Title: Homecare

Created by:

- Aaron Asibbey
- David Ustyan
- Jack McKinstry
- Nate Weaver
- Nick Henley

Project Description:

Our Application is an interactive app to pair privately practicing nurses with patients in need of care at their own homes and their families.

Patients' family members are able to view notes from when nurses check in. That way, they can ensure the nurse is visiting their family member that needs care. They also can view their currently prescribed medications and stay in the loop on their overall care.

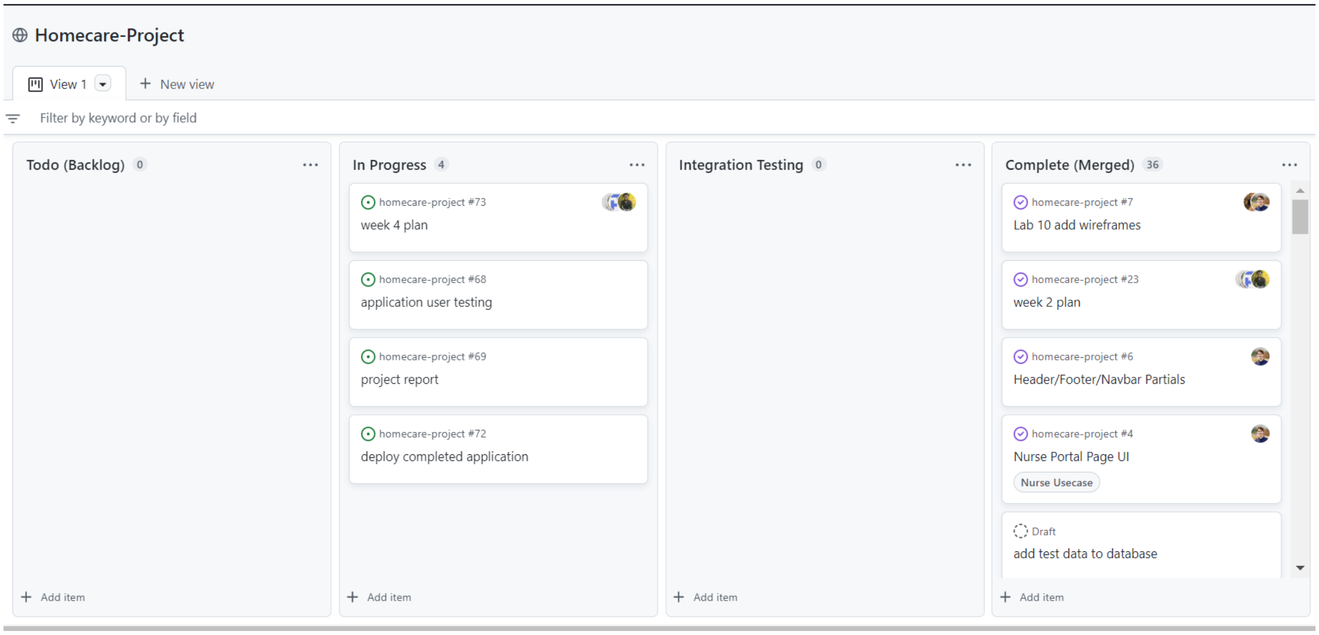
Nurses are able to open an overview for each of their patients. This includes information such as medication, schedule for check-in’s, notes and special circumstances for patients, and notes on prior check-in’s from them and other nurses. They also can post an update for when they complete a check-in on a patient.

The super-user is an administrative account that creates new users for both nurses and patients, assigns nurses to patients, assigns medication to patients, and views all patients. This user is meant to be the owner of an agency and manages data of the entire agency.

This project will improve relations between family, patients, and nurses; improve organization of agencies; store patient data in a unified place as well as easily display it; and overall improve the care that nurses can provide.

Project Tracker:

Link to project tracker: here



Homecare-Project

View 1 | New view

Filter by keyword or by field

Todo (Backlog) 0

Add item

In Progress 4

homecare-project #73
week 4 plan

homecare-project #68
application user testing

homecare-project #69
project report

homecare-project #72
deploy completed application

Add item

Integration Testing 0

Add item

Complete (Merged) 36

homecare-project #7
Lab 10 add wireframes

homecare-project #23
week 2 plan

homecare-project #6
Header/Footer/Navbar Partials

homecare-project #4
Nurse Portal Page UI
Nurse Usecase

Draft
add test data to database

Add item

Video:

The video demonstrating our project can be found here in the repository.

VCS:

Version Control System repository:
https://github.com/jackmckinstry/homecare-project

Individual Contributions:

Nick Henley My contributions include the patient portal, sessions, database, and admin portal. I was responsible for the entire patient portal page, creating the UI and backend integration. For sessions, I chose the permission and user id variables that are stored in the session. I created the original database design that was perfected by the other group members. Finally, I updated the admin portal to be more consistent with the design of the rest of the website. I used docker to run the site, ejs and bootstrap for ui, nodejs for the backend, and SQL + Postgres for the database.

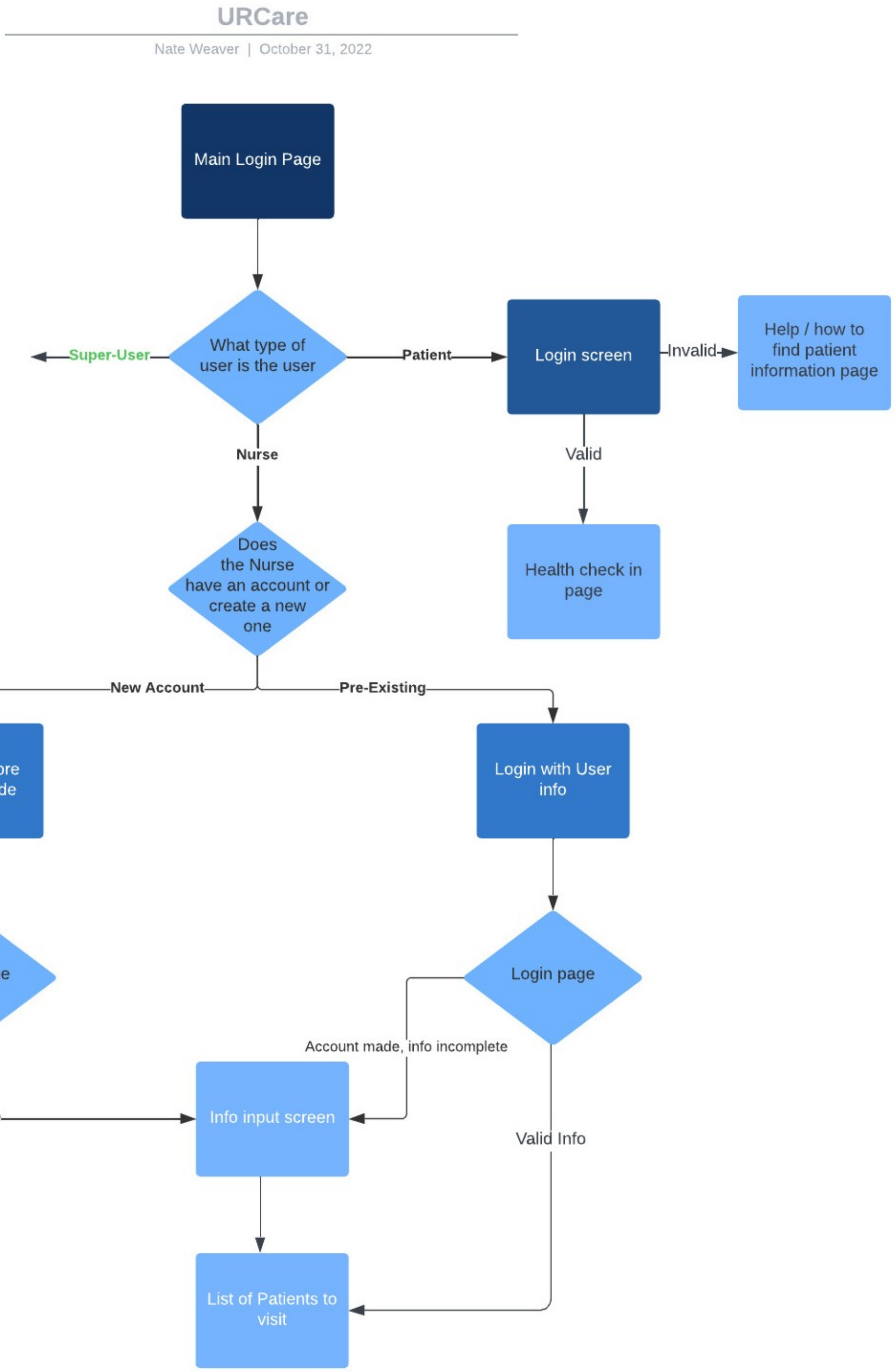
Nate Weaver In this project I worked on lots of technologies and made a good amount of contributions to the project. One of my main contributions was building the home page and adding some of the functionality to it including helping on the menu, header, and footer. Apart from these additions I made a functioning carousel for the home page using JS, CSS, and HTML/EJS. I also was tasked with the job of hiding menu functionality when you have certain permissions. For example the logout should be hidden if the person isn't logged in. This was pretty hard and took up a lot of time for me. Other than that I had done some small changes in the UI and in the front end side of things.

Aaron Asibbey I was able to contribute a lot into the overall idea and structure, superuser UI and functionality, and login UI for the project. One of my main responsibilities was creating a superuser UI and making sure it was able to redirect to different portals that an admin would need to be able to do such as add medications to patients, assign nurses to patients, and add new nurses and patients. This required more EJS pages having to be created, JS, SQL, CSS, HTML, Bootstrap, and etc. I also worked on the UI for the login and making it more professional using skills in CSS. I introduced the project idea through inspiration of a home health company my mom owns that I used to work for, so i was able to use real world knowledge to implement the features and ideas that made this idea realistic in terms of the what a nursing home health website would need as well as which new ideas we could add to ours to differentiate it from similar existing websites.

David Ustyan My primary contributions to the group project were the login and register page functionalities. I created both login and register using JS, SQL, CSS, EJS and Bootstrap and linked them to the database so that when an account is registered it updates and adds to the database and also used sessions so that someone can't log in unless they are logging in as a registered account that they just registered. I did this by using sessions so that when an account is trying to log in, it cannot do so unless there is an open session for it, and registering an account creates the session, allowing somebody to log in with those credentials.

Jack McKinstry I created the UI and implemented the backend features (method endpoints and writing SQL queries) for the functionality of the entirety of the nurse portal page utilizing HTML, Bootstrap, EJS, NodeJS, and PostgreSQL. Using the same tools, I also implemented the “assign nurse to patient”, “view all patients”, and “add medication” features in the superuser/admin portal. Through implementing these features, I restructured parts of the database tables. I also utilized the sessions management in order to require valid permissions to access certain methods (pages). Additionally, I did a significant amount of debugging in the final integration of the project.

Use Case Diagram:


URCare
Nate Weaver | October 31, 2022
Main Login Page
What type of user is the user
Super-User
Patient
Nurse
Login screen
Invalid
Help / how to find patient information page
Valid
Health check in page
Does the Nurse have an account or create a new one
New Account
Pre-Existing
Login with pre existing code
Login with User info
Login Page
Login page
Account made, info incomplete
Valid code
Info input screen
Valid Info
Help page / Info page about how to be registered
List of Patients to visit

Test Results:

For information on our user acceptance testing plan, see here

- **What are the users doing?**

  The users interacted with the software properly, they were able to comprehend the way the servers worked, the notes sections and could successfully predict the result of their actions before it happened.

- **What is the user's reasoning for their actions?**

  It was intuitive to find the login button and were able to easily identify the the flow of the websites design

- **Is their behavior consistent with the use case?**

  Yes it was very consistent with what we had intended

- **If there is a deviation from the expected actions, what is the reason for that?**

  The one thing that left the tester confused was the switch patient function on the nurse portal page. This feature is slightly weird but definitely would make sense with more use of the website.

- **Did you use that to make changes to your application? If so, what changes did you make?**

  We did not use this as means to change the website's design as the design we had implemented works similarly to it would in an ideal world. As said before though this may not be the most intuitive process, it is one that works and as previously stated will become more intuitive with use

Deployment:

How to run the application locally:

1. Clone the repository with `git clone https://github.com/jackmckinstry/homecare-project.git`
2. Navigate into the `/homecare-project/project_components/` directory
3. From a shell within the `/project_components/` directory run: `docker-compose run npm install` on the initial run in order to install node modules OR `docker-compose up` on all starts following the initial
4. Wait for Docker to print `Server is listening on port 3000` and `Database connection successful` in the shell terminal
5. Navigate to `localhost:3000` in any web browser

The application was also deployed to the CU server and can be accessed here while connected to the CU Boulder wifi network.