CIS4339 Fall 2021 - Final Project

Synopsis

For your final project you will be building a web application following a standard full-stack architecture using the MEVN stack (MongoDB, Express, VueJS, Node). The project will be executed as a group project. You will be assigned to a team by the instructor. The goal is to develop an application that will support the management of activities in a non-profit organization in the Houston area. We have partnered with 3 organizations and your team will have the choice to implement one of the following 3 options.

Option A: Community Family Centers

Community Family Centers/Centros Familiares de la Comunidad (CFC) (https://www.communityfamilycenters.org/index.php) is a multi-service non-profit organization that has offered a wide range of bilingual services to families within Houston's impoverished Greater East End for over 45 years. Their goal and mission is to equip families with the tools they need to become self-sufficient members of the community. Through four core programs: Adult Education, Family Support Services, Early Childhood - Los Niños Early Childhood Montessori Program and Youth Services they strive to help families break the cycle of poverty and illiteracy so that they can move toward self-sufficiency and economic advancement.

The web application, will help to build a basic data platform allowing CFC to implement a more efficient and data driven way to provide their services.

Your first task is the implementation of an *Intake Form* page for the collection of information for new clients tha can be used by the CFC Community Health Workers. Details for the intake form data can be found in Appendix A. The intake form data needs to be stored so that it can viewed and edited at a later point, but in such a way that the history of the edits is also tracked.

Next, there needs to be a page that allows a CFC Community Health Worker to record activities with clients. Those activities evolve around the 4 core programs and the user should be able to easily be able to "attach" the activities to a client, select to which of the 4 core services the activity belongs and record short notes, date/time as well as time spent on the activity. The activities must also be linked to a Community Health Worker acting as the relationship manager. It will also require the relationship manager to be able to come back at a later date to choose whether the client used the services (Yes/No) and whether the underlying need was addressed (not at all, partially, fully).

Finally, there needs to be a page that can be used by a supervisor to view the activities by date/time and/or client and/or Community Health Worker (table format). In addition the page must show at least one summary graph related to activities.

Option B: Bread of Life

Bread of Life (https://breadoflifeinc.org/) began in 1992 by serving 500 hot meals a day to homeless individuals in midtown Houston. With the help of a close partnership with Beyonce and Tina Knowles-Lawson, they have coordinated numerous anti-hunger initiatives and provided emergency support during natural disasters, in addition to their outreach to the homeless. They need data tracking and reporting to support their activities in homeless outreach, anti-hunger campaigns, and health outcomes in the communities served.

The web application will help to build a basic data platform for a more efficient and data driven approach to their services.

They need a page that provides intake information on individuals served at certain distribution events/services. Details can be found in Appendix B. IMPORTANT: There needs to be a page where a project manager can update the list of events/services offered. The list of possible distribution events/services has to come from entries in the DB and is not fixed for the intake forms. Events should have a type, location and date stored for them. The form should only list the 3 most current events/services.

Next, there needs to be a page that allows the project managers to know how many times and in what ways each individual has accessed their services. There should be a table that shows the individual's history and gives the total.

Finally, there needs to be the page that a supervisor can use to see the total activities by event (such as a single day food distribution) or by place of residence (zip code) of the people served. There should also be at least one summary graph to show individual use patterns or patterns across the population of users.

Option C: Restoring Justice

Restoring Justice (https://www.restoringjustice.org/) is an organization which provides holistic legal defense and social services in Houston. Holistic indigent defense, also referred to as holistic public defense when provided by the Public Defender's Office, refers to approaches to legal defense that engage "teams of professionals that address a range of the client's needs rather than simply a heroic solitary lawyer who represents a defendant solely at criminal trial." This approach attempts to address not only clients' legal needs, but also seeks to ameliorate the impact of arrest and incarceration (e.g. employment, housing, nutrition, access to health care). While there has been an increase in the adoption of holistic public defense, it is still a relatively new approach that requires a great deal of further assessment. The web application, will provide a basic data platform allowing Restoring Justice to implement holistic defense principles by collecting, managing and presenting client's data.

Your first task is the implementation of an Intake Form page for the collection of client's data that can be used by the legal team as well as the social workers. Details for the intake form data can be found in Appendix C. The intake form data needs to be stored so that it can viewed and edited at a later point, but in such a way that the history of the edits is also tracked.

Next, there needs to be a page that allows a case manager to track referrals to other social service agencies and whether the client followed through on that referral and resolved the underlying issue. The type of social service offered are: nutrition, housing, legal, family services, clothing, transportation, identification, immigration, anger management, mental health, drop-in centers, disability services, job training, substance use disorder services, other, and drug testing. The page should also allow to enter the name of the case manager, record short notes, date/time as well as time spent on the case. It will also require the case managers to be able to come back at a later date to choose whether the client used the referral (Yes/No) and whether the underlying need was addressed (not at all, partially, fully).

Finally, management staff will need to be able to get a summary report that characterize both their clients and overviews of the work of their legal and social service by date/time and/or client and/or Case Manager (table format). In addition the page must show at least one summary graph related to cases. Eventually, they hope to be able to measure their staff's time and track this over time.

Requirements (same for all options)

This project will be developed in **2 Sprints**. The first sprint focuses on the backend development, setup of the MongoDB and all the functionality of the API services. The second sprint focuses on the UI development in JavaScript using VueJS. Both **code submissions** will happen into the same repository on GitHub Classroom. Do not forget to comment and document your references.

Each Sprint for the project is due at midnight of the listed due date. No late submission will be accepted. Part of the submission for Sprint 2 will be a 5-minute long **presentation video**, giving an overview over the functionality of your application features, and explaining the technology, frameworks, etc. that you have used. You will also have to submit individual peer-evaluations.

You will not need to implement a security layer for this web application (no login and roles etc.).

Select your project option together with your Team - Deadline September 9th, 2021

We are using Github Classroom to setup the remote repo for this assignment. The link to accept the assignment and create the private GitHub repo is posted in Blackboard. Accept the link and an empty GitHub repository will be created for your team.

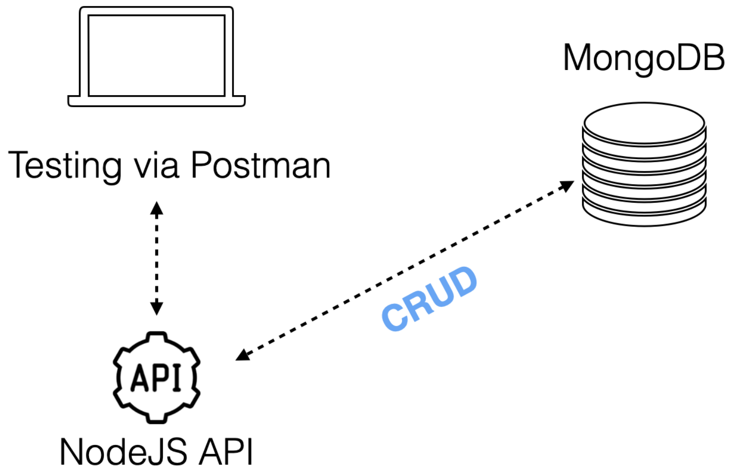
Since this is a group assignment you will be asked to join/form a team in GitHub classroom. There are three common cases, and you will have to follow one of them:

a. There are no groups yet. You need to enter the name of a new group (following the group formation table provided in Blackboard) to create it.
b. There are one or more groups already formed. You need to click on the existing group that you should join (see group formation table).
c. Your group is not listed yet. You need to enter the name of the new group to create it.

Your team must update the GitHub README file by **September 9th, 2021** with information on what project option you have selected.

Details for Sprint 1 - Deadline October 7th, 2021

For Sprint 1 you will have to design and implement all the services **(back-end)** you will need so that your app can interact with your database (CRUD operations). These services need to be implemented using MongoDB, Express and Node. You will be testing your code without the need to have the pages implemented that make up the web-based user interface for the app. A crucial part for Sprint 1 will be the design of a data model for MongoDB to effectively support the application needs. Your team will only have to submit the link to your github repo in Blackboard.



Details for Sprint 2 - Deadline November 11,2021

For Sprint 2 you will have to design and implement the **front-end** part of your web application. The front-end user interface should be based on a Node, Express and the VueJS stack. Bootstrap and/or Material Design should be used for the UI design aspects. Your front-end business logic has to connected to your backend. You will use the same github repository as for Sprint 1.

Your user interface has to follow basic UI design principles (and will be graded accordingly):

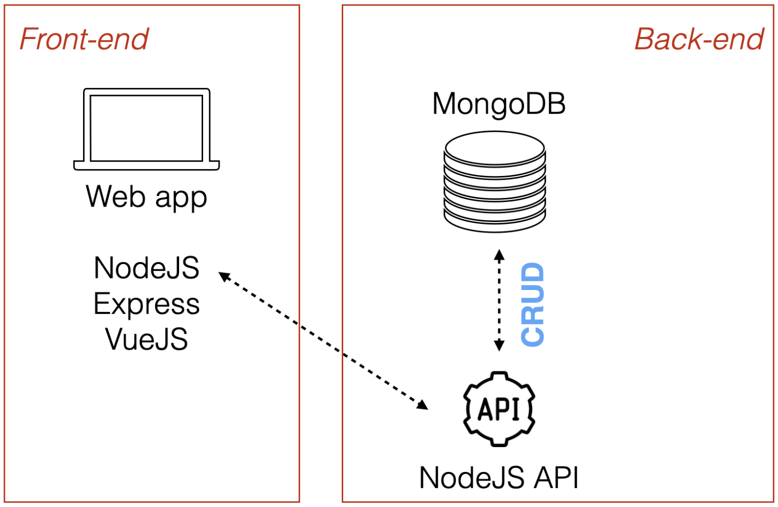
- Keep the interface simple.
- Create consistency and use common UI elements.
- Be purposeful in page layout.
- Strategically use color and texture.
- Use typography to create hierarchy and clarity.
- Make sure that the UI communicates what’s happening.
- Think about the defaults for forms etc.

Part of the submission for Sprint 2 will be a 10-15 minutes long group **presentation video**, giving an overview of the application, explains the features, and describes the technology, frameworks, etc. that you have used.

NOTE

Changes to backend due to front-end development

You are allowed to make changes to your code base for the backend after October 7, 2021. We will however grade the version that was submitted originaly for Sprint 1.



Peer Evaluation - Due November 18th, 2021

As part of this assignment you will need to evaluate 2 of your classmate’s projects. Peer evaluations will be submitted via *Blackboard*. The github repositories of all teams will be made available to you after Sprint 2 has finished. There will be a detailed rubric provided for how to evaluate the project code and presentation.

Other Requirements

- Make sure your project compiles and runs locally and the github README provides instructions for local setup. If the project doesn’t run, you will forfeit points.

What to Turn in

Commit your source code (properly commented) to your private repository in the Github classroom. Do not zip the files together. Your github repository must show multiple **meaningful commits for each of the team members** illustrating how you worked through the problem. Please push your commits on a regular basis.

There will be 2 assignments listed in Blackboard - one for each Sprit. For Sprint 1 submit the link to your github repository containing your project via Blackboard. For Spring 2 submit the link to your vidoe presentation. **Video MUST be uploaded into MS Streams and accessible** for everyone in our class team.

A 3rd assignment listed in Blackboard will be for the peer-evaluation portion of the project. This asignment will open up with all details right after the deadline for Sprint 2.

> IMPORTANT
>
> The deadlines for this project are not flexible since we need to allow for enough time for the peer-evaluation and grading.

Extra Credit

You can earn extra credit if you are able to deploy the full-stack application into the cloud. Do not attempt this until you have finished the other parts of the project. We will grade you on the local deployment first!

- Make sure the username and password for the database you setup are credentials you’re willing to share. Do not use personal passwords for this homework, which you might be using anywhere else.

Basic Grading rubric for the project

Remember the whole project accounts for 30% of the overall grade in the class. All team members will receive the same points based on the assumption that a team has equally distributed commits for all team members and every team member contributed equally. The Peer-Evaluation points are given on an individual basis.

Item	Points in Sprint 1	Points in Sprint 2
MongoDB data model setup	20	
Code for API	10	
Comments explaining API code	10	
Code for interaction with MongoDB	5	
Comments explaining interaction with DB	5	
Code for UI and business logic to work with backend (3 Pages)		15
Following UI design prinicples in front-end		5
Comments explaining UI code and business logic to work with backend (3 Pages)		10
Video presentation		10
Quality of Peer Evaluation		10
Total	50	50

Deduction 50% if code doesn’t compile	(up to -50)	
Deduction for not submitting link to Github	-5 points	
Deduction for not submitting link to MS Streams		-5 points
Deduction for not having multiple meaningful commits done by each team member showing progress	-20 points	-20 points
Extra credit for Cloud deployment	10 points	

Last updated 2021-09-03 15:20:36 -0500