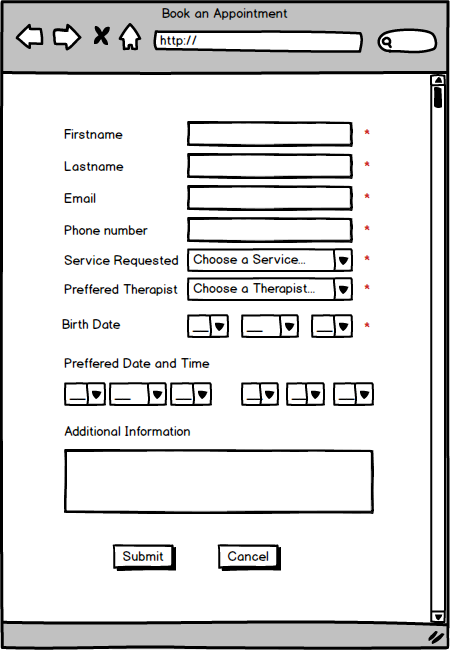
### Flow Chart

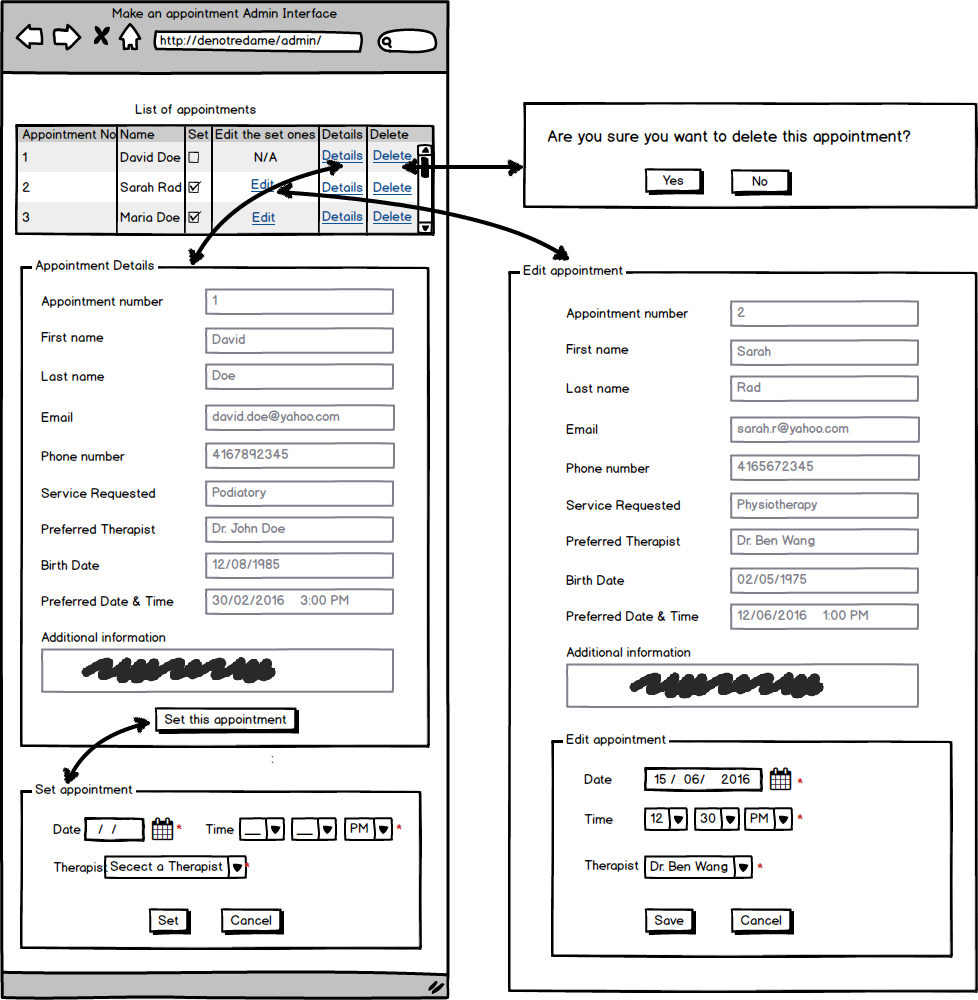
****



### 

### Wireframes





## Navigation Management

### Narratives

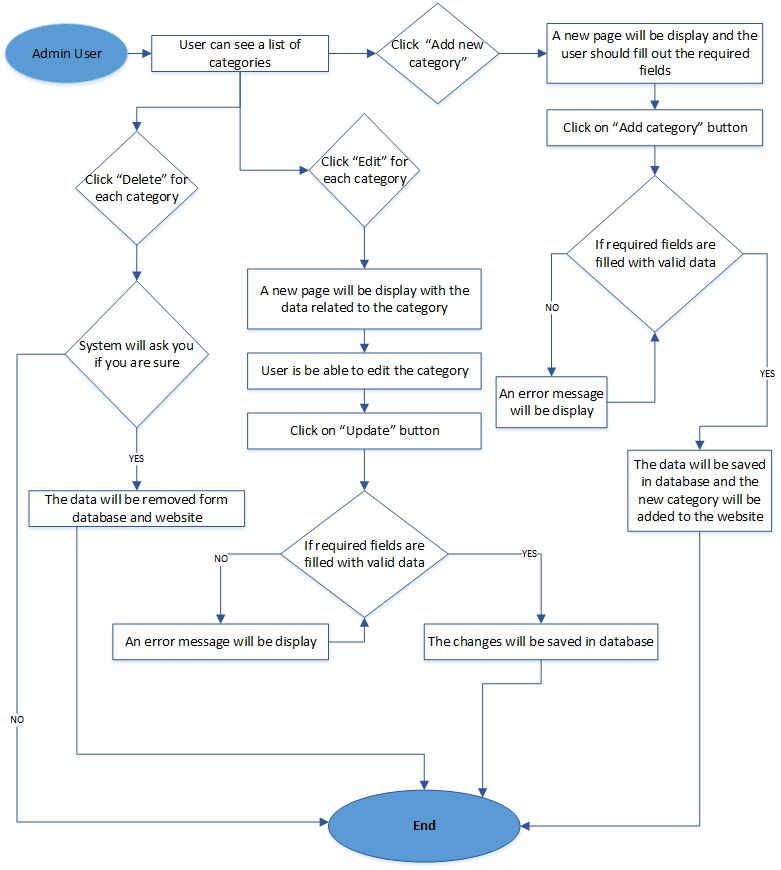
In order to add a new link (category) or edit and delete the current ones, we need to be able to manage the navigation which only admin users will be able to make it once they log in successfully into the content management system. They can see the list of categories and by clicking on “Edit” or “Delete” they can make changes.

If they click on the “add new category” button a new page will be displayed in which they are supposed to enter a name and choose a parent link from a dropdown list and write a description and finally click on “Add category” button. System will check if any required fields are left blank or filled with invalid data if so an error message will be displayed. Otherwise new data will be stored in database and the new link will be added to the website.

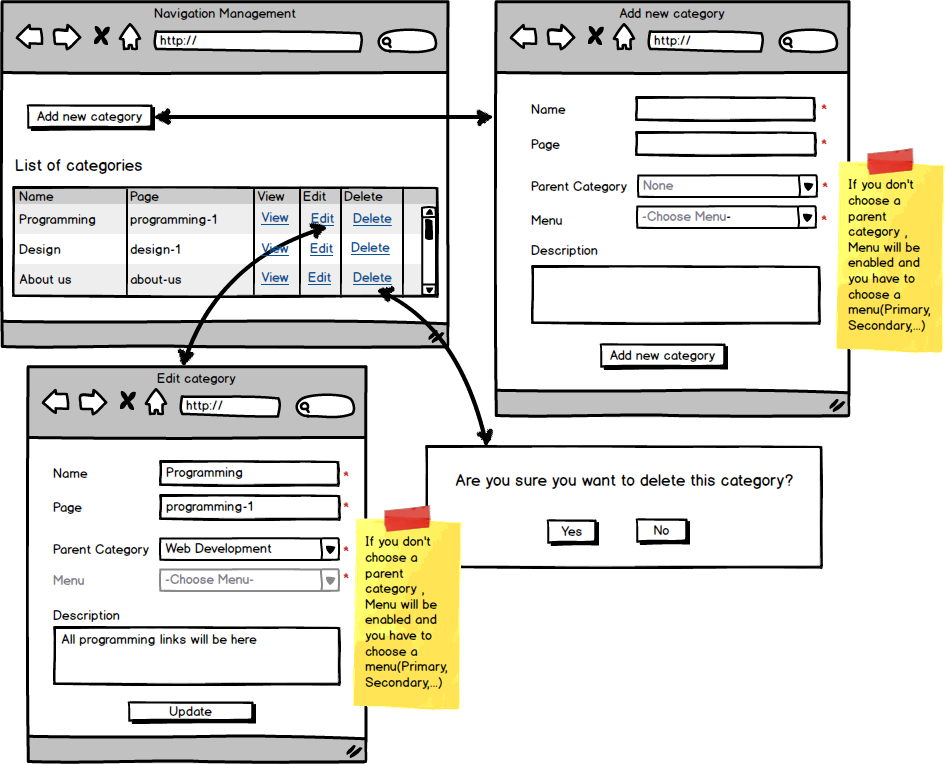
### Database

****

### Flow Chart

****

### Wireframes

****

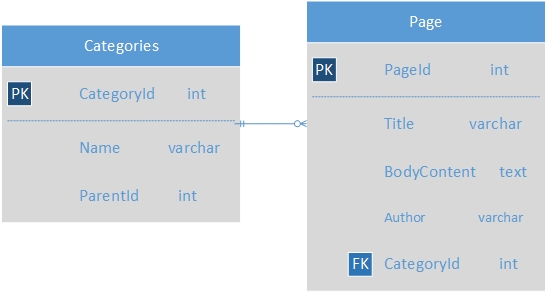
## Add a new page

### Narratives

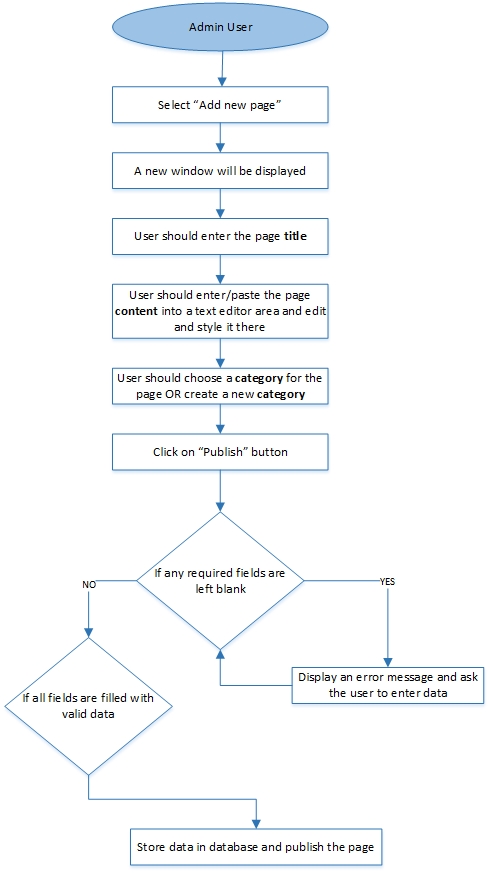
Only admin users have access to create a new page once they log in successfully into the content management system. When admin users click on “Add new page” button, a new window will be displayed and they are supposed to enter a title for the page and there is a text editor in which they can type or paste the content and edit and style it. Next step is to either choose one of the existing categories which is displayed to them or create a new category by clicking on the “Add a new category” button which will give them a form that they are supposed to give a name to the new category and select the parent category from a dropdown list options.

At the end they are supposed to click on “Publish” button. System will check if any required fields are left blank, if so an error message will be shown. Only if no required fields are left blank or filled with invalid data, system will store the data in the database and publish the new page.

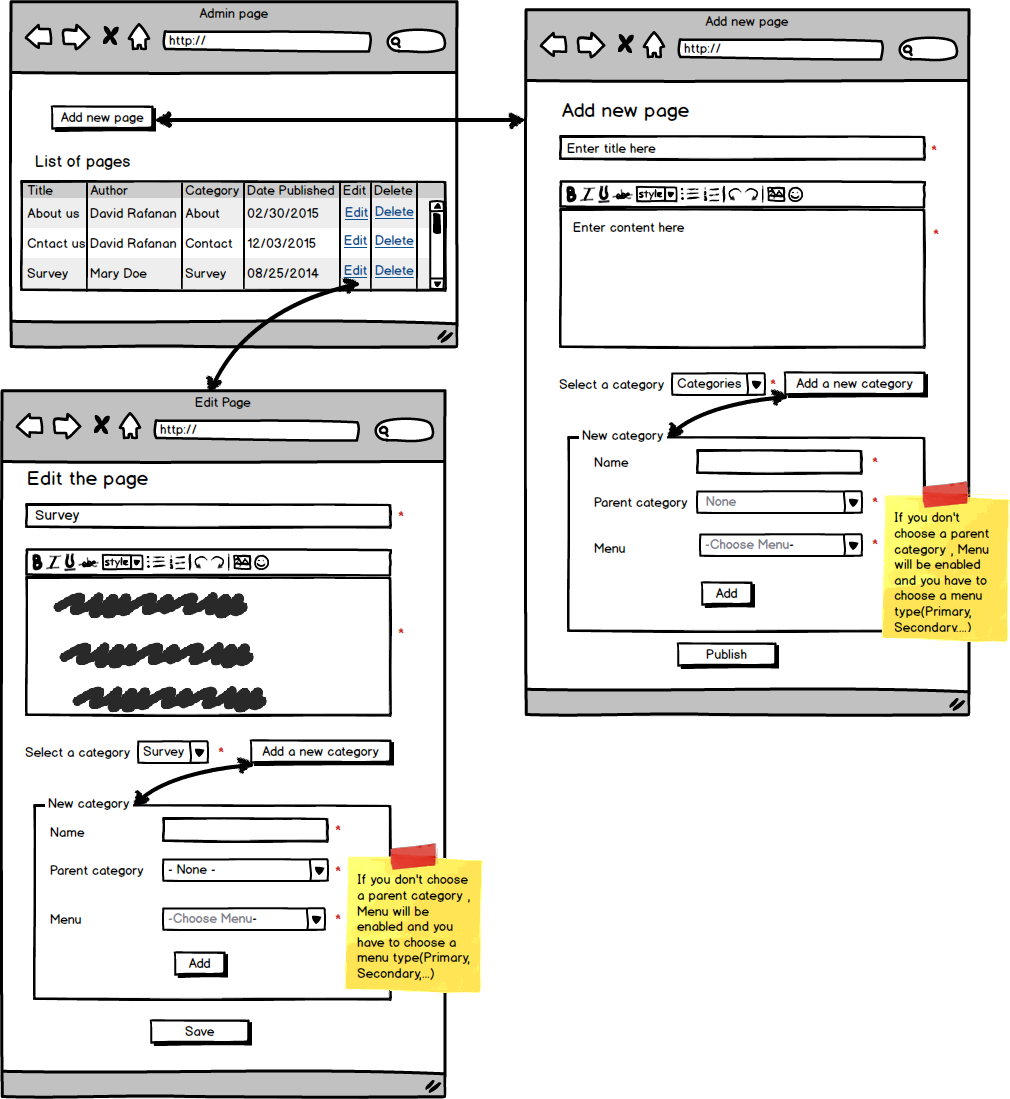
### Database



### Flow Chart

****

### Wireframes

****

## Bin Liu

## Feedback Page

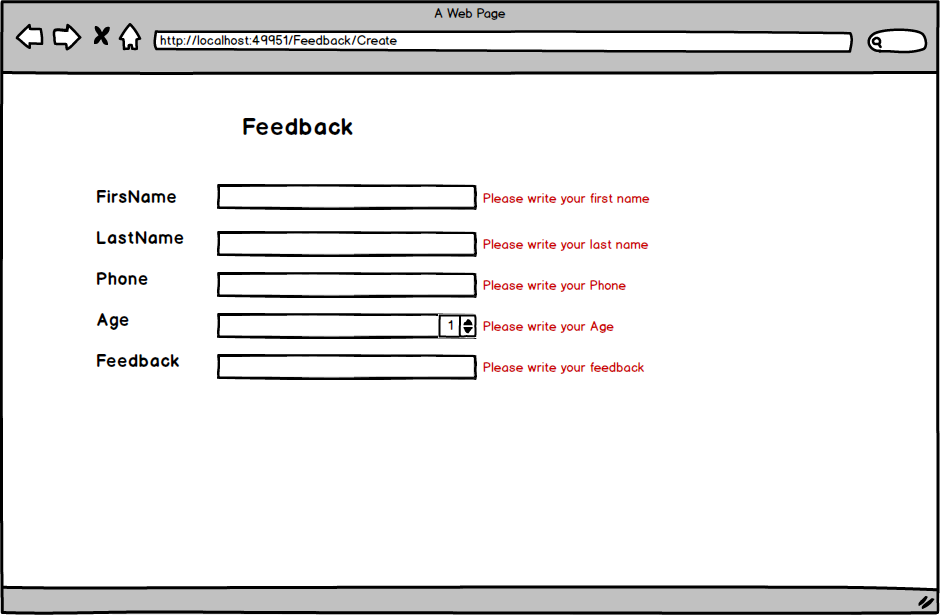
User will fill out a form to finish Feedback posting. The form will include First name, Last Name, Phone, Email, Age, Feedback, department ,After user finish fill out the form, Then user will click send message or go back to finish. By using this feature, we are able to see users’ feedback quickly and better communicate with them.

The database will look like this:

## Feed Back Database







## Job Posting Page

User will fill out a form to finish Job posting. The form will include the Job Title, Location, Email, Shift, Ft, Pay, description, Resume. After user finish fill out the form about posting page, the form will ask user are you sure you want finish? Then user will click post job or cancel to finish. This feature will benefit human resources staff and allow users to easily identify the positions they are interested in.

The database will look like this:

## Job Posting Database







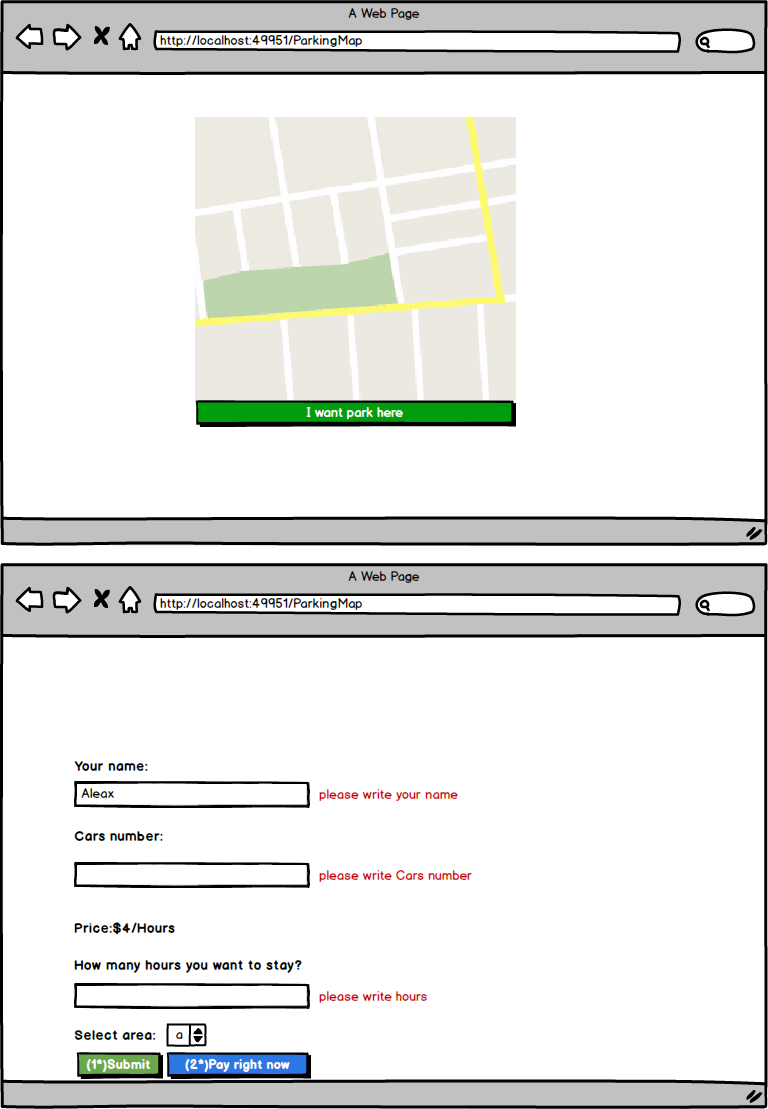
## Parking payment Page

User should login first, then go to my count page, the page will include the new location, payment derails, hours, email, phone, user can check and pay bill in details page, set payment password in setting page. People will be frustrated when they find no parking spot left at the hospital. This feature allows the user to claim their spot ahead of time and pay through online banking.

The database will look like this:

## Parking payment Database





## PHP Online exam Page

User just choose what kind of job you want to apply, then go to apply page, the page will include the Job title, Sex, Marry, Date of birthday, Educational, graduation, Phone number, introduction, user should fill out job information form first, then do the small online exam, after finish it put verification code and submit. The web manager can check the

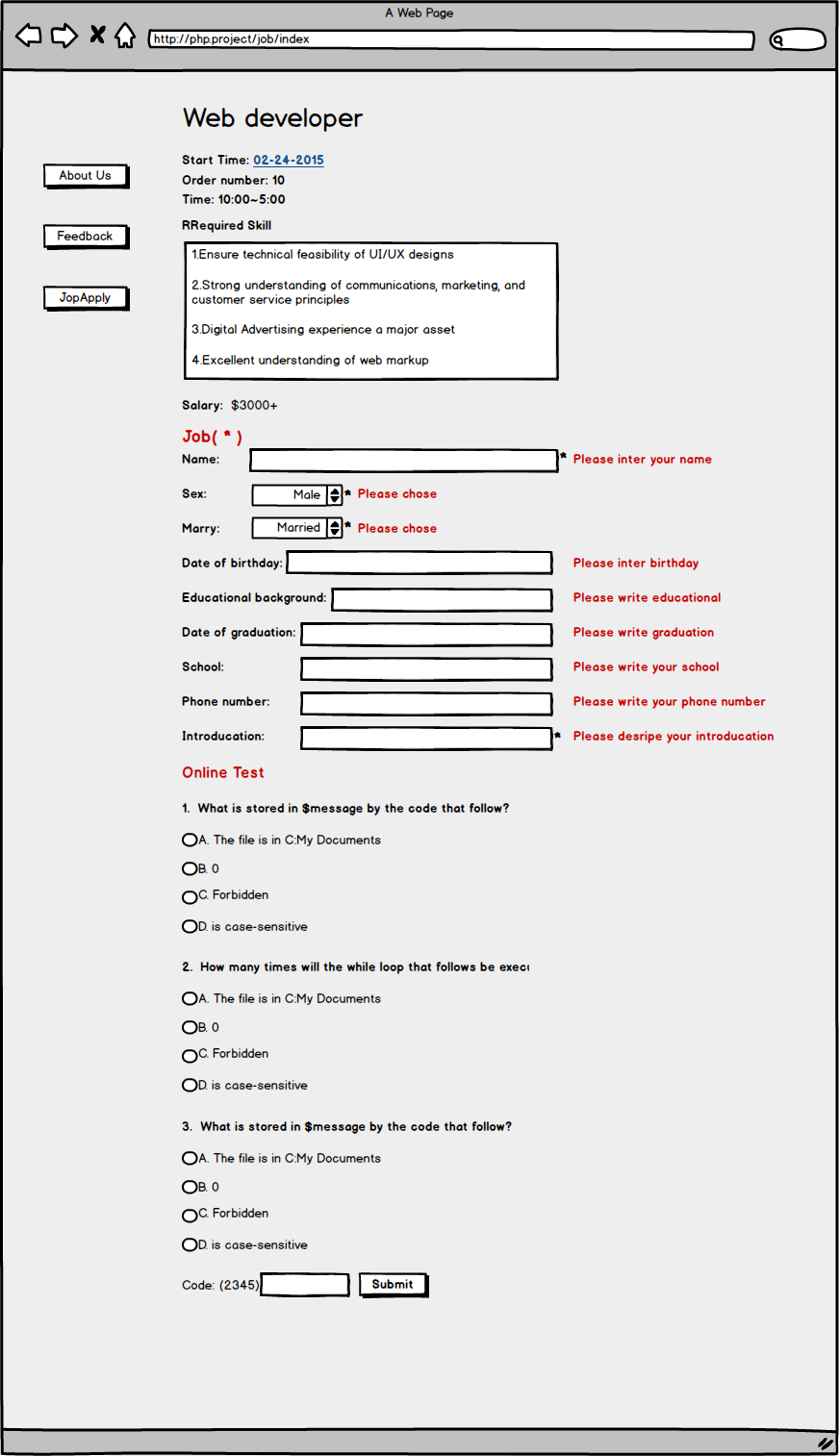
Applicant information and their text mark in the webmyadmin page then depend them mark and information to decide which one they want to hire.

The database will look like this:

## Parking payment Database







Event Registration (Community)

The only users that can register for an event are those who are logged into the system. Registering Events is a 2 step process. Users have to browse through a list of events, and when they click on an event that interests them, they are taken to a new page where more detailed information is presented along with a register for the event button. After users have read through the description, they can either select the button “*Register for Event*” or go back and browse through more events. When they click on the “*Register for Event”* button, the information is automatically stored in Eventstbl database. It stores the user’s first name, last name, date of event and their email, so that a confirmation email is sent to the user when they register for an event reminding users that they have registered. The email is sent immediately after they click on “*Register for Event*" is clicked.

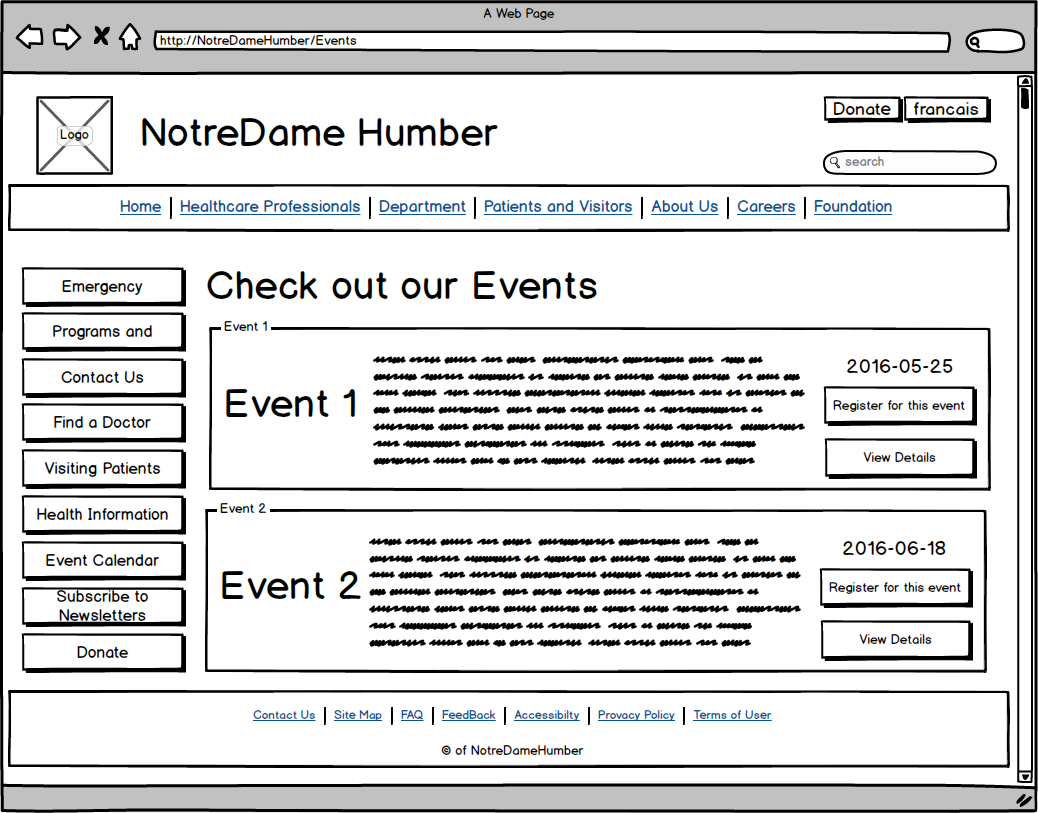
Event Registration Database (Community)



Event Registration Flow Chart (Community)

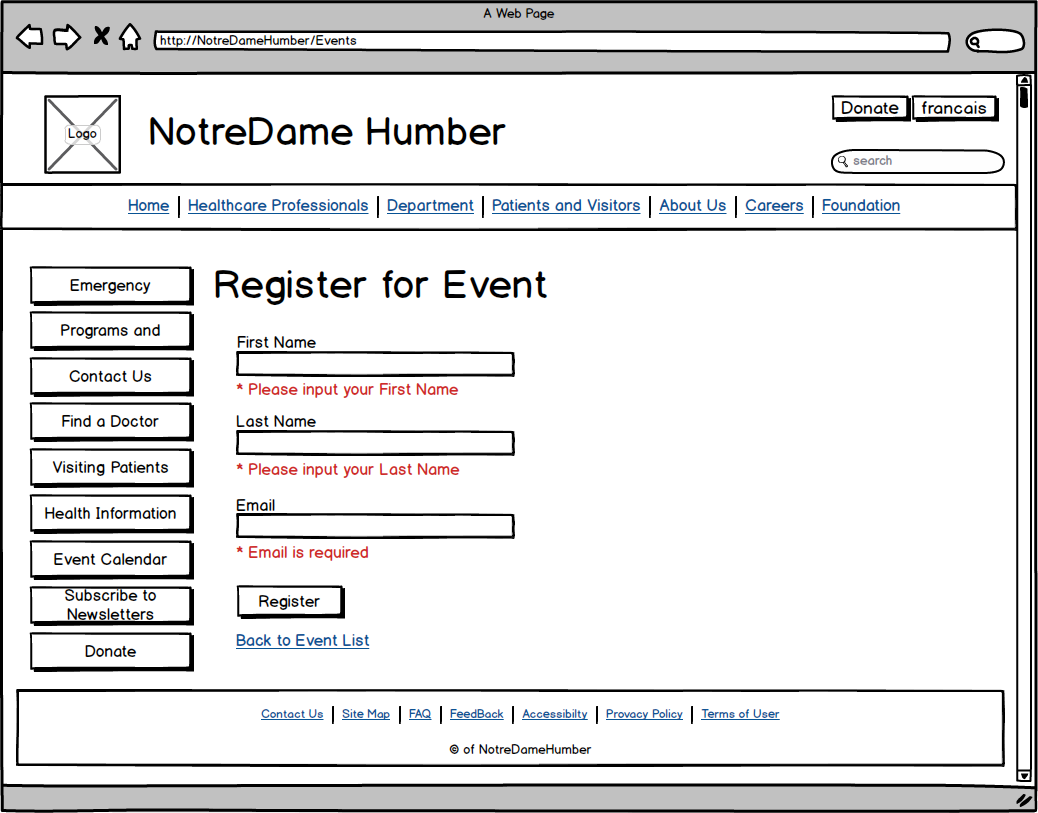


List of Events (Community)

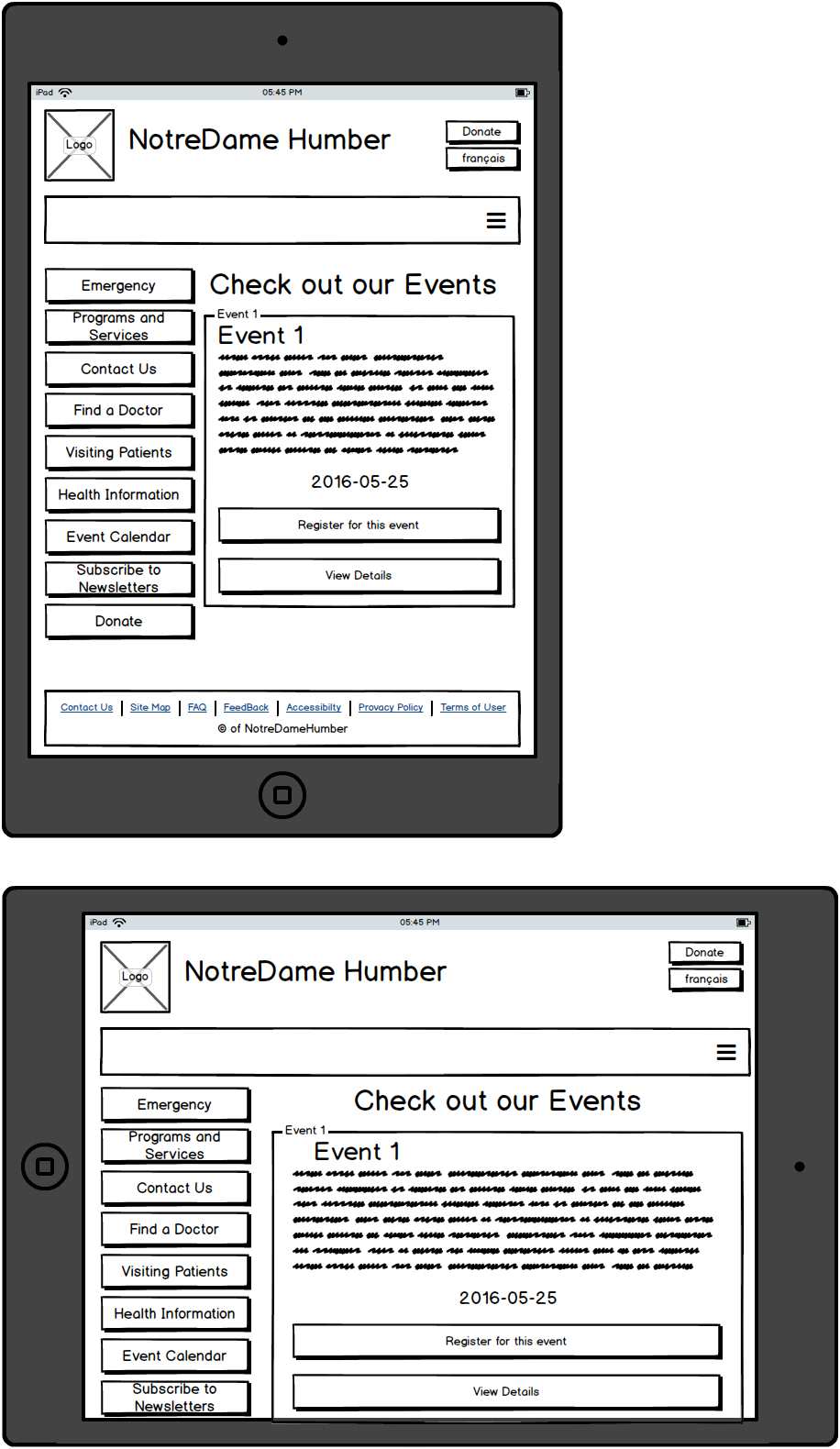


\* This page allows users to browse through events at a minimum level. Only the title of the event and the date is included, however, when the title is clicked they are taken to another page for a more detailed description. The calendar allows users to browse through events based on date. If a user would rather attend an event based on a particular date, the calendar will highlight days with events. When the users click on the calendar, a list if events for the chosen date will be displayed.

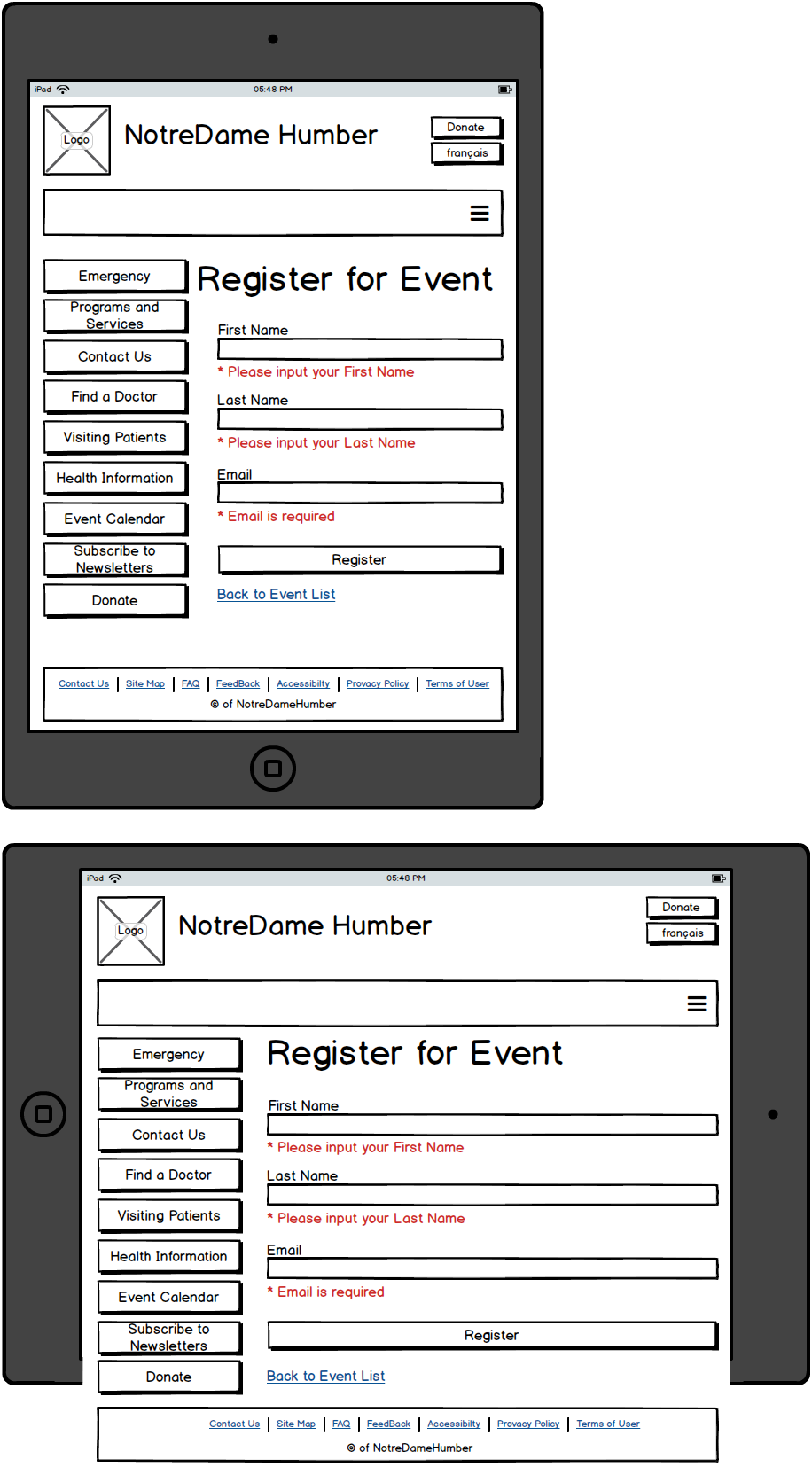
Event Registration Wireframe (Community)



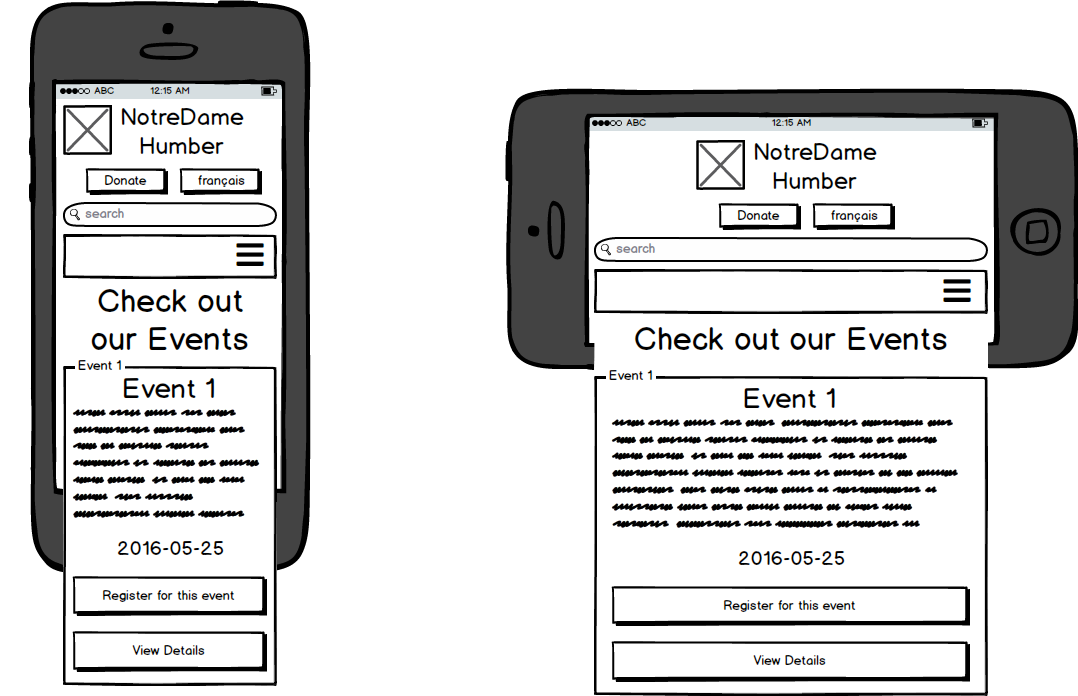
Event List Tablet Wireframe (Community)

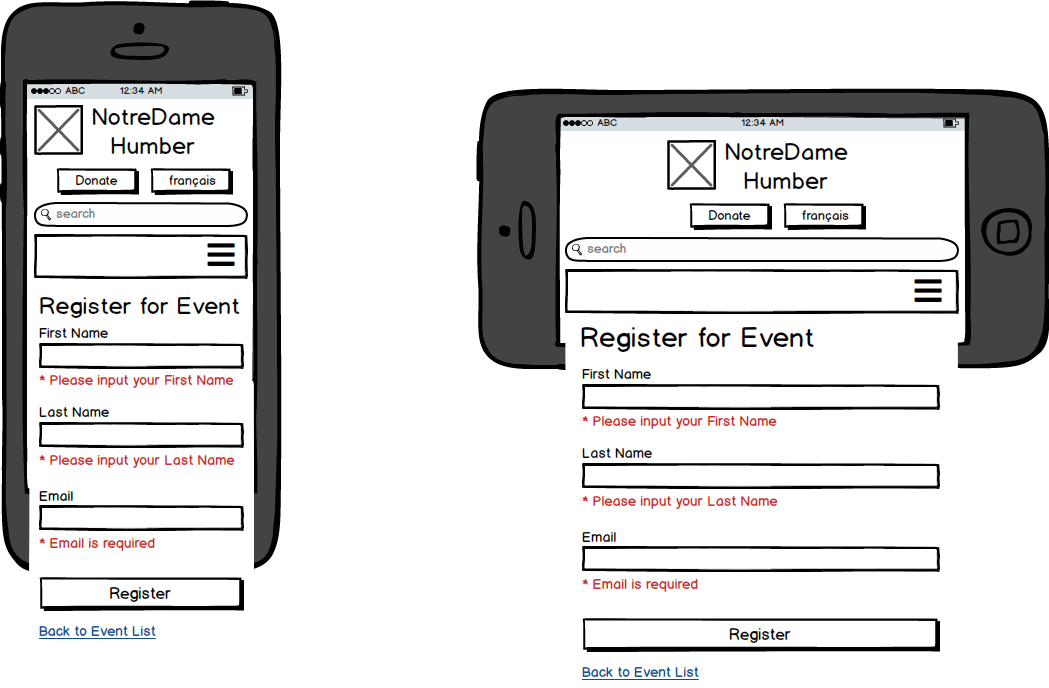


Event List and Registration Tablet Wireframe (Community)



Event List and Registration Mobile Wireframe (Community)





Event Creation (Administrator)

Administrator users can log into the hospital system and create events. They are asked to fill out a form which will ask for the Name of the Event, Address; where the event takes place, Date; when the event will happen, start time and end time, fee, Contact Email; who to contact for the event, phone; contact the event’s host via phone number, a website; if they have one, and finally, a description of the event. Only the phone and the email will not include validation, as this information is not relevant when registering for an event. Also, there is no guarantee that administrators will put in the correct phone number if it is mandatory. An email is mandatory.

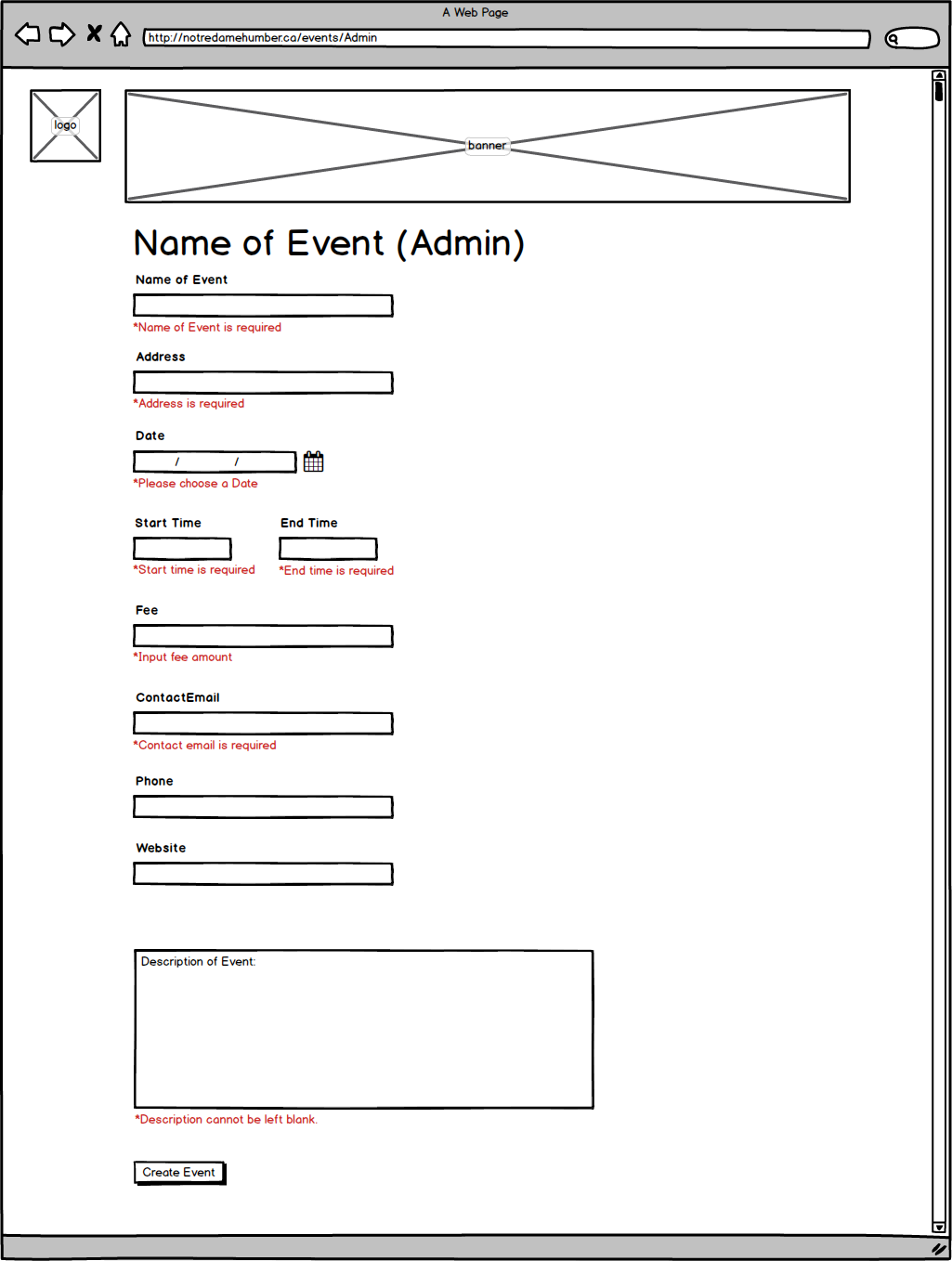
After filling out the form, they will click on the button called “*Create Event*.” The information is stored into the database, and then displayed in a list on the Events page for users to browse through.

Event Registration Database (Administrator)



Event Creation Flowchart (Administrator)



Event Registration Wireframe (Administrator)

Live Chat Narrative

A live chat allows users to interact directly with the hospital. Users can instantly ask questions instead of calling the hospital.

Users are required to input their name on a prompt box that appears before any chat takes place. If users do not input their name, the prompt box will keep appearing asking users to input name. The name value is used as an identifier for the administrator. Users do not have to input their real names.

As the conversation is proceeding, each message is being stored into the database. It will store the users name, administer name, a time stamp for users(timestamp\_u), timestamp for administrations(timestamp\_a) of the messages, and messages exchanged between the users and the administrators.

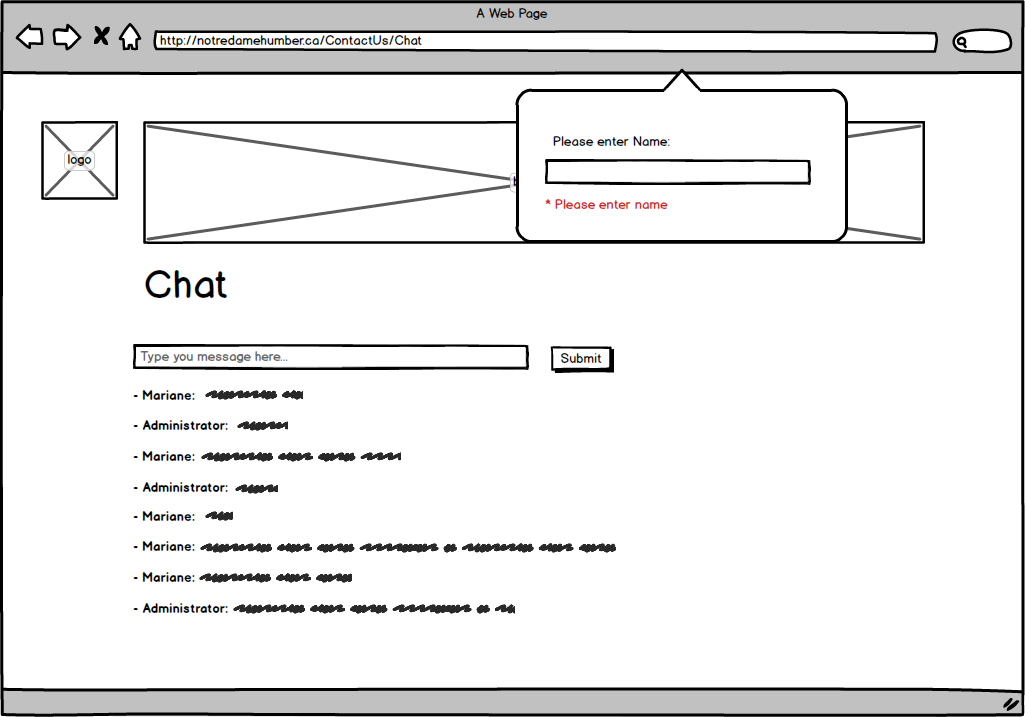
Live Chat Database



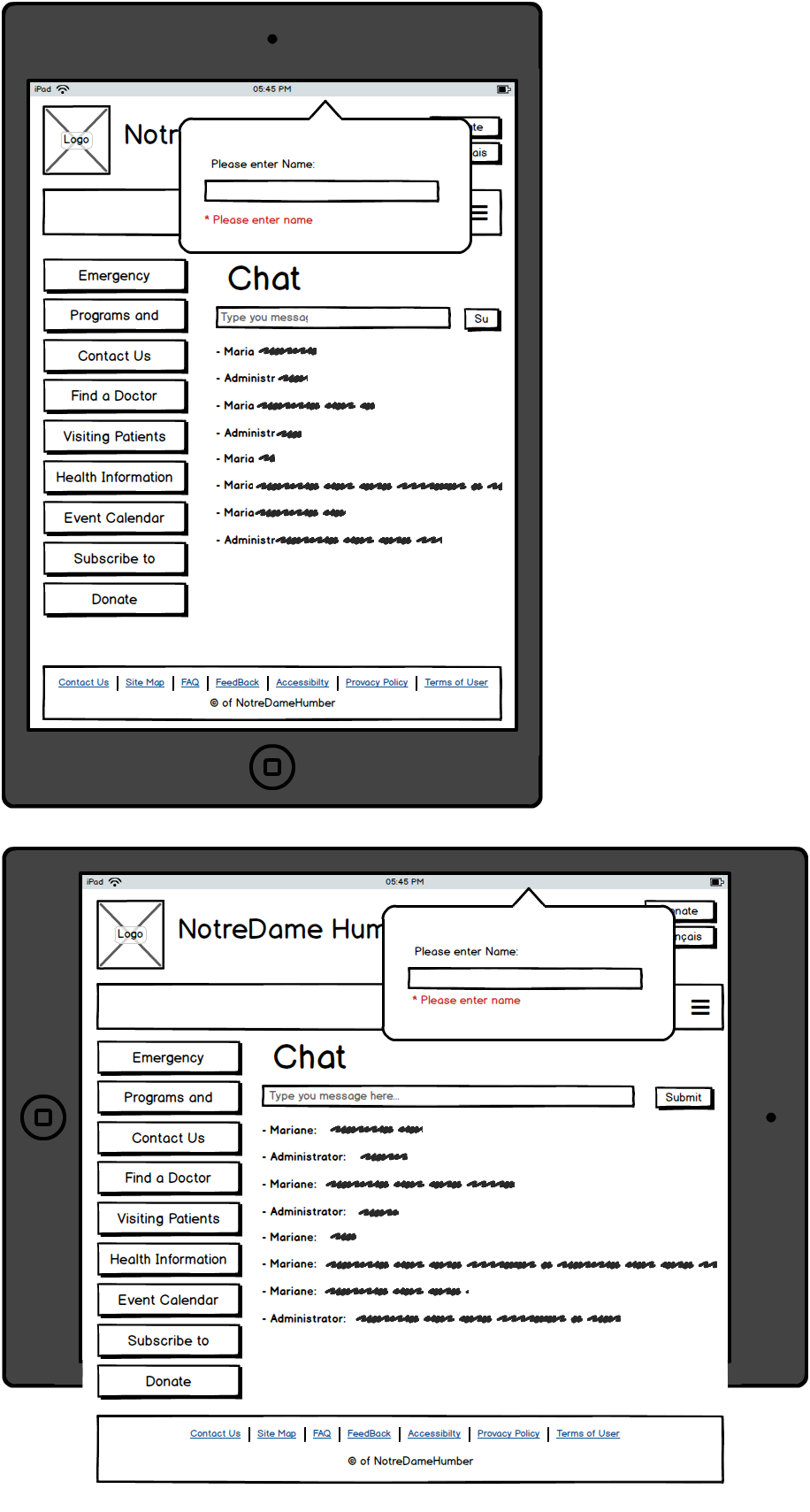
Live Chat Flow Chart



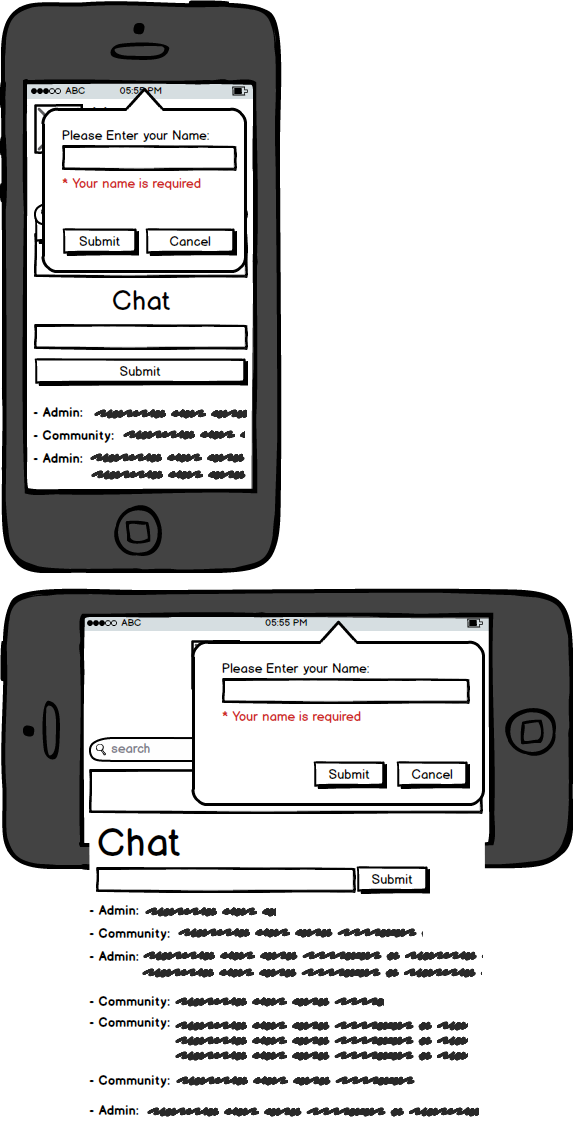
Live Chat Wireframe



Live Chat Tablet Wireframe



Live Chat Mobile Wireframe



Newsletter

Users are presented with a form to register for a newsletter. Users must provide their first name, their last name, and their email addresses, users will have to type in the captcha code before hitting the “*Join Us!* button”. If users do not provide a name, email, or type in the captcha code, the validation will present users from submitting the form. Red text will indicate which text fields need to be filled. Once the form is submitted, the data is stored into a tblNewsletter database, where the user’s first name, last name and email will be stored. Additionally, an id is created a date stamp call date will be stored in the database to indicated when the user first registered for the newsletter.

Newsletter Database



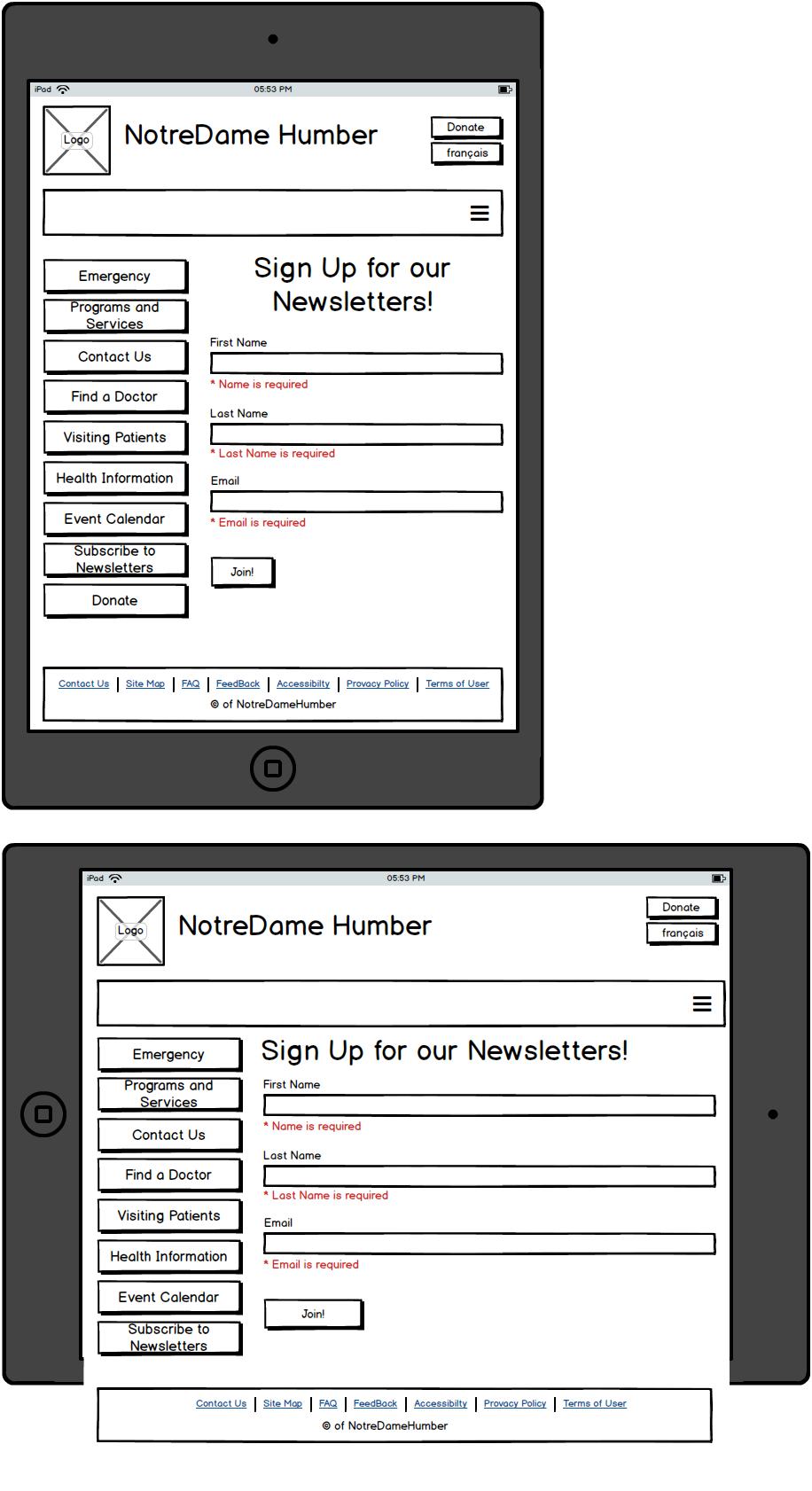
Newsletter Flow Chart



Newsletter Wireframe



Newsletter Wireframe Tablet



Newsletter Wireframe Mobile



# Development schedule (n/a)

# Maintenance plan

Up to 90 days after handover the development team is to provide unbilled maintenance for the website. The development team can be contacted by whatever means within and up to the 90 days to provide maintenance.

A copy of the website will be retained by the development team when the website launches. The version of this copy (the programming codes, database(s), HTML, CSS, and third party libraries) will be the version of the launch. In case of website crash, the website can be relaunched with this version.

Administrative permission information is to be passed onto NDH upon completion of the project. These permissions refer to the usernames, emails, and passwords used to access the content management system of the website. Administrative permissions cannot be manipulated through the content management system. Forgotten administrative permissions must be retrieved from a database administrator or through the last development team familiar with the database contents of the website. Therefore, it is important to retain a copy of this information somewhere, at NDH’s discretion, other than the database.

# Handover plan

Once website has been launched on host with completed features, administrative privileges understood, training of hospital administration complete, public testing done, and final sign off from both client and developer, the website is considered handed over.

Completed features are features that work as intended with no additional considerations needed. Administrative privileges are the credentials required to access the admin portions of the website. Training of hospital administration includes developer guidance on how to use both the public and administrative sides of the website.

Refer to the “maintenance” section of this document for post-handoff information and website changes.

**NOTREDAME HUMBER SIGN-OFF AGREEMENT**

This agreement is effective from 27 of April 2016

Document Purpose

This agreement acknowledges that the client has reviewed the progress and services by Group 5 as per the obligation outlined in the requirements documentation used during the course of the meeting.

Agreement Parties

Between NotreDame Humber Hospital Team

And Group 5

Terms

The client is satisfied that all agreed work has been produced to the satisfactory standard and that all contractual obligations have been met to the client’s expectation.

I have read and understood the above agreement:

Signed by NotreDame Humber:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Bernie Monette Date

**NOTREDAME HUMBER SIGN-OFF AGREEMENT**

This agreement is effective from 27 of April 2016

Document Purpose

This agreement acknowledges that the client has reviewed the progress and services by Group 5 as per the obligation outlined in the requirements documentation used during the course of the meeting.

Agreement Parties

Between NotreDame Humber Hospital Team

And Group 5

Terms

The client is satisfied that all agreed work has been produced to the satisfactory standard and that all contractual obligations have been met to the client’s expectation.

I have read and understood the above agreement:

Signed by NotreDame Humber:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Sean Doyle Date

**NOTREDAME HUMBER SIGN-OFF AGREEMENT**

This agreement is effective from 27 of April 2016

Document Purpose

This agreement acknowledges that the client has reviewed the progress and services by Group 5 as per the obligation outlined in the requirements documentation used during the course of the meeting.

Agreement Parties

Between NotreDame Humber Hospital Team

And Group 5

Terms

The client is satisfied that all agreed work has been produced to the satisfactory standard and that all contractual obligations have been met to the client’s expectation.

I have read and understood the above agreement:

Signed by NotreDame Humber:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Yasemin Fanaeian Date