

Automated Fitness Instructor Certification Website

PROJECT PLAN

Team Number: sddec19-04
Advisor: Dr. Mohammed Selim

Team:
David Bane / Group Facilitator
Ryan Menster / DB Admin
Christine Hicaro / Group Admin
Maxwell Talley / Scribe

Team Email: sddec19-04@iastate.edu
Team Website: <http://sddec19-04.sd.ece.iastate.edu/>

Table of Contents

1 Introductory Material	4
1.1 Acknowledgement	4
1.2 Problem Statement	4
1.3 Operating Environment	4
1.4 Intended Users and Intended Uses	5
1.5 Assumptions and Limitations	6
1.6 Expected End Product and Other Deliverables	7
2 Proposed Approach and Statement of Work	9
2.1 Objective of the Task	9
2.2 Functional Requirements	9
2.3 Constraints Considerations	9
2.4 Previous Work And Literature	10
2.5 Proposed Design	10
2.6 Technology Considerations	11
2.7 Safety Considerations	11
2.8 Task Approach	12
2.9 Possible Risks And Risk Management	12
2.10 Project Proposed Milestones and Evaluation Criteria	13
2.11 Project Tracking Procedures	15
2.12 Expected Results and Validation	15
2.13 Test Plan	16
3 Project Timeline, Estimated Resources, and Challenges	16
3.1 Project Timeline	16
3.2 Feasibility Assessment	17
3.3 Personnel Effort Requirements	17
3.4 Other Resource Requirements	20
3.5 Financial Requirements	20
4 Closure Materials	21
4.1 Conclusion	21
4.2 References	22
4.3 Appendices	22

List of Figures

- Figure 1: Use Case Diagram
- Figure 2: Block Diagram
- Figure 3: Automated Fitness Instructor Certification Website GANTT Chart

List of Tables

- Table 1: Major Tasks

1 Introductory Material

1.1 Acknowledgement

Throughout this project, we will work with a number of different people. Dr. Mohamed Selim is our faculty advisor. His guidance will be very useful to our team throughout the process. We also worked with student Nathan Schaffer, who started the project that ours is currently based off of. Nathan assisted us in initially getting the project off the ground.

1.2 Problem Statement

At the beginning of the semester, our client asked us to help them in the creation of a certification website for their instructors. At the time we received the project, the client kept track of all instructor information by hand in spreadsheets. This data includes basic information such as name and email, as well as certification level, CPR certification expiration, and workshops attended. Handling the information this way takes time away from people who could otherwise be doing more meaningful work. Bringing this process online would both increase the accuracy of the information as well as greatly decreasing the amount of bookkeeping work they have to do.

To bring the client's certification process online, we will need to design and implement a new website for them. This website will include a database to store all of their instructor information, upcoming workshops, and gym locations. The website will also include a fully functional user interface so instructors and managers can access all the data they need as well as automating as much of the certification process as possible. This website should greatly decrease the time our client needs to spend on their instructor certification process.

1.3 Operating Environment

The operating environment for this project is relatively simple. With it being a web application it's initial intended use is by a computer. This computer being at a business will most likely be within an office. Any data being stored will be on an offsite server maintained by a third party. With this being, the environment causes minimal concerns for the safety of the project and those involved.

1.4 Intended Users and Intended Uses

The intended use of this product is for instructors to be able to submit and apply for certifications. This will allow for much of the work to be automated instead of the current system being by hand. There will be two types of users for the system, managers and trainers. Trainers will be given a login whereas managers will have to submit a registration form. This also creates two sets of permissions towards available data.

Many of the uses are shared between the types of users. The shared uses are viewing workshops, locations, notifications and routine login behaviors. The uses unique to managers are viewing a master trainer list, reviewing applications and deciding to approve or decline those applications. For trainers their unique use is the registration and submitting their applications. A server is also an actor in this scenario. The server will be in charge of authentication, sending notifications and sending messages.

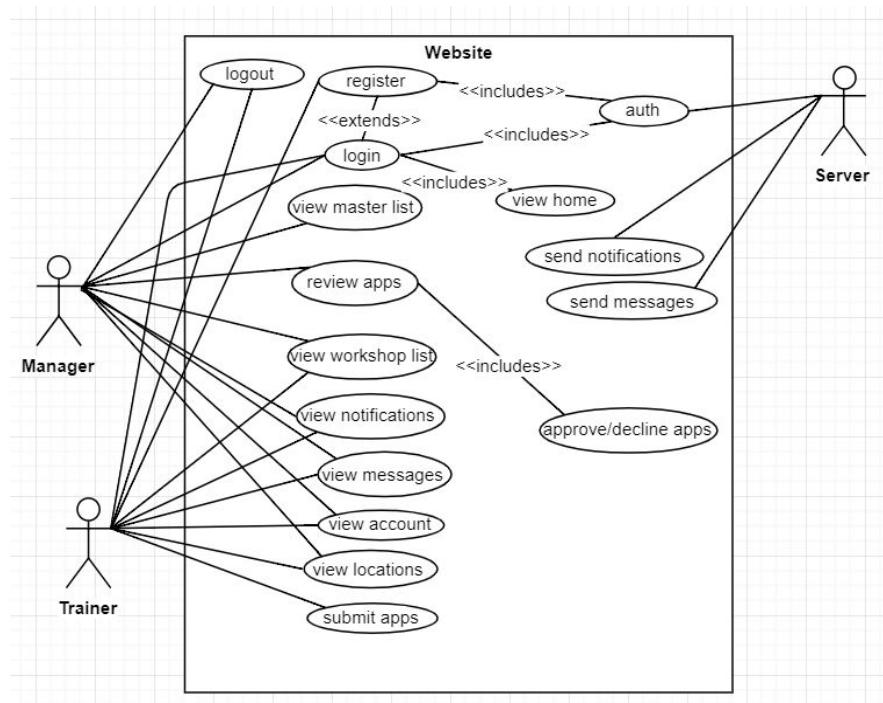


Figure 1: Proposed Use Case Diagram

1.5 Assumptions and Limitations

- Assumptions
 - An existing style has been set and can be used if we wish
 - A previous intern had created an initial frontend of the website
 - We can base our design off of his already existing design
 - We are free to create the backend with whatever technology/programming language that we wish
 - Currently, we plan to use Express, a Node.js framework
 - We are free to create the frontend with whatever technology/programming language that we wish
 - We will be using React, a JavaScript library that specializes in creating user interfaces
 - We will be using material-ui, a React framework, to assist in development
 - We are free to use any hosting platform that we desire, provided that its costs are manageable (i.e. around the same cost as Amazon Web Services)
 - Currently, we plan to use Microsoft Azure
 - A database will be provided via the hosting platform
 - Up-to-date instructor certification information will be provided at the start of implementation
 - The website will not need to be too scalable
 - User count will remain relatively consistent throughout the website's lifecycle
 - The product will be used only on computers
 - Mobile devices will not need to be accounted for
 - Only Managers can create accounts for Instructors
 - There are only two types of users: Managers/Owners and Instructors
- Limitations
 - Current limitation of expected requirements
 - Will require future meetings with client to verify that our current direction is correct
 - Budget will be practically nonexistent for our team
 - Test cases can be difficult to create, due to number of components involved
 - Extensive database information is currently unknown
 - No current structure for the backend is known
 - Original client is no longer with the project

- Will need to decide which components that the client initially wanted will be kept and which will be changed
 - Will need to determine if a new “faux client” will be implemented
- Group members’ schedules do not line up well with each other
- Website must be intuitive and easy to navigate through
- Group members are not familiar with the programming languages (React and Express) that will be used for the project.

1.6 Expected End Product and Other Deliverables

There are three primary functional components that will be delivered at the end of the project: a frontend, a backend, and a database. These components will work together to create a fully functional Automated Fitness Instructor Certification Website. This website will streamline the certification process that the client currently does by hand. The website itself will have multiple components and deliverables that can be delivered to the client iteratively as they are completed.

First, we plan on having a final design plan laid out by the end of the first semester of the project. Next, we will have the full user interface of the website created and delivered during the first full month of the second semester. Following that, we will deliver a functioning database that will be filled with up-to-date instructor certification information, location information, etc. that can be used in the communication between the database and the backend of the website by the middle of the second semester. A functioning backend that can communicate with both the frontend and the database will be delivered before Thanksgiving break. Finally, the completed Automated Fitness Instructor Certification Website will be delivered by the end of the second semester. No other additional materials will be delivered alongside the finished website unless the client requests them.

- Final Design Plan - Delivery Date: May 3rd, 2019
 - This will be presented to the stakeholders that are invested in the project, if applicable, to verify that the plan laid out in the Final Design Plan is correct and satisfies all of the expectations that the client has. If a third party gives confirmation that the plan laid out in this document is satisfactory, our team will begin the early stages of implementation. If the document is not satisfactory, our team will meet with the third party and discuss the changes that need to be made. Our team will then recreate the document until the client deems it satisfactory.
- User Interface - Delivery Date: September 13th, 2019
 - The visuals of the frontend will be completed and delivered by the selected date. This will include any visual objects required in the website, such as buttons, text fields, tabs, etc. The functionality of these objects might not be implemented at

this time, as it requires a functioning backend to work successfully. The frontend is expected to be intuitive and easy to navigate. This frontend will be delivered to the the stakeholders, if applicable, to verify that the design is up to their expectations.

- Functioning Database - Delivery Date: October 18th, 2019
 - A functioning database will be delivered that contains the up-to-date instructor certification information that is used in the current certification process. This information will be able to be added to the database, removed from the database, and modified within the database, among other essential functionalities. Data within the database will likely be connected to other data within the database through the use of foreign keys or a similar mechanism. This database will have the ability to communicate with the backend of the website and will be hosted via Microsoft Azure.
- Functioning Backend - Delivery Date: November 22nd, 2019
 - A functioning backend will be delivered that is able to connect to and communicate with the database to send and/or receive data such as an instructor's certification level or the location of an upcoming workshop. The backend should also connect directly to the frontend of the website and allow for full functionality of the different objects within the frontend to be achieved. The backend will be created using Express, a Node.js web application framework that specializes in backend support. The backend will be hosted via Microsoft Azure.
- Automated Fitness Instructor Certification Website - Delivery Date: December 13th, 2019
 - The fully implemented Automated Fitness Instructor Certification Website will be delivered that will fulfill all the requirements decided upon during the first semester of the project. This website will allow a Manager to create an account for an Instructor that defaults to a Certification level of "Trainee." As an Instructor's career advances, they can apply for recertification or an increase in Certification level, which a Manager then approves or disapproves. The website will be accessible to any computer that provides the proper credentials and have a robust user interface and database. The user interface will be intuitive and easy to navigate through. The source code used for the project's frontend and backend will be available to the client via ISU's Gitlab, as well as be available on the server hosted by Microsoft Azure. No additional documentation will be provided unless the client requests them.

2 Proposed Approach and Statement of Work

2.1 Objective of the Task

The goal of the task is to create a web application that provides a service that automates the certification of fitness instructors. With the web application, instructors will be able to apply for a new certification level while managers will be able to approve new certification levels and submit trainer and workshop data.

2.2 Functional Requirements

The following is a list of functional requirements of the project

- A well constructed database schema: the database must be set up in such a way that is intuitive and efficient to execute queries.
- Displaying lists: there must be a way to display lists such as the instructor list, workshop list, and location list quickly and correctly.
- Authentication of employee or instructor: checks must be made that the users entering the system are only employees and instructors.
- Use of individual trainers data: the use of this data will be used to automate instructor certification.

2.3 Constraints Considerations

Constraints:

- Instructor and Manager data must be protected, as it is sensitive data
- Project must abide by ABET standards
- Project potentially must abide by IEEE standards

Non-functional requirements:

- System should only be accessible by instructors, managers, and owners. Not by the general public.
- The database should be completely secure, accessible only by management.
- Both the web server and database should be set up in a maintainable way so that another developer can make changes to this codebase in the future if needed.

For our project, we are building a both the front and backend of our website, as well as the database. There is not a generally followed protocol for creating web applications. Our website shouldn't be considered unethical by any protocols, though.

2.4 Previous Work And Literature

Similar products are bound to exist in this space but to the best of our knowledge, none exist in the public space that are hosted by a gym. There are online fitness instructor certification sites, but that is a general certification that can be taken to different gyms. This certification will be specific to a gym's specific requests.

Our team was given previous work to start from in the form of a partially functioning frontend application hosted on Amazon Web Services. We were told that the frontend is not very in-depth and we could start over if we so wished. As of now, the frontend is built in the frontend framework Angular. This framework is made by Google and has a fairly steep learning curve. None of us have worked with it before, so we chose to go with React, a much more reliable framework instead. Moreover, we decided to use Material-UI, a React UI framework, to help build the UI in a cohesive, clean way. We also made a similar switch from Amazon Web Services to Microsoft Azure.

We will be able to reference the previous work given to us and use it as a guide to what the web application should have. Besides giving us a basic outline of application, the previous work given to us does not provide us with any more value, for we are re-developing the entire project from the ground up. As for literature, there is an abundance of online resources for learning React and Material-UI, including the React and Material-UI documentation, tutorials, and blogs. There is plenty of information available for Microsoft Azure, provided by the Microsoft Azure documentation, the Microsoft Azure support team, and other online resources.

2.5 Proposed Design

The design that our team proposed to use is a simple Client-Server design. This is a very common design approach for web applications, as it involves the two primary components of a typical web application: a client (frontend) and a server (backend). This is a simple software architecture design to implement, and our team believes that we can successfully implement that design into the project. There are some downsides to this design approach, but our team believes that we can successfully implement this design into the project and that the benefits outweigh the costs. If necessary, there do exist alternative designs that work well with web applications, such as the Web Application Architecture. However, our team has little experience with this design style and feels more confident working using the Client-Server approach.

Overall, our design for the project is pretty simple. We will have a frontend of the application that is essentially the user interface. The frontend will communicate with the backend, which we will manually create using a programming language that will be decided on in the future. This backend/server will provide a central area of communication between the frontend and the database, which our team will also create. The backend will be responsible for

most of the work behind the scenes, such as analyze certification applications. The database will be where the data related to Instructors, Managers, workshops, and more are stored. These three components will seamlessly interact to accomplish all of the use cases that our team has come up with. We are confident that following the planned design will allow us to complete the project on time.

2.6 Technology Considerations

For our project we have to implement a backend. In order to do this we had to consider what service we would use as our server. For this we decided to consider Amazon Web Services, Google Cloud or Microsoft Azure and decided that Microsoft Azure would be the better option. This decision was due to the availability and reliability of the service. The frontend framework had already been set as AngularJS but there was an option to redesign this. We decided that it would be best to use a more reliable, up-to-date framework and switch to React. There had also been attempts at connections between the front and backend in the past so we look to just fix the issues that exist. With roughly a semester to implement, planning for efficiency will be our best option.

2.7 Safety Considerations

For this project the safety considerations are extremely minimal. With this being a web application, the only location it will be used is on a computer. This environment is typically a safe, comfortable environment. The only safety issues that could arise is achieving the certifications submitted on the website or injury when working with a trainer listed on the website.

2.8 Task Approach

Our solution is separated into three parts: the database, the web app, and Microsoft Azure. Our database and web app will be hosted through Microsoft Azure. Through the web app, you will be able to access two different profiles: manager or instructor. Depending on the profile you login with, the user will be able to see the information listed under the respective profile.

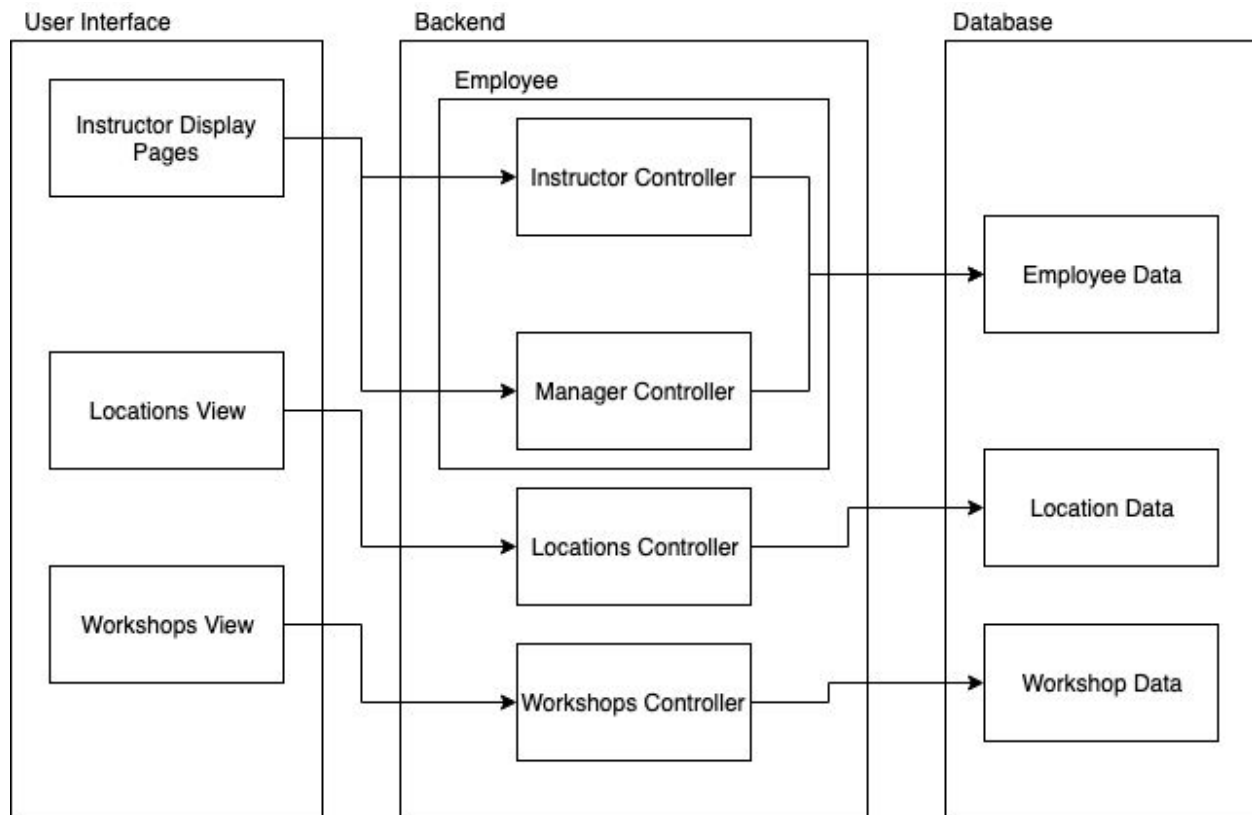


Figure 2: Block Diagram

2.9 Possible Risks And Risk Management

There are a few concerns/risks involved with this project, but none are very devastating. The largest of these concerns is that our team schedules do not line up well with each other. This means that there are very few instances where every team member is available to meet up to work on the project. This can cause issues if a project roadblock appears that requires every team member to work together to solve. This also makes it difficult to coordinate meeting times with any third party, as they also have their own time constraints. This can cause delays in the project, such as if one team member requires assistance from another member, but they cannot meet with each other for an extended period of time. This can also lead to features being blocked from being completed, which can cause the project to be delayed.

Much of the project is hosted via Microsoft Azure, which only the Scribe, Max Talley has experience with. Other team members will have to take the time to learn how to work with Microsoft Azure, which can delay project progress, depending on the steepness of its learning curve. It is possible that, at the beginning, Max Talley will be responsible for a majority of the Microsoft Azure related work. This will likely be the case until other team members become proficient enough to successfully contribute.

The frontend of the website will use React to format the user interface. This is a Javascript framework that is very difficult to learn, as it has a steep learning curve due to how the code itself is compiled. Every team member has dealt with React in previous classes at Iowa State University, but none feel that they have the appropriate skill to work with it without having time dedicated to learning it beforehand. Each of the team members will take time to learn how to use React through tutorials and any documentation available. This will initially create a delay of the project, depending on how long it takes each team member to have a proficient grasp on the framework. This means that as each team member grows stronger in their skillset that they help those around them that are struggling a bit more.

A minor risk is the possibility of Microsoft Azure Services going down. The chance of this happening is very low, but it is still a possibility that our team must keep in mind. Since much of this project will be hosted via Microsoft Azure, our team must prepare for a significant loss in time if the services go down during the project implementation. If Microsoft Azure goes down, our project essentially cannot be worked on.

2.10 Project Proposed Milestones and Evaluation Criteria

- User Interface is Complete
 - The user interface is what someone sees when they visit a website. One of the first things that needs to be completed for this project is the user interface, as it is the only aspect of the project that is likely to stay the same throughout the project's lifecycle. Once our team feels like the user interface is finished, it will be tested. The testing will be done to ensure that the user interface works well on different devices of different sizes, as there is a possibility that this website will need to be accessible and readable on an iPad or mobile device.
- Database created
 - The database is essential to ensure that instructor certifications are correctly performed and stored. This is the backbone of the entire website. Creating the database is an important first step towards completing the project. Once the database is created, it can be tested by seeing if data can be entered, removed, and modified. These three functionalities are extremely important to have working correctly. An additional test would be to see if foreign keys implemented into the database work as expected.

- Backend can connect to database
 - The next milestone is verifying that the backend can connect to the newly created database. This is important because it is the only way for the website to receive and send data to the database. This is necessary for the completion of the project, as without this, no data about Instructor certification can be sent or stored inside the database, which makes it impossible to use the website how it was meant to be used. This can be tested by having the backend send data to the database, as well as receive data from the database. If this can be done correctly and consistently, it will be considered complete.
- Backend can connect to frontend
 - The backend must be able to connect to the front end because that is how we will be able to add functionality to the buttons and different objects inside of the front end. Without this connection, the entire website will be useless, as the front end will not be able to perform any operations, like submitting a certification request, for example. Once a user on the front end is able to interact with the website via pushing buttons or submitting forms, this milestone will be considered complete.
- Backend complete
 - Once the backend is able to connect to both the database and the front end, and the core functionality of the website is working, testing will begin to see whether or not the backend can be considered complete. This testing will involve a user on the frontend making a request or submitting a form that the backend then processes and communicates with the database to complete. If this can be done reliably and consistently without any errors whatsoever, the backend can be considered complete.
- Project complete
 - Once all of the functionalities have been implemented, our team will begin to test the website as a whole. This involves executing many test cases designed to check for every functionality that was supposed to be implemented. This includes, making accounts, editing account information, submitting a certification request, adding workshop, etc. If all of the test cases pass successfully and the overall design is approved, the project will be considered complete.

2.11 Project Tracking Procedures

Our team plans to use weekly status reports from every member to keep track of any progress made each week. This will allow us to hold each other accountable and ensure that everyone is putting in the work necessary for the project to succeed. Our team plans to

implement an Agile Development Process model. With this, each team member will be assigned tasks to do for every iteration/sprint. Each task will come with a description and a delivery/due date. It is each team member's responsibility to ensure that their tasks are completed on time and are of high quality. It is common that some tasks will not be able to be completed on time every iteration, so we will be sure to spread the work out enough so that the occasional missed deadline does not result in a significant delay to the project.

The group also plans to use Iowa State University's GitLab to store source code, as well as to create and keep track of milestones. This tool will be extremely useful, as it allows us to work on the project at the same time without having to worry about overriding someone else's work. When a team member commits to the project, he/she must provide a descriptive commit message that accurately and clearly describes the changes that he/she made to the project. If any merge conflicts occur due to multiple members modifying the same file, it is the responsibility of those members to meet with each other and resolve the conflict(s). The group will also use a tool such as Trello to help with streamlining the Agile process.

2.12 Expected Results and Validation

The desired outcome is to have a robust web application that automates instructor certification. It can be confirmed that the web application behaves as intended if an employee or instructor can successfully log in and utilize the services provided by the web application (i.e. instructor certification automation, instructors applying to new level, automated emails sent to instructors) with ease, there is efficient data collection, and there is exemplary user satisfaction with the end-product.

2.13 Test Plan

The current test plan is to create a git flow testing system where another team member has to test a commit before it can be merged into the development branch, and then another approval must be made before moving development into the master branch. Testing will be frequent and carried out throughout the entire project up until completion. Unit tests may also be plausible when necessary.

3 Project Timeline, Estimated Resources, and Challenges

3.1 Project Timeline

For the first semester, we will focus on getting all of the necessary details to quickly and effectively create the Automated Fitness Instructor Certification Website in the second semester. This will be done consistently throughout the semester by communicating with the client, as well as by completing the relevant assignments in the Senior Design course. These assignments, along with our team’s constant communication with the client, will hopefully provide us with enough information to be fully prepared to start implementing the project during the second semester.

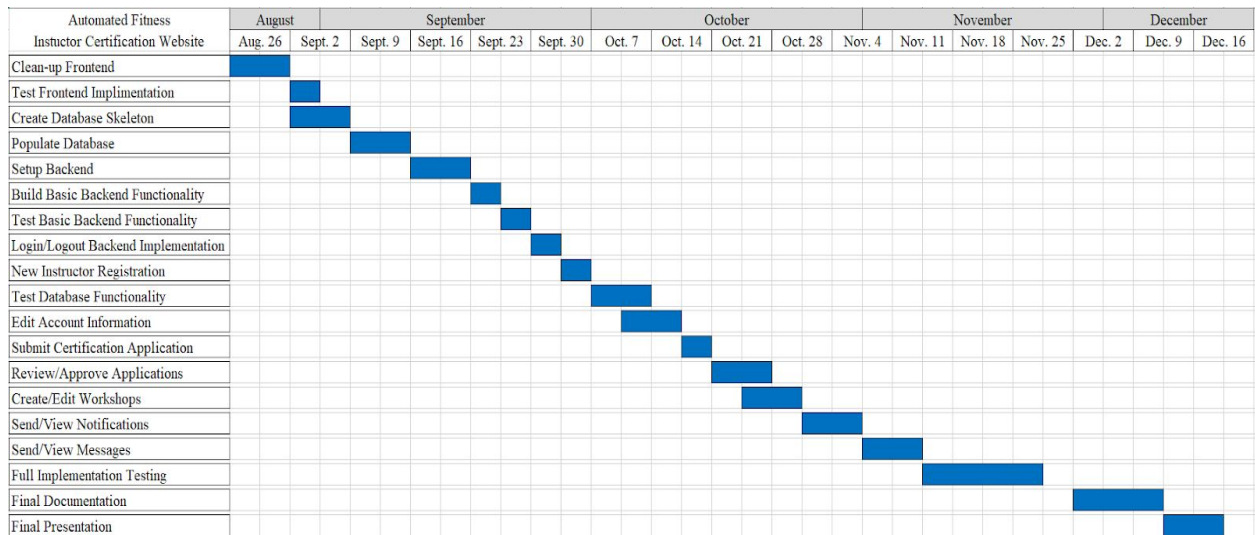


Figure 3: Automated Fitness Instructor Certification Website GANTT Chart

Description of Figure 3: Automated Fitness Instructor Certification Website GANTT Chart

For the implementation of the Automated Fitness Instructor Certification Website, our team chose to use an Agile approach. In this approach, we will designate tasks that will work towards accomplishing certain features of the project. Since this approach is iterative, the tasks were selected to be smaller in scale. This allows for constant feedback from the client. If the tasks were large and the client did not approve of the solution to a certain task, major changes in the project might occur. Keeping the tasks smaller allows for more flexibility when it comes to scope changes. At the beginning, our team will focus on the larger architectural features of the project, like getting the frontend cleaned up and getting the backend functioning. The typical pattern involved in these major components is Set Up > Implement > Test. If this chart is followed, by the end of the semester, we will have completed the Automated Fitness Instructor Certification Website. If our team finds that some tasks can be done concurrently with others, the project could end up being ahead of schedule, which would allow for stretch goals to potentially be completed.

3.2 Feasibility Assessment

Our team expects to finish the project within the allotted amount of time. We believe that even with the unforeseen challenges that are bound to pop up, we will make our deadline. All of us have at least some experience with creating web applications, which will be an advantage in completing on time. A possible challenge that may arise during the process is new requirements. If the client decides they want a feature added partially through the project, we will have to adjust and fit it into the timeline. Other challenges include team members having to learn how to use React, Express, and Microsoft Azure. The time needed to learn the necessary components of these has a potential to be large, which can cause a delay in the project.

3.3 Personnel Effort Requirements

As seen below, there many tasks that our team plans to complete by the end of the second semester. Implementation of the different features of the website will take the most time, but plenty of time will also be spent on testing to make sure that the website runs as expected. The total estimated time to complete this project is 570 hours. With our schedules, this comes out to about 16 or so weeks. This will provide us with enough work to last the entirety of the second semester.

Major Tasks

Task	Description	Estimated Time
Clean Up Frontend	The frontend of the website must be cleaned so that it visually appears how it should in the final implementation of the project.	30 hours
Test Frontend Implementation	The frontend of the website must be tested on different sized monitors, as well as different devices to ensure that the website is properly laid out on all of them.	10 hours
Create Database Skeleton	A basic skeleton of the database must be created so that future implementation is simplified as much as possible. This includes creating the tables and foreign keys, if necessary.	30 hours
Populate Database	The database will be populated with up-to-date information on Instructors and their Certification Levels. The database will also include account information, workshop information, and other key information needed for the completion of the website.	40 hours
Setup Backend	The backend of the website must be set up using Express. The backend should be able to be connected to through a terminal when this task is completed	35 hours
Build Basic Backend Functionality	Basic backend functionality will be created during this task. This includes connecting to the database, connecting to the frontend of the website, communicating with the database, etc.	20 hours
Test Basic Backend Functionality	The basic backend functionality will be tested to ensure that it is communicating cleanly with both the database and front end of the website. This testing will be relatively rigorous so that it does not need to be returned to in the future.	15 hours
Implement Login/Logout Functionality	The first component that we will create is the ability to login and logout of the website. This is a basic component that will be important to have in the future of the project.	20 hours

Implement Registration Functionality	The ability for a Manager to register a new user will be done in this task. This is the only way for a new user to be registered, so it is very important that this task gets implemented early.	20 hours
Test Database Functionality	The database will be tested to ensure that it is sending and receiving data properly. This can be done through the website or through HTTP Requests. This testing will be vigorous to ensure that the database is functioning correctly and does not require additional testing in the future.	30 hours
Enable the Ability to Edit Account Information	The ability for a Manager or Instructor to edit their account is very important, as that is where they store their personal information, such as BMI, email, etc. Allowing the users to change this information will provide them with flexibility if their information changes in the future.	25 hours
Enable the Ability to Submit a Certification Application	The ability to submit a certification application is one of the most important parts of this project. This is the foundational aspect of the project and must be completed for the project to be considered complete. An Instructor should be able to fill out the application with the relevant information and submit it for approval.	20 hours
Allow Managers to Review/Approve Applications	This task is very important to implement for the project to be considered complete. Managers should have the ability to look at the applications submitted by Instructors. Here, they will make the decision on whether or not to approve the given Instructor's application. They will be able to see all of the information that the Instructor submitted.	40 hours
Enable to Ability to Create and Edit Workshops	Workshops are a important when it comes to an Instructor applying for certification or recertification. It is here that they are able to learn the necessary skills. A manager or authorized user should be able to create a new workshop or edit existing workshops.	30 hours
Enable Notification	Notifications should be sent to Instructors or	30 hours

Sending and Viewing	Managers when important events happen or are about to happen, such as workshops.	
Enable Message Sending and Viewing	Messages should be able to be sent and viewed by Managers and Instructors. These messages will relay important information about workshops or other events, among other things.	25 hours
Full Implementation Testing	Near the end of the semester, extensive testing should be done on the system as a whole. This testing will determine whether the current product is suitable to present as completed.	90 hours
Create the Final Documentation	The final documentation will be created to present to the Senior Design class, along with any third parties.	40 hours
Host the Final Presentation	A poster and presentation will be made to showcase the Instructor Certification Website to the Senior Design staff.	20 hours

Table 1: Major Tasks

3.4 Other Resource Requirements

For this project there will be no extra parts or materials required. This is a web application so the only tools used are on a computer. Our entire team currently has access to individual computers and the software required.

3.5 Financial Requirements

There will be next to no financial requirements on our part for this project. Overall, there is a financial requirement for acquiring the rights to use Microsoft Azure. This is expected to be purchased and able to be used by early June 2019. There is a potential for a financial requirement to be necessary when building the database for the project, though the financial burden for this is expected to be very minimal.

4 Closure Materials

4.1 Conclusion

Currently, employees manually enter instructor certification data into spreadsheets, which is a tedious and timely process. Moreover, the current instructor certification information process is also costly considering employees spend their time entering in the instructor certification information when they could be doing other work that has equal or higher priority. Automating the instructor certification process within a web application that both managers and instructors can utilize will help keep the instructor certification data be more consistent and accurate while providing an organized approach to how the data is collected and managed.

A web application that automates the instructor certification process is the overarching solution to a current manual process problem. A new database will be created to keep track of instructor information, upcoming workshops, and attendance, along with anything else that is currently kept track of manually. The user interface for the web application will be intuitive and organized in such a way that any instructor or manager can use it with ease. Although the users will be restricted to managers and instructors, these users will be able to access the web application anywhere. As a result of this web application, the employees will be able to quickly and efficiently work through the instructor verification process without the hassle of manually entering everything into spreadsheets.

4.2 References

It is worth noting that we use “the client” vaguely when talking about our client instead of “Farrell’s” because Farrell’s eXtreme BodyShaping is no longer our client.

[1]"Amazon Web Services (AWS) - Cloud Computing Services", *Amazon Web Services, Inc.*, 2019. [Online]. Available: <https://aws.amazon.com/>. [Accessed: 24- Apr- 2019].

[2]"Angular", *Angular.io*, 2019. [Online]. Available: <https://angular.io/>. [Accessed: 24- Apr- 2019].

[3]"Express - Node.js web application framework", *Expressjs.com*, 2019. [Online]. Available: <https://expressjs.com/>. [Accessed: 24- Apr- 2019].

[4]"Farrell's eXtreme Bodyshaping", *Extremebodyshaping.com*, 2019. [Online]. Available: <https://extremebodyshaping.com/>. [Accessed: 24- Apr- 2019].

[5]"Microsoft Azure", *Azure.microsoft.com*, 2019. [Online]. Available: <https://azure.microsoft.com/>. [Accessed: 24- Apr- 2019].

[6]"React – A JavaScript library for building user interfaces", *Reactjs.org*, 2019. [Online]. Available: <https://reactjs.org/>. [Accessed: 24- Apr- 2019].

4.3 Appendices

No hardware was used for this project, so there are no data sheets or related materials available. Common programming libraries will be used within the web application, so there are no appendices to list for this project.