INDIANA UNIVERSITY SOUTHEAST

**From**: Chad Wilson  
 Conner Mayfield  
 Harris Chaudhary

**Project for P445**

**Software Detail Design**



**Date Submitted: 10/31/21**

**Overview:** This project will be used by Humana that will allow them to have a log of who does what on each individual system and project that they have. This will mainly be used by the software development team there that will allow them to log into the program depending on their status that will decide what they can and cannot do. Most of the functionally will come from the software dev team to add in what they did to each project with a specific amount of time that they have. It will allow the project manager to see who has completed and what their team has done over the given specific amount of time or even what has happened in the total lifetime of the project.

**Data Design:**

* Landing Page – This is the first component the user sees. The purpose of this component is to offer a friendly introduction of the software and allow for a place where another component the “Login System” to be placed. There is very little input for this component buy the user since it is just a landing page that is a part of the UI. The output, however, is very important because this page gives access to the login system for the user which allows the user to access the main application dashboard. The component process to convert input to output is actually the login system on the landing page. When a user logs in it connects the landing page to the user/employee dashboard and a user cannot access the dashboard without proper credentials. Any user or employee can look at the landing page though. The design constraints and performance requirements of this component are rather quite simple. All their needs to be design and preform wise for this component is a user-friendly UI where a login system can live. The process (pseudo-code) algorithm for this component is going to be HTML, CSS, JavaScript, and TypeScript for the user interface (UI).
* Login System - This is the second component the user sees. The purpose of this component is to allow security within the overall software system. A username and password will be granted by clicking on the register button. Once the user/employee has their credentials they can login and then access the user/employee dashboard. The input for this component is when the user registers a new account in the software system, clicks on the forgot password button, and logs into the software system successfully. The output for this component is you receive a new password to login/ you login to the system successfully and you access the user/employee dashboard. The process to convert input to output is the authentication process of the login system. If a user tries to sign up or log in to the system but doesn’t have the correct permissions or their password is wrong then no correct/successful output will be displayed. The design constraints and performance requirements of this component are an email, username, and password file with also a forgot password button as well. Furthermore, design aspect of this component is identical to the performance requirements that are needed for this component. The process (pseudo-code) algorithm for this component is going to be HTML, CSS, JavaScript, and Typescript for the user interface (UI) while if there is any backend code involved it will be in JSON JavaScript, and C#.
* User/Employee Dashboard - This is the third component the user sees and is the main component of this software system. The purpose of this component is providing the main functions to the user/employee and allow them to do their tasks as desired. The input for this component is when a user/employee clicks on a certain feature within the dashboard. This could include clicking on the admin button if they have admin privileges and want to change a due date of a project, add an update on the current progress of a project, etc. The output of this component is that the particular field on the dashboard is updated and displayed to the user, users, employee, or employees. The process to convert input to output are the functions that are tied to each button/tab throughout the entire dashboard and makes sure everything works. The design constraints and performance requirements of this component are the particular fields of the dashboard that were requested by the client. This includes fields like project ID, project due date, current employee (developer) working on the project, admin access, database connectivity and much more. The process (pseudo-code) algorithm for this component is UI elements like buttons and widgets built in HTML, CSS, TypeScript, and JavaScript. While the code for the backend of this component will be in JSON JavaScript, and C#.
* Database - This is the fourth component which a user or employee most likely doesn't see. The purpose of this component is to be a place where all or most of the data for the software is stored. The input for this component is the users/employee's data which is their username, email address, and password whenever they register a new account or log in to the software. The output for this component is data going back to the system to allow the user to create an account or log in successfully. The process to convert input to output is email address, username, and password of the User/Employee. All of this data is stored within the database and every time a user/employee enter data into the software system this data inputted and outputted out of the database. The design constraints and performance requirements of this component are quite simple because the client just needed a database to store the employee data. While the software is being developed, we will be using a testing SQL database for the data but whenever the software is ready for deployment in Spring 2022 the Humana developers/engineers will most likely have to connect this application to their database due to client confidently. The process (pseudo-code) algorithm for this component is connecting a SQL database to the software by creating backend services to read the data from the database. These services/controllers will most likely be done in JavaScript or C#.

Database ER Diagram:

Diagram

Description automatically generated with medium confidence

**Architecture Design:**

* Diagram

  Description automatically generated
* Diagram

  Description automatically generated
* Diagram

  Description automatically generated

**Interface Design:**

* Introduction: Inside of this part we will be talking about how the computer and the user will be communicating together. While our program will be running it will be connected with the main database that Humana uses. They will be talking when the user request something. Whenever the user will request it will send a message to that database on either what the user needs or to get something the user needs. The main purpose of this part is for the user to be able to speak to the backend and get the information that they need or to give the information to the server/database.
* Input: The main input for this will be using keyboard and mouse from the user and then that will communicate with the frontend that is being displayed based off of what the website that we are using is feeding to it. That information will then be sent to the backend and that will gather the corresponding info.
* Output: The output will be a message on the screen based off the information that the backend/database is going to gather. The message will show what the user has requested or if they do not have access to the information then it will show an error message for them. If the user is trying to get something that does not exist, they will get a does not exist message, but if they can create something based off their status then they will be prompted to create it.
* Process to switch from front end to back end: Whenever the user request something then it will immediately be sent to a checker to see if they have access to that information and or that area of information. If they do not then it will send an error message. If they do, then it will be sent through a system that will ask if the database has anything in it. If not then it will be created if applicable.
* Constraints for it will be able to find all of the user algorithms that we need to implement.

**Procedural Design:**

* Diagram

  Description automatically generated

**Break down of individual contributions:**

Chad: will be dealing all the documentation and getting all of the project management organized. Will also deal with everything that involves users and processes. Also, I will be dealing with most of the user characteristics, the constraints, assumptions, and dependencies. I will also deal with all the memory constraints

Harris: Will oversee helping conner and myself by helping conner code and by helping me do some of the documentation. Will be dealing with all the operations and the side adaptation requirements, will also be dealing with all the memory constraints and with the communication interfaces

Conner: Will be dealing with the System Interfaces, Hardware interfaces, Software interface, and User interface. Conner will be dealing with most of the web-based application and that connects them together. He will also be dealing with all of the data connection between databases and our program.

**Key personnel for this project are:**

Chad: the team leader who also is making a giant list of all our functions and characteristics. I will also oversee all the documentation of the projects and speaking with the sponsor.

Harris: will be teaming up with Conner and getting all the angular and web base set up so we can have a running version of our program. Will also be doing some of the documentation.

Conner: will be the main programmer and he will be getting all the web-based items set up and connecting everything together.

Hollie: The sponsor. She is the employee at Humana that has given us the project and she has been speaking with Chad a lot about the specifics of the data and types.