Group A Final Project Report

Andy, Briana, Marc, Matt

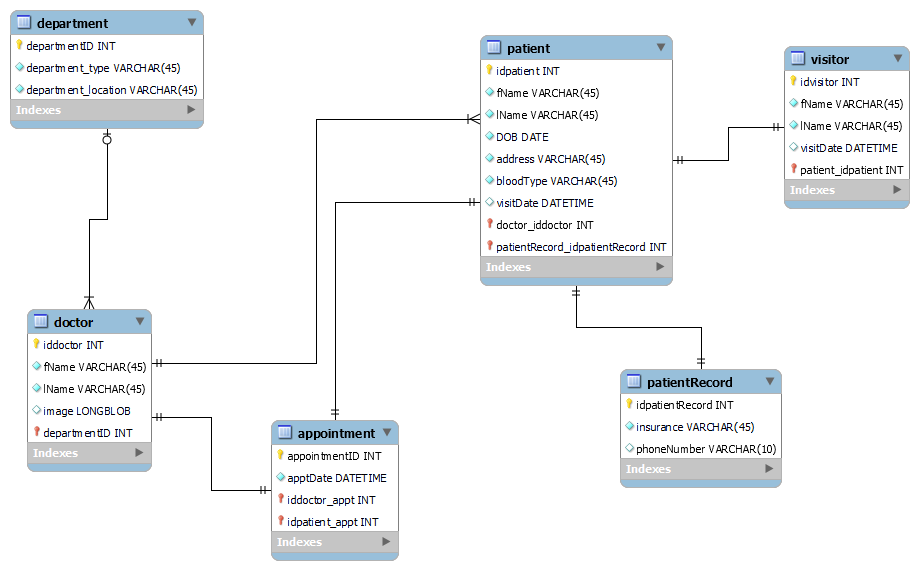
Our group project focused on implementing a database to simulate a hospital portal where a patient can schedule appointments, view doctors and their information, and visitors to the hospital are able to schedule visits with a patient. We used a server to host a webpage that was coded in php and connect to the database to accomplish the tasks we listed. This project has direct ties to real world applications, so we felt that it was an interesting project to attempt. While an actual hospital database would take a long time to develop and has many different constraints, we were able to construct a basic model that uses business rules and procedures/triggers to simulate what an actual interface of this kind would do.

Our overall goal was to create as realistic and seamless an experience as we could and we were able to accomplish this by utilizing a database coded in MySQL and a frontend web page coded in php that was hosted on a server. We began creating a webpage and hosted it using the AWS server. After that we were able to use PHP code to connect to our backend which was a database. We then began to implement our relational schema by adding entities to the database. Using these entities we then were able to implement our webpage to help us execute CRUD operations. Once we were successfully able to insert data, retrieve specific information from that table, update the table values, and delete data we then set out to implement the extra credit additions. We added drop down menus, created procedures and triggers, got data to fetch in real time, and uploaded images to our web page using bootstrap. After these things were finished we continued to refine the visual aspects of our project and made our webpage look aesthetically pleasing.

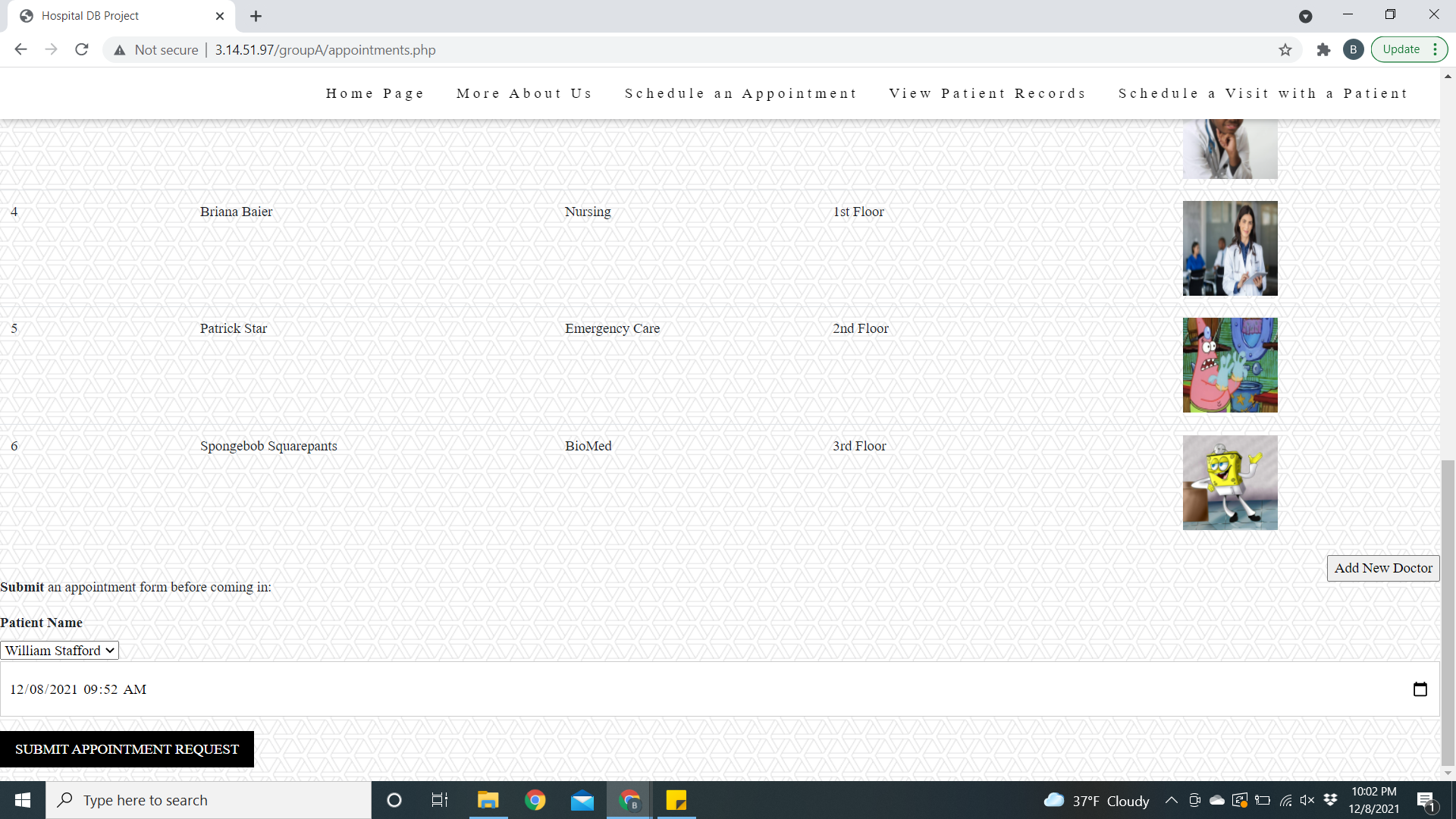
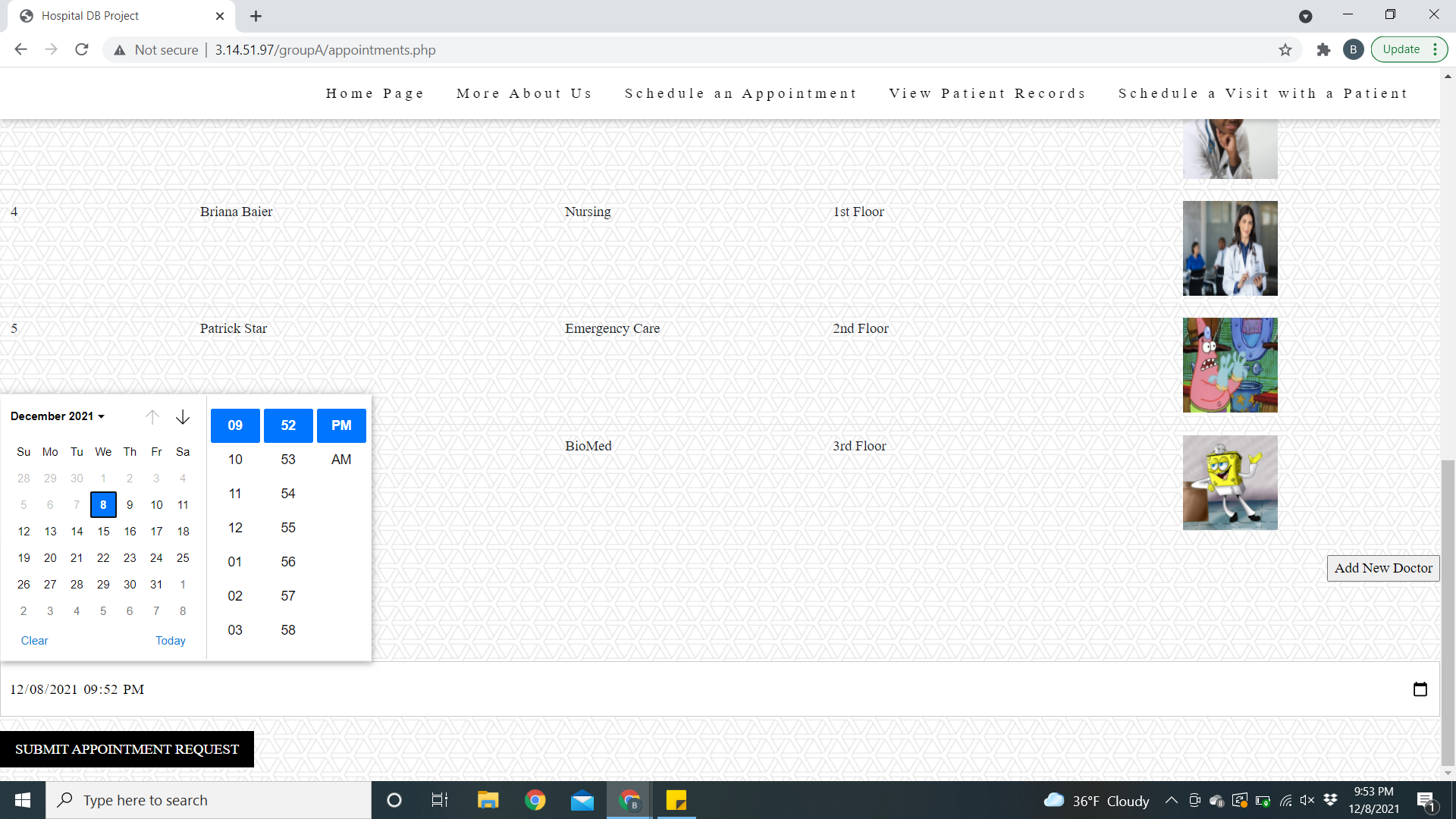
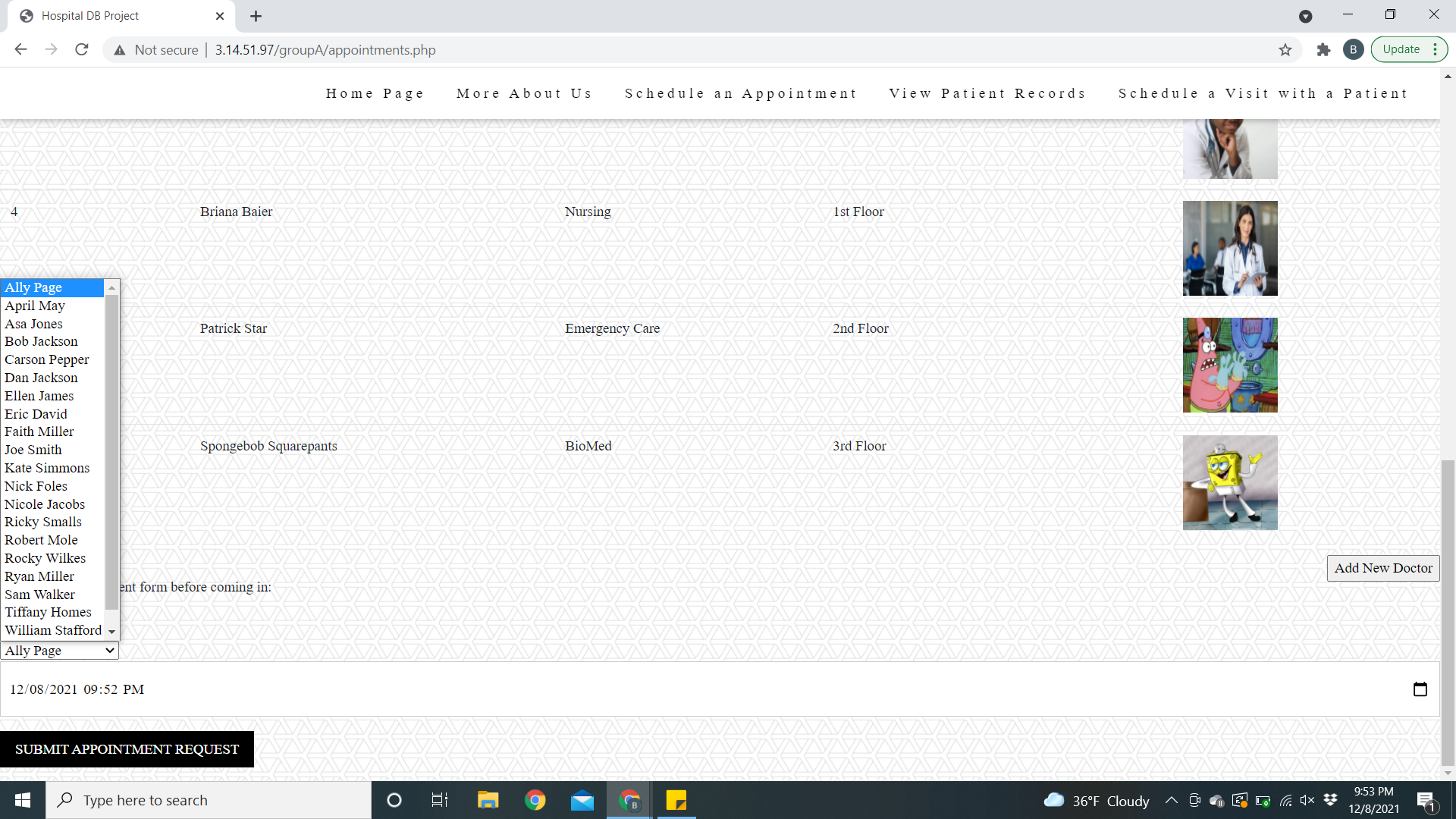
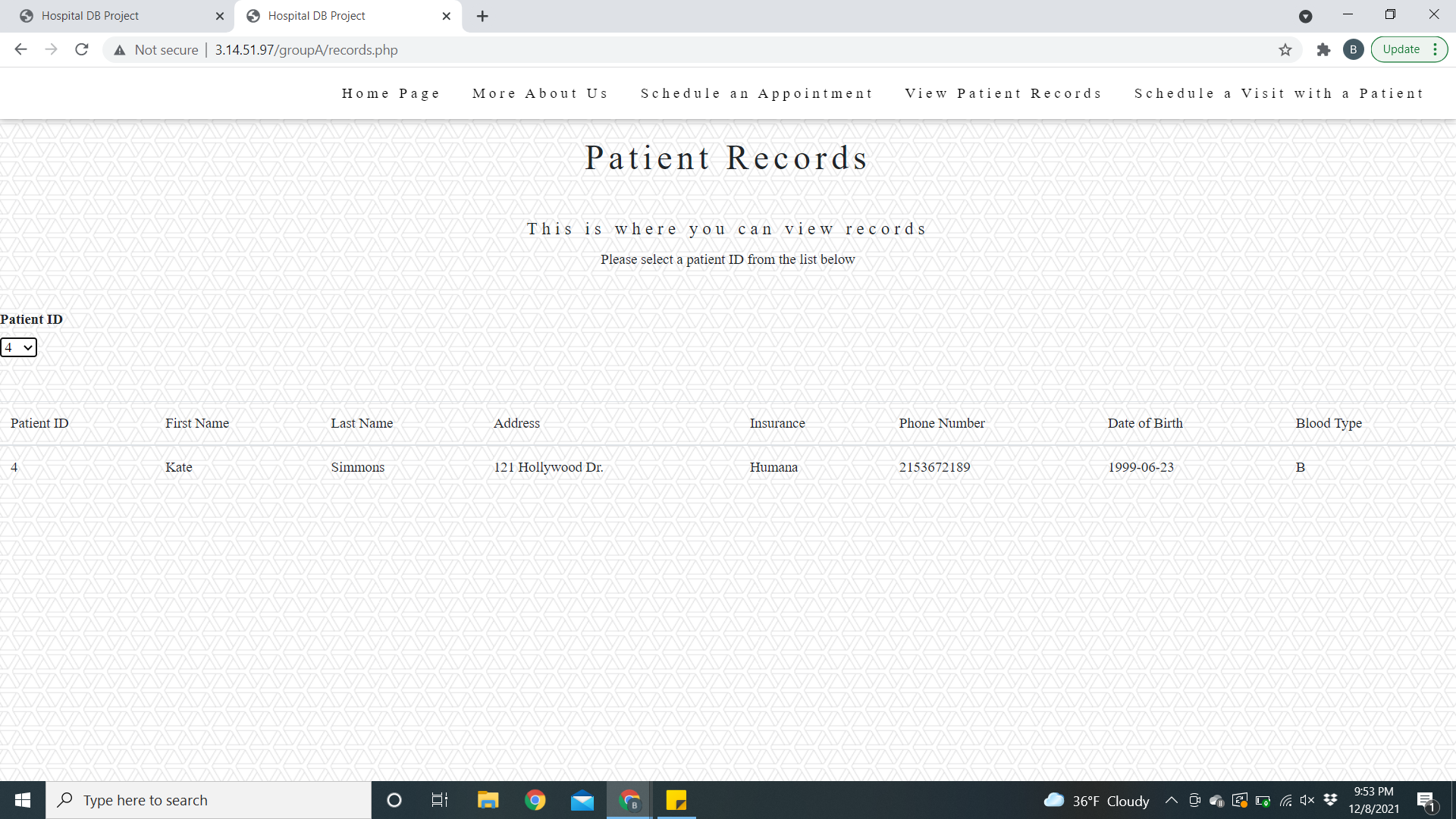
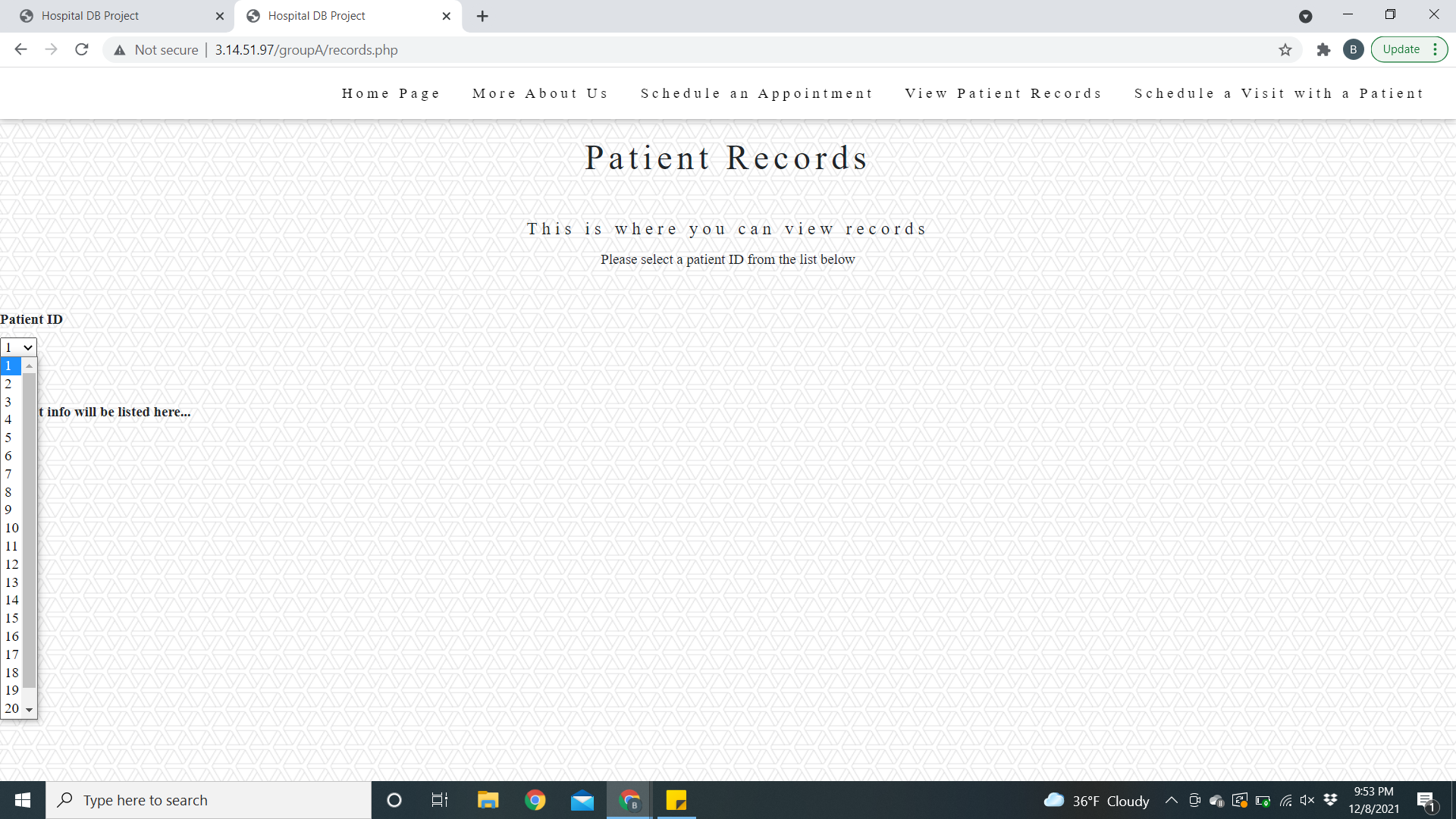
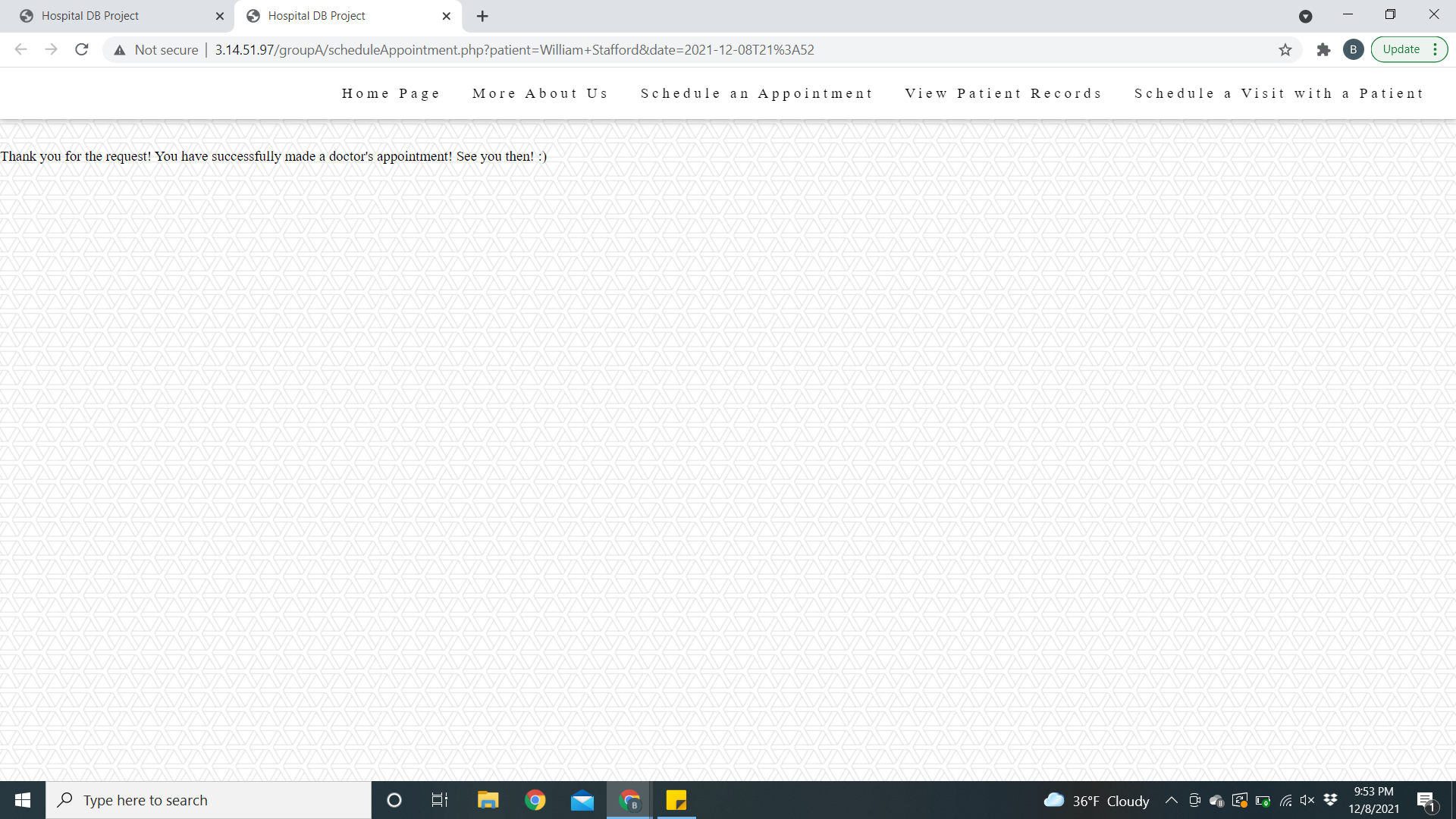
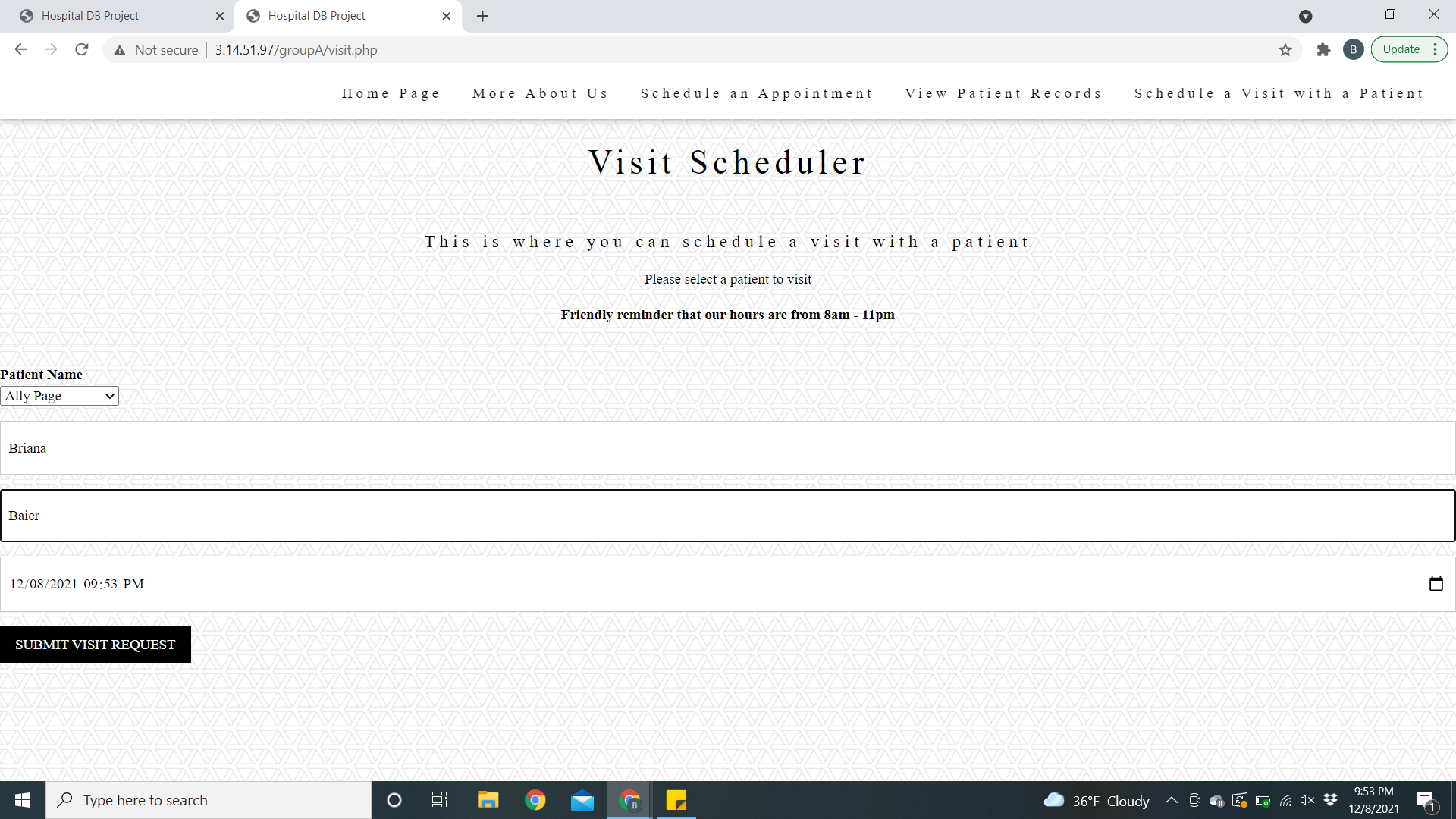
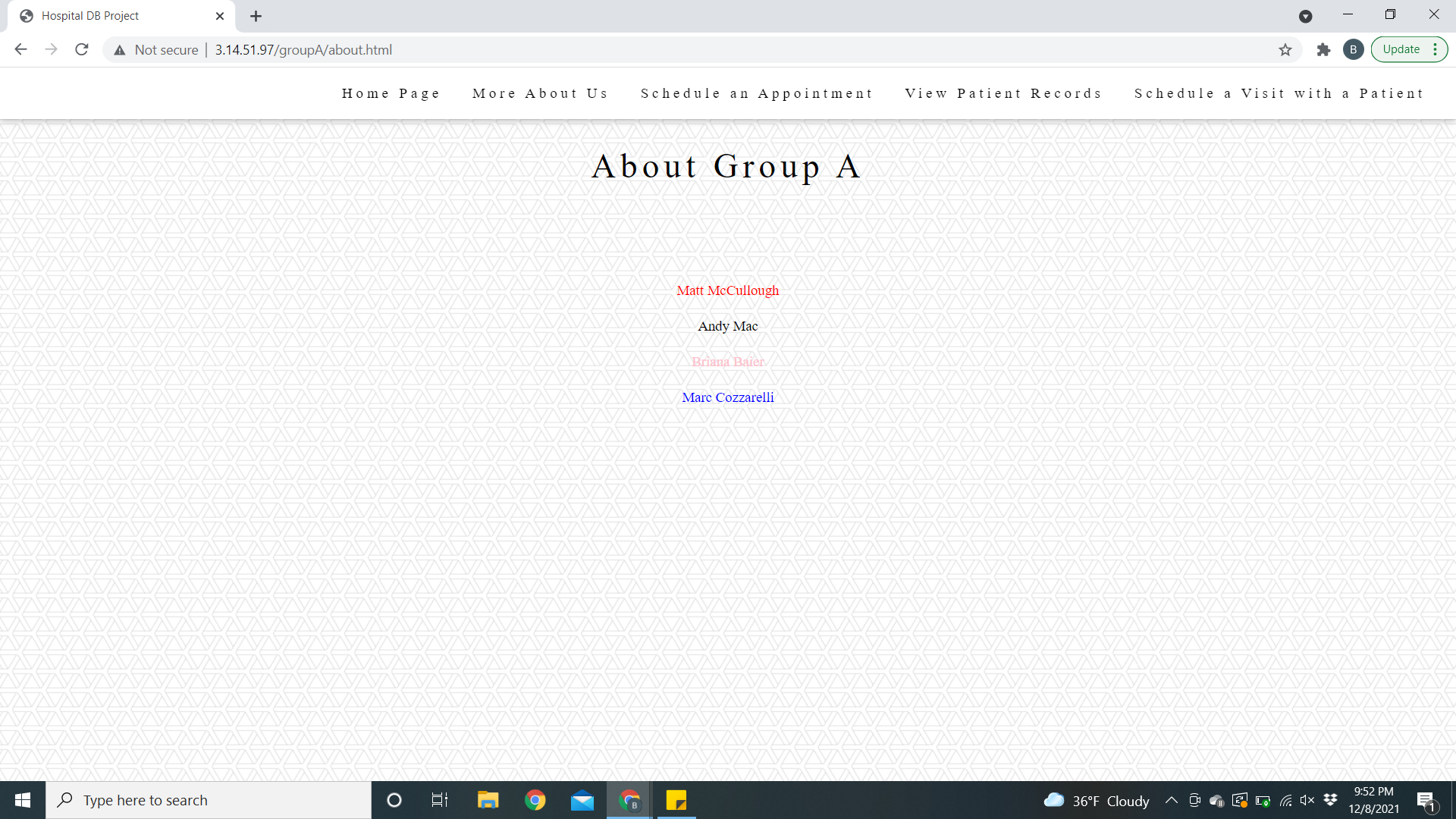
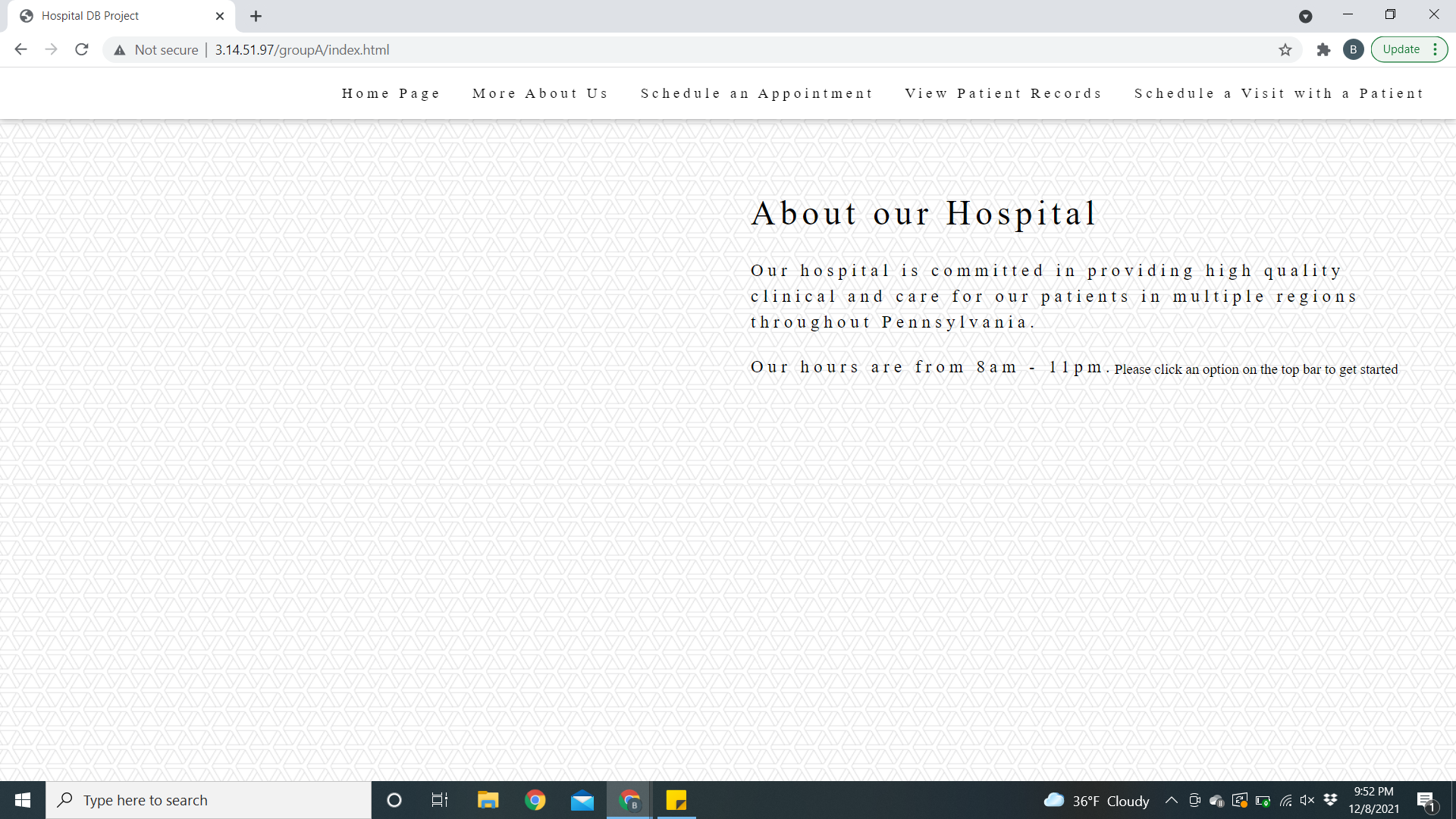
As mentioned previously we executed the extra credit of creating procedures and triggers. The triggers that we implemented were to check if the data was valid when a user is attempting to insert it into the table. Two of our triggers are to make sure that the times an appointment or a visit with a patient are scheduled aren’t able to be double-booked. We also have a trigger to ensure that the patient that a visitor is trying to schedule a visit with is actually in the database and is available. Finally we have a trigger that relates to integrity constraints that will not allow duplicate entries for patient names, this also helps with the ease of coding in PHP and with scheduling patients so that there are no mixups. In order to maintain that integrity of our data we don’t allow for multiple appointments at the same time and we also auto incremented the patient IDs in order to prevent people from choosing the same ID as another patient.

In the creation of our relational schema we had to consider what kind of business rules are involved based on the data that we would be using and the entities relationships to one another. Our first rules had to do with the tables and their relationships, basically where the foriegn keys go and how these entities will interact with one another. We came up with the following list of business rules to use when creating our relational database schema. The rules are as follows: can’t have a visitor past hours, only 1 patient can visit doctor at a time, only 1 visitor to patient, a doctor can belong to one department but a department can have many doctors, a doctor can have many patients and a patient can see many doctors, a patient must only have one record and a record can only belong to one patient. We found that these business rules helped us maintain the quality of our information that we were storing in the database.

Below is our EERD we created using WorkBench:



And here are some screenshots showing our live demo:



Conclusion:

This project enabled us each to gain exposure to database development, management, servers, php, and teamwork in these contexts. Through starting from scratch with coming up with an idea and then implementing that idea both from the front end and the back end we were able to gain valuable exposure to what the real world industry will be like. In this project, the php and web page development was mainly done by Matt including the linking between the DB and php together, loading the images, and creating the triggers. Andy also helped with some of the PHP code and styling the website, created some of the entities in the DB, inserted the test data, and helped with a lot of the debugging of the website. Briana created tables, modified tables, wrote a trigger, wrote business rules, and wrote the report for the project summary. Marc helped write some code for the DB, created the final presentation, wrote business rules, and created the project task tracker. Our group had great communication throughout the process and is very satisfied with the end result of our work.