CPSC 471

Group 79 Final Project

Group members: Heidi Toews (UCID 30094995), Alisha Lalani (UCID: 30123098),

Harsweet Singh (UCID 30117846), Kusumpreet Kaur Heer (UCID: 30114618)

HIPO diagram:

Shop:

* Browse items
* Add item to cart
* View items in cart
* Remove item from cart
* Checkout
  + Add to billing history

Appointment:

* View available dates
* Add appointment
  + Choose pet
  + Choose type of appointment
  + Add to billing history
* Cancel appointment
* Reschedule appointment (time & date)

Client:

* View pets
  + View medical history
* View upcoming appointments
* Edit client
  + update/save info
* View billing history

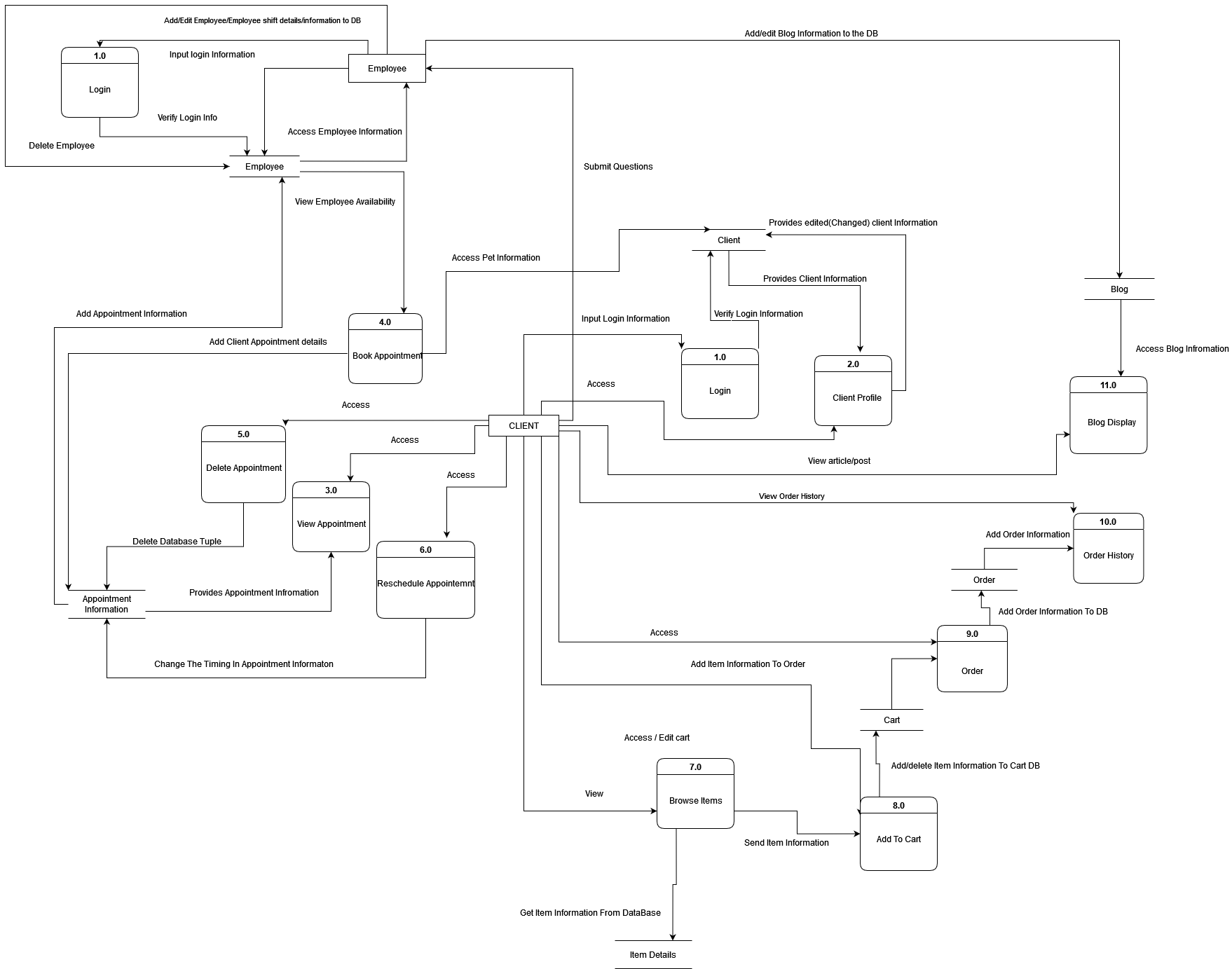
Employee

* Add shift
* Remove shift
* Edit employee
  + enter/edit info
  + update/save info
* Vet
  + View medical history
  + Edit medical history
    - Add condition/treatment
  + View appointments
* Admin
  + Book appointment
  + Send appointment reminder

Blog

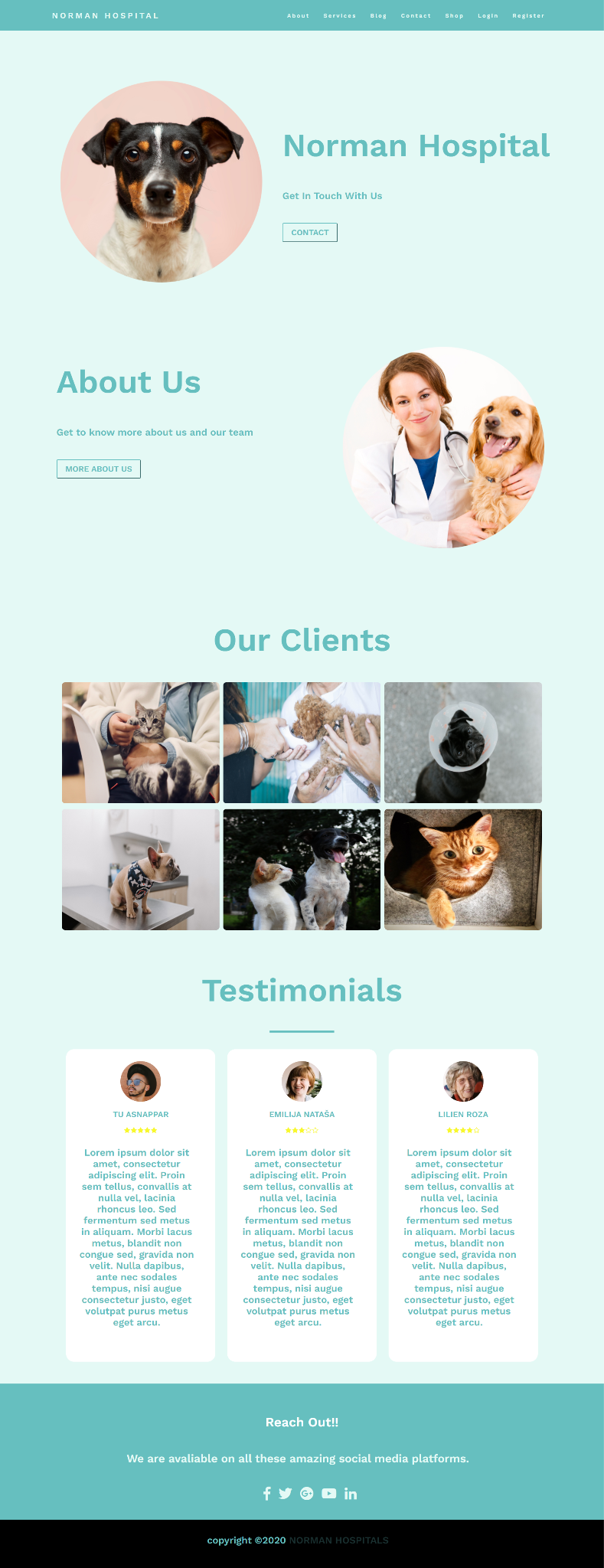
* View article
* View post
* Submit question

DFD diagram:



Dummy website:

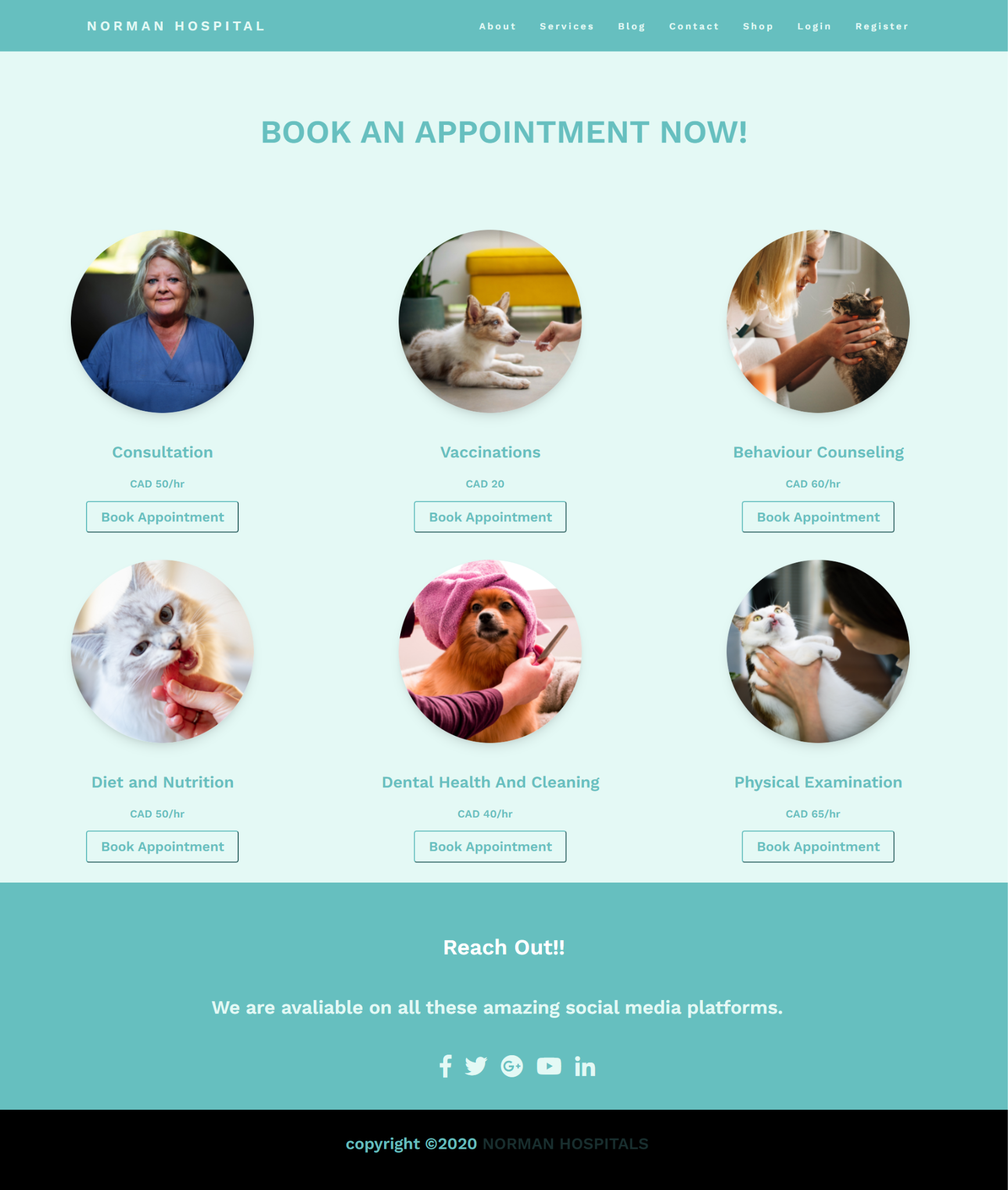
Home:



About:



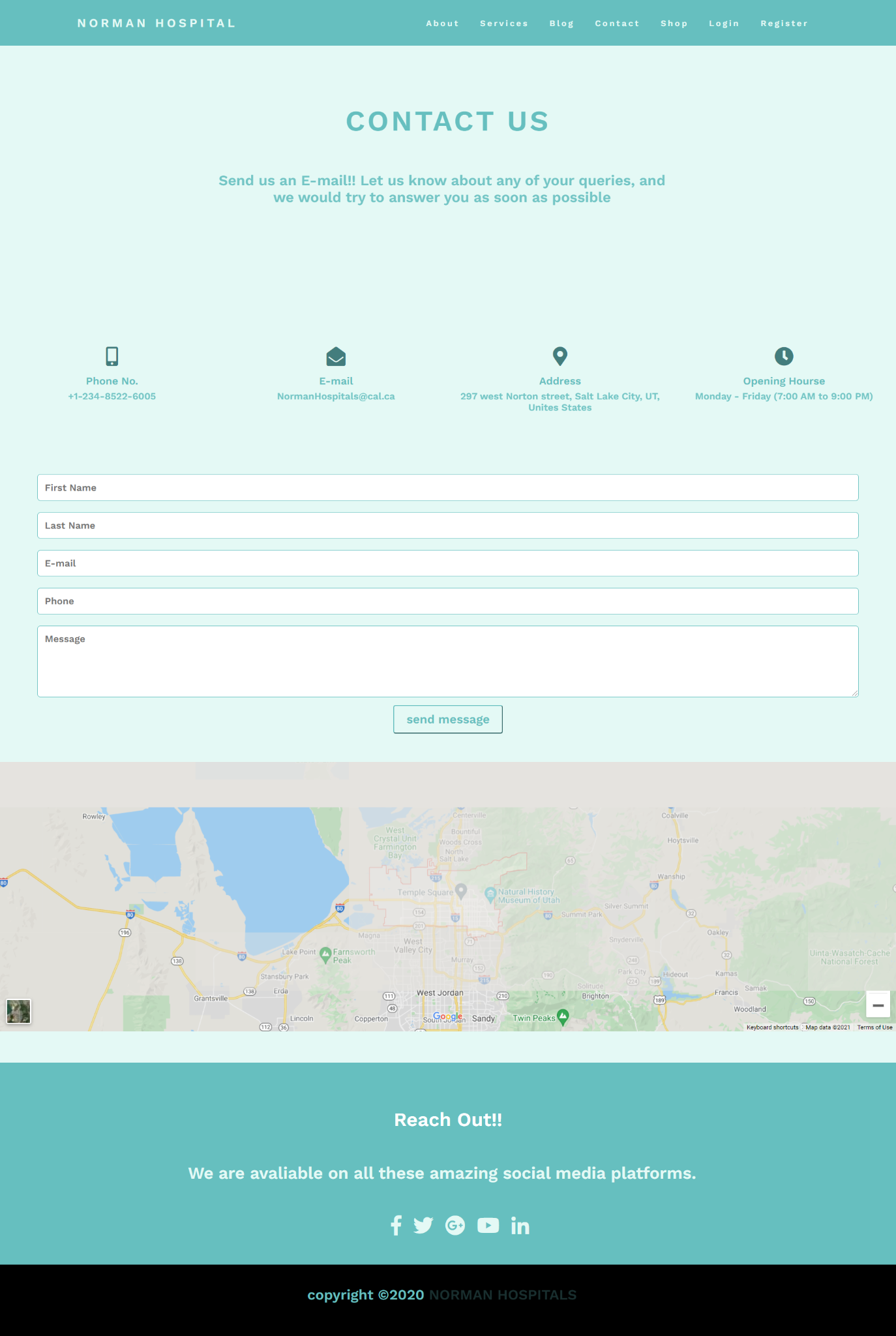
Services:



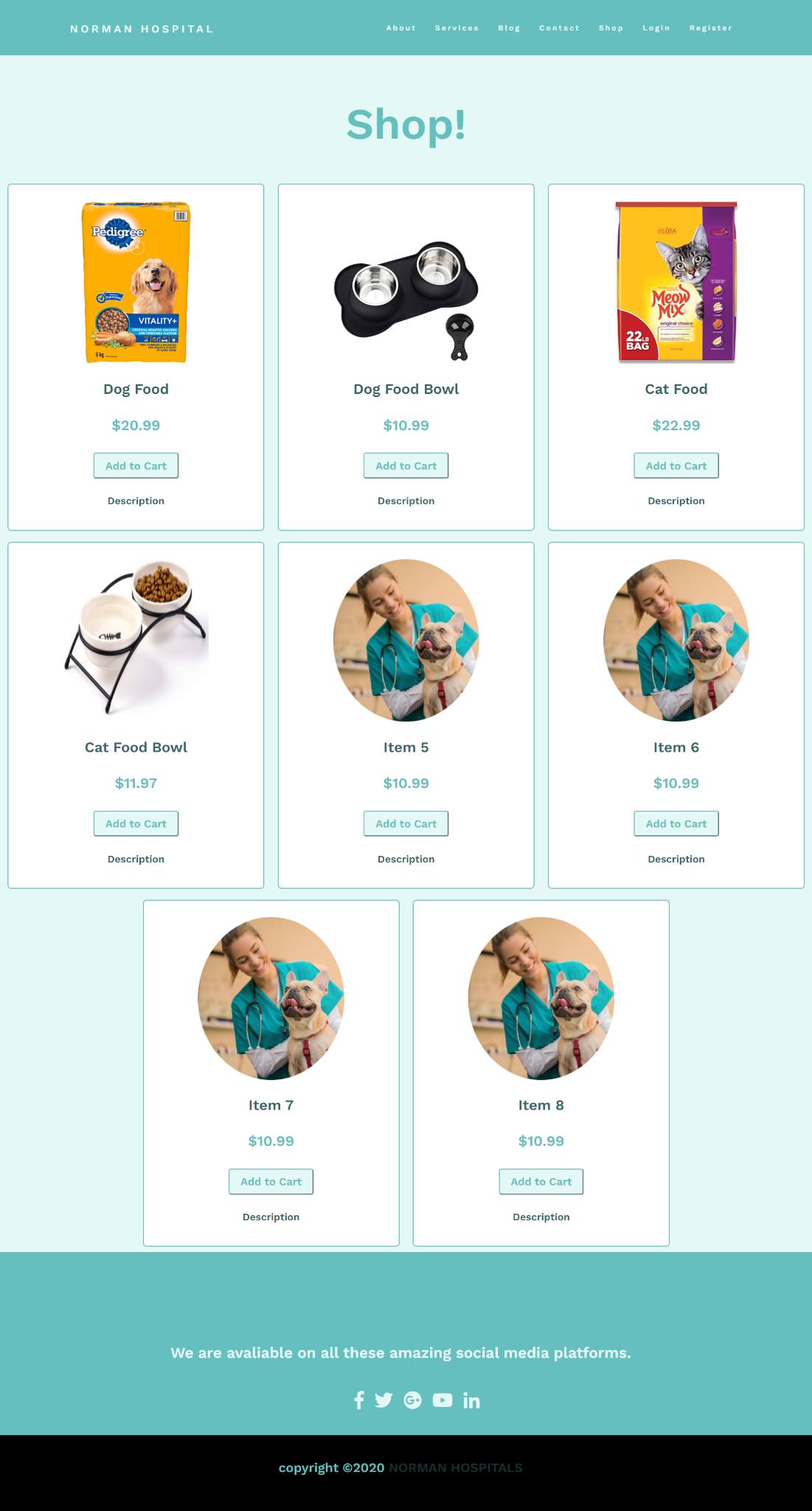
Blog:



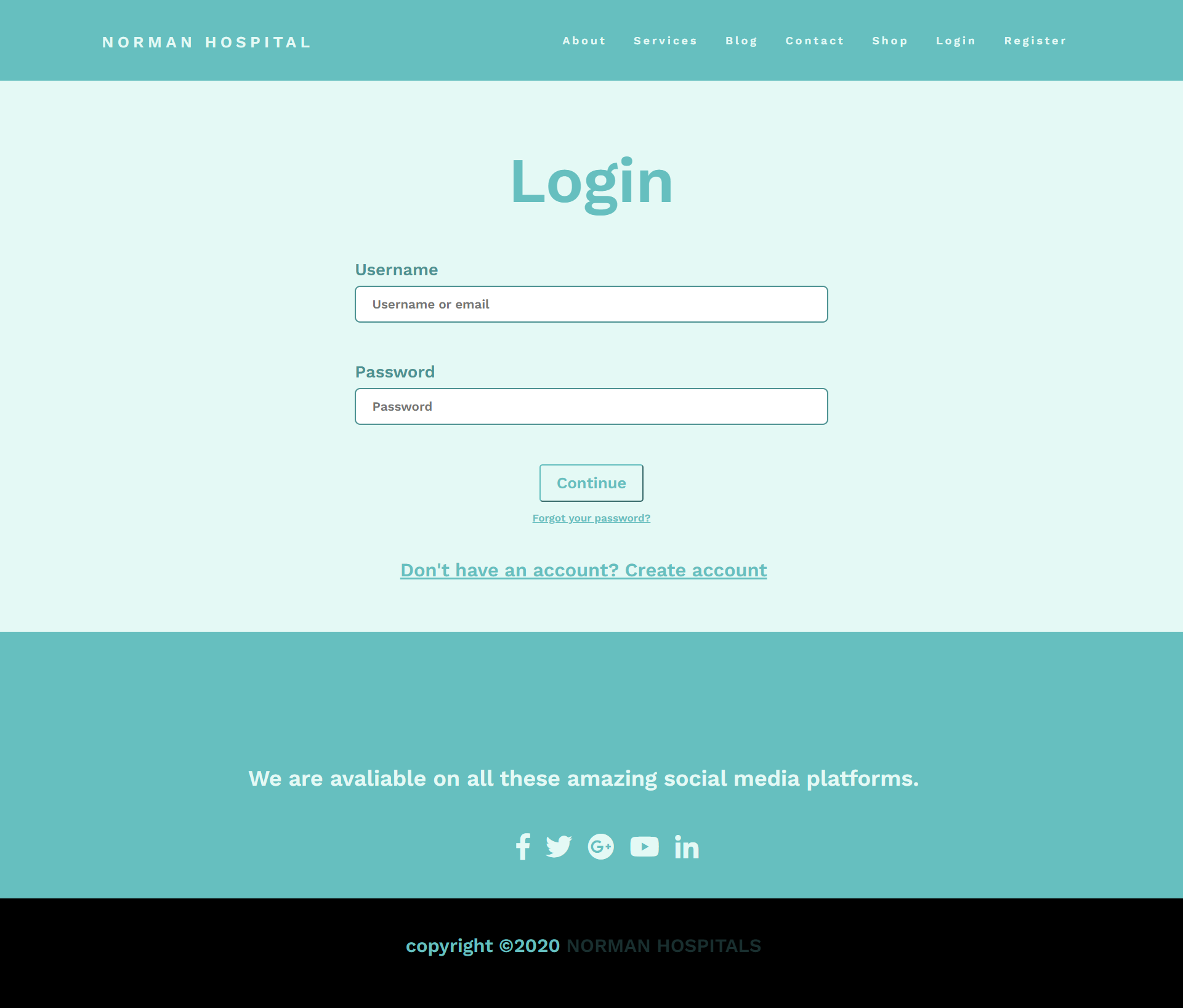
Contact:



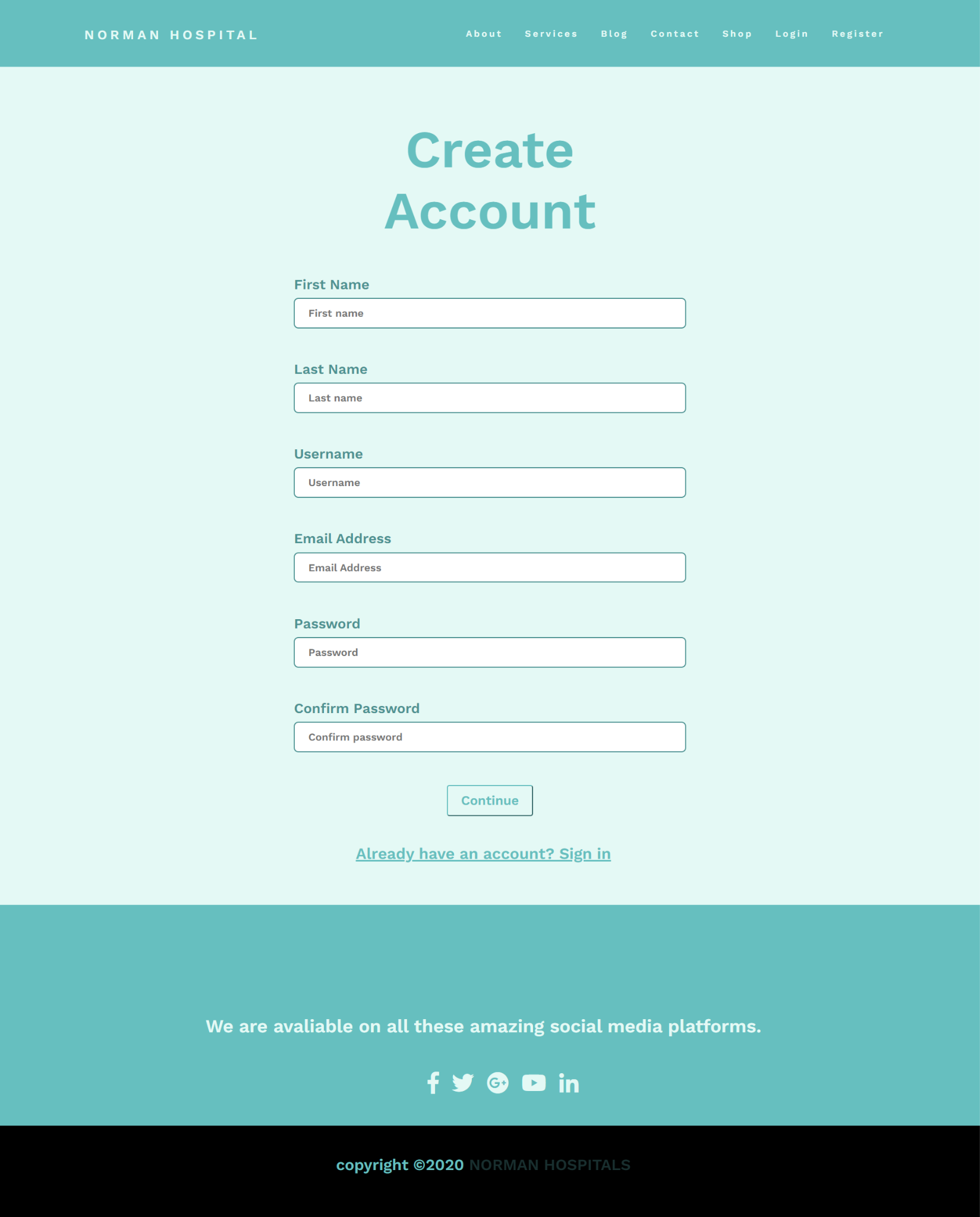
Shop:



Login:



Register:



**API Endpoints:**

**Blog Question Endpoints:**

Endpoint 1:

Description: Post a question for the blog page.

URL: <http://localhost/api/blog/question>

Parameters: client\_ID

Method: POST

Input: [{

“FName”: string

“LName”: string

“question”: string

“Username”: string

}]

Output: ques\_ID

Endpoint 2:

Description: Get question for the blog page.

URL: URL: <http://localhost/api/blog/question>

Parameters: ques\_ID

Method: GET

Output: [{

“FName”: string

“LName”: string

“question”: string

“Username”: string

}]

Endpoint 3:

Description: Get all questions for the blog page.

URL: <http://localhost/api/blog/question>

Method: GET

Output: [{

“FName”: string

“LName”: string

"question": string

“Username”: string

}]

Endpoint 4:

Description: Delete a question from the blog page.

URL: <http://localhost/api/blog/question>

Method: DELETE

Parameters: ques\_ID

Endpoint 5:

Description: Edit a question for the blog page.

URL: <http://localhost/api/blog/question>

Method: PUT

Input: [{

“FName”: string

“LName”: string

“question”: string

“Username”: string

}]

**Blog Endpoints**

Endpoint 1:

Description: Get post for the blog page.

URL: [http://localhost/api/blog](http://localhost/api/blog/question)

Parameters: article\_ID

Method: GET

Output: [{

"post": string

“Vet\_author”: string

}]

Endpoint 2:

Description: Add post for the blog page.

URL: [http://localhost/api/blog](http://localhost/api/blog/question)

Method: Post

Input: [{

"post": string

“Vet\_author”: string

}]

Endpoint 4:

Description: Delete post for the blog page.

URL: [http://localhost/api/blog](http://localhost/api/blog/question)

Method: Delete

Parameters: article\_ID

Endpoint 5:

Description: Edit post for the blog page.

URL: [http://localhost/api/blog](http://localhost/api/blog/question)

Method: Put

Input: [{

"post": string

“Vet\_author”: string

}]

**Client Endpoints:**

Endpoint 1:

Description: Update client Information.

URL: <http://localhost/api/client>

Method: Put

Parameters: ID

Input: [{

“Username”: String

“Password:” String

“Name”: String

“Address”: String

“Client\_ID”: int

“Email”: String

“Phone”: i**nt ?? or string**

}]

Endpoint 3:

Description: Get a client’s Information.

URL: <http://localhost/api/client>

Method: Get

Input: ID

Output: [{

“Name”: String

“Address”: String

“Client\_ID”: int

“Email”: String

“Phone”: i**nt ?? or string**

}]

Endpoint 4:

Description: Get all clients’ Information.

URL: <http://localhost/api/client>

Method: Get

Output:

Endpoint 4:

Description: Get username

URL: <http://localhost/api/client>

Method: Get

Input: ID

Output: [{

“Username”: String

}]

Endpoint 5:

Description: Get Password

URL: <http://localhost/api/client>

Method: Get

Input: ID

Output: {[

“Pw”: String

}]

Endpoint 6:

Description: Set user information

URL: <http://localhost/api/client>

Method: POST

Input: {[

“Username”: String

“Password:” String

“Name”: String

“Address”: String

“Client\_ID”: int

“Email”: String

“Phone”: i**nt ?? or string**

]}

Appointment Endpoints:

Endpoint 1:

Description: Get all available appointment times

URL: <http://localhost/api/appointment>

Method: Get

Parameters: job = “available“

Output: {[

“Date”: String

“Time”: int

]}

Endpoint 2:

Description: Get all available appointment times on a specific day (by pressing a date on the calendar)

URL: <http://localhost/api/appointment>

Method: Get

Parameters: Date, job = “available”

Output: {[

“Date”: String

“Time”: int

}]

Endpoint 3:

Description: Update an available appointment time on a specific day

URL: <http://localhost/api/appointment>

Method: PUT

Parameters: Date, Time, job = “available”

Input: {[

“Date”: String

“Time”: int

}]

Endpoint 4:

Description: Delete an available appointment time on a specific day

URL: <http://localhost/api/appointment>

Method: DELETE

Parameters: Date, Time, job = “available”

Endpoint 5

Description: Get all scheduled appointment times

URL: <http://localhost/api/appointment>

Method: Get

Parameters: job = “scheduled”

Output: {[

“Date”: String

“Time”: int

}]

Endpoint 4:

Description: Get a specific appointment

URL: <http://localhost/api/appointment>

Method: Get

Parameters: app\_ID, job = “scheduled”

Output: {[

“Date”: String

“Time”: int

“Vet”: String

}]

Endpoint 3:

Description: Schedule an appointment - Note: when scheduling an appointment, that same time slot should be removed from the available appointment times … BUt this needs to be from all available appointment time slots

URL: <http://localhost/api/appointment>

Method: Post

Parameters: job = “scheduled”

Input: {[

“Date”: String

“Time”: int

}]

Endpoint 4:

URL: <http://localhost/api/appointment>

Description: Delete a scheduled appointment

Parameters: app\_ID, job = “scheduled”

Endpoint 5:

URL: <http://localhost/api/appointment>

Description: Update a scheduled appointment (Reschedule it) - I don’t think this should be allowed from the client’s end

Method: Put

Parameters: app\_ID, job = “scheduled”

Input: {[

“Date”: String

“Time”: int

}]

**Item Endpoints:**

Endpoint 1:

Description: Get all items

URL: <http://localhost/api/item>

Method: Get

Output: {[

“item\_NO”: int

“pet\_Category”: String

“inventory”: int

“name”: String

“seling\_Price”: int

]}

Endpoint 2:

Description: Get a specific item

URL: <http://localhost/api/item>

Method: Get

Parameters: item\_NO

Output: {[

“item\_NO”: int

“pet\_Category”: String

“inventory”: int

“name”: String

“seling\_Price”: int

]}

Endpoint 3:

Description: Delete an item

URL: <http://localhost/api/item>

Parameters: item\_NO

Endpoint 4:

Description: Update an item

URL: <http://localhost/api/item>

Method: PUT

Parameters: item\_NO

Input: {[

“item\_NO”: int

“pet\_Category”: String

“inventory”: int

“name”: String

“seling\_Price”: int

]}

Endpoint 5:

Description: Add a new item

URL: <http://localhost/api/item>

Method: POST

Input: {[

“item\_NO”: int

“pet\_Category”: String

“inventory”: int

“name”: String

“seling\_Price”: int

]}

**Cart (Purchase) Endpoints: - there shouldn’t be an update here right?**

Endpoint 1:

Description: Add a given item to cart (for a client)

URL: <http://localhost/api/client/cart>

Method: POST

Parameters: item\_NO, client\_ID

Output: {[

“item\_NO”: int

“pet\_Category”: String

“inventory”: int

“name”: String

“seling\_Price”: int

]}

Endpoint 3:

Description: Delete a given item from cart (for a client)

URL: <http://localhost/api/client/cart>

Method: DELETE

Parameters: item\_NO, client\_ID

Endpoint 4:

Description: Checkout (Get all items in the cart)

URL: <http://localhost/api/client/cart>

Method: Get

Parameters: client\_ID

Output: {[

“item\_NO”: int

“pet\_Category”: String

“inventory”: int

“name”: String

“seling\_Price”: int

]}

Endpoint 5:

Description: Get a specific item in cart

URL: <http://localhost/api/client/cart>

Parameters: item\_NO, client\_ID

Output: {[

“item\_NO”: int

“pet\_Category”: String

“inventory”: int

“name”: String

“seling\_Price”: int

]}

**Billing History Endpoints**

Endpoint 1: View all billing history purchases for a specific client

URL: <http://localhost/api/client/BillingHistory>

Method: Get

Parameters: client\_ID

Output: {[

“Date”: String

“service\_or\_product\_name”: String

“cost”: int

]}

Endpoint 2: View a specific purchase for a specific client

URL: <http://localhost/api/client/BillingHistory>

Method: Get

Parameters: bill\_ID, client\_ID

Output: {[

“Date”: String

“service\_or\_product\_name”: String

“cost”: int

]}

Endpoint 3: Add a purchase to billing history for a specific client

URL: <http://localhost/api/client/BillingHistory>

Parameters: client\_ID

Method: Post

Input: {[

“Date”: String

“service\_or\_product\_name”: String

“cost”: int

]}

Endpoint 3:

Description: Delete a purchase

URL: <http://localhost/api/client/BillingHistory>

Method: Delete

Parameters: item\_NO

Endpoint 4: Edit a purchase (update)

URL: <http://localhost/api/client/BillingHistory>

Method: Put

Input: {[

“Date”: String

“service\_or\_product\_name”: String

“cost”: int

]}

Employee Endpoints:

Endpoint 1:

Description: Set Employee Information

URL: <http://localhost/api/employee>

Method: Post

Input: {[

“Name”: String

“Specialty”: String

“Address”: String

“Phone #”: String

}]

Endpoint 2:

Description: Get all Employee Information

URL: <http://localhost/api/employee>

Method: Get

Output: {[

“Name”: String

“Specialty”: String

“Address”: String

“Phone #”: String

}]

Endpoint 3:

Description: Get a specific Employee’s information

URL: <http://localhost/api/employee>

Method: Get

Input: SIN

Output: {[

“Name”: String

“Specialty”: String

“Address”: String

“Phone #”: String

}]

Endpoint 4:

Description: Update a specific employee’s information

URL: <http://localhost/api/employee>

Method: Put

Parameters: SIN

Input: {[

“Name”: String

“Specialty”: String

“Address”: String

“Phone #”: String

}]

Endpoint 5: Remove an employee

URL: <http://localhost/api/employee>

Method: Delete

Parameters: SIN

**Shift Endpoints:**

Endpoint 1: Add a shift for an employee

Method: Post

Parameters: SIN

Input: {[

“Start\_time”: int

“End\_time”: int

“Date”: String

]}

Endpoint 2: Get a specific shift

Method: Get

Parameters: SIN, **shift\_ID?**

Output: {[

“Start\_time”: int

“End\_time”: int

“Date”: String

]}

Endpoint 3: View all shifts for an employee

Method: Get

Parameters: SIN

Output: {[

“Start\_time”: int

“End\_time”: int

“Date”: String

]}

Endpoint 4: Update a shift for an employee

Method: Put

Parameters: SIN

Input: {[

“Start\_time”: int

“End\_time”: int

“Date”: String

]}

**Pet Endpoints:**

Endpoint 1:

Description: Add Pet (Here the information is different from the pet’s medical history) for a client

URL: <http://localhost/api/client/pet>

Parameters: client\_ID

Method: Post

Input: {[

“Name”: String

“Age”: String

“Weight”: int

“Color”: String

“Sex”: String

“Species”: String

“Pet category”: String

]}

Endpoint 2:

Description: Get a single pet’s Information

URL: <http://localhost/api/client/pet>

Method: Get

Parameters: pet\_ID, client\_ID

Output: {[

“Name”: String

“Age”: String

“Weight”: int

“Color”: String

“Sex”: String

“Species”: String

“Pet category”: String

]}

Endpoint 3:

Description: Get all client’s pets’ information

URL: <http://localhost/api/client/pet>

Method: Get

Parameters: client\_ID

Output: {[

“Name”: String

“Age”: String

“Weight”: int

“Color”: String

“Sex”: String

“Species”: String

“Pet category”: String

]}

Endpoint 3:

Description: Update Pet Information

URL: <http://localhost/api/client/pet>

Method: Put   
Parameters: pet\_ID, client\_ID

Input: {[

“Name”: String

“Age”: String

“Weight”: int

“Color”: String

“Sex”: String

“Species”: String

“Pet category”: String

]}

**Pet Medical History Endpoints**

Endpoint 1:

Description: Set Pet Medical History

URL: <http://localhost/api/client/pet/medHistory>

Method: Post

Parameters: pet\_ID, client\_ID

Input: {[ Description }]

Endpoint 2:

Description: Get a Pet’s Medical History records

URL: <http://localhost/api/client/pet/medHistory>

Method: Get

Parameters: pet\_ID, client\_ID

Output:{[

“Description”: String

}]

Endpoint 3:

Description: Get all clients’ pets’ medical history records

URL: <http://localhost/api/client/pet/medHistory>

Parameters: client\_ID

Output: {[

“Pet\_name”: String

“Description”: String

}]

Description: Update Pet’s Medical History

URL: <http://localhost/api/client/pet/medHistory>

Method: Put

Parameters: pet\_ID, client\_ID

Input: {[

“Description”: String

}]

Endpoint 4:

Description: Delete a Pet’s Medical history

URL: <http://localhost/api/client/pet/medHistory>

Parameters: pet\_ID, client\_ID

Note: no add here, since 1 pet has 1 medical history

**Rewards Card Endpoints**

Endpoint 1:

Description: Get all rewards cards

URL: http://locahost/api/rewardsc

Method: Get

Output: {[

“Points”: int

]}

Endpoint 2:

Description: Get a specific rewards card

URL: http://locahost/api/rewardsc

Method: Get

Parameters: client\_ID

Output: {[

“Points”: int

]}

Endpoint 3:

Description: Add a rewards card

URL: http://locahost/api/rewardsc

Method: Post

Parameters: client\_ID

Input: {[

“Points”: int

]}

Endpoint 4:

Description: Update a rewards card

URL: http://locahost/api/rewardsc

Method: Put

Parameters: client\_ID

Input: {[

“Points”: int

]}

Endpoint 5:

Description: Delete a rewards card

URL: http://locahost/api/rewardsc

Method: Delete

Parameters: client\_ID

Input: {[

“Points”: int

]}