Homework 6

11/13/2012

CS 361

Simplifying Nutritional Tracking with QR and Bar Codes

Customer: Kevin Stephenson | Group Members: Colin Bradford, Kyle Cesare, Emerson Hovekamp, Taya Juenemann, Tudor Marcu, Peter Rindal, Joshua Vilwock, and Nick Pepperling

Project Information

What is the URL where your software can be tried out or downloaded? Describe any special instructions for using or installing the software (e.g.: Firefox browser only, Windows only, JRE must be installed, etc.) (approx 1/2 page)

Website URL:

http://web.engr.oregonstate.edu/~rindalp/foodSite/index.php

Special Instructions:

- No specific browser requirements.
- In order to access the full functionality and all features of the website, you will need to register (create an account) and log in.

The User Stories we chose to Implement:

1. User creates an account (personal profile with their information that is saved so that the user doesn't have to re-enter it).
2. User can search for a food item. (by ID number)
3. User (or a restaurant owner) can upload a food item to the database.

User Stories Breakdown

For each user story due today, answer the following (total for all user stories due today, approx 2 pages):

- *Which pair(s) of teammates worked on that user story's tasks?*
- *What do the relevant unit tests do?*
- *What problems, if any, did you encounter?*
- *How long did each task require?*
- *What is the current status (implemented? tested?)*
- *What is left to be completed?*

Generate Sequence of Numbers (Barcode)

- Worked on by: Peter and Josh
- Unit tests:
  - Ensure that only users with correct permissions may generate numbers
  - Ensure that invalid data is rejected
  - Ensure that duplicate data is rejected

- Problems: Unfortunately, we were not able to fully implement the client's idea of scanning QR codes. Given our resources, we had to talk to the client and simplify the requirements by changing the scope of the project. We decided to go with the simpler approach of allowing users to type in an identification number, rather than having to delve into computer vision to be able to read a QR code from the camera.

- Time: ~4 hours

- Current Status: This feature is currently complete. Users must be registered and logged in to enter new foods. On the Food Upload screen, they are presented with a form of the food name, description, number of calories, and an option to upload an image. Upon submitting the food, they are taken directly to the food's page to view their submission.

- Remaining tasks: This feature has remaining tasks, unless the client requests changes. Foreseeable changes may include UI changes to more closely adhere to the client's original vision. He may also want to enter either more or less data about the food, depending on the amount of data he wants to display to the user. In future user stories that require this data, we may also find that we haven't stored enough characteristics to calculate values we need (e.g. calculating caloric requirements based on activity levels).

Search by Number

- Worked on by: Peter and Taya

- Unit tests:

    - Ensure that searching a food ID returns the correct food
    - Ensure that searching an invalid food ID will return an error

- Problems: Aside from the problem mentioned in the story above, we had no major problems with implementing this feature.

- Time: ~5 hours

- Current Status: This feature is currently completed. A user enters the index page and is presented with a prompt to enter the food ID, which they will find on the food product they are interested in. They enter the ID and Submit, which brings them to the food page. On the food page, they are presented with the food name, an image of the food, a short description, and the number of calories.

- Remaining tasks: We still need to review this story with the client to ensure it adequately meets his vision. Right now, we are unsure how much information he wants to be displayed on the food page. For instance, users could be presented with full nutritional facts, including fat content, sugar content, etc. However, this may be defeat the simple nature of the app.

Enter Personal Information

- Worked on by: Peter and Nick
- Unit tests:
    - Ensure that unrealistic user data is not allowed
    - Ensure that users cannot create duplicate accounts
    - Ensure that data is saved to the database upon submission
    - Ensure that reports are regenerated when data is updated
- Problems: The vision statement was a little vague on this feature, so we had to piece together what pieces of information the client wanted to be entered and stored.
- Time: ~6 hours
- Current Status: This feature has been fully implemented.
- Remaining tasks: There are a couple of possible tasks that may have to be added depending on client input and any future problems we run into. The client may want more information to be made available to users. We may also run into a case where we need more information about users to accurately track their health. For instance, we may need them to expand on the 'Activity Level' field, beyond just a single number describing how many hours of activity they have per day. This all depends on the future stories we decide to implement.

Diagrams Usefulness

For each spike and UML sequence diagram that you developed this week, answer the following (total, approx. 1/2 page):

- *Was the spike or diagram useful? Why or why not?*

Our UML Sequence diagrams were very helpful to us in visualizing the design of each of the user stories that we implemented. Using the UML Sequence diagrams that we created, we were able to clearly identify and understand the structure as well as the flow of the user story that the diagram was depicting, which helped us to implement it with thoroughness. The UML Sequence diagrams were also a good means of communication between the group members of our team, to make sure that everyone understood the structure or each user story. It was good way of visualizing what we needed to accomplish, rather than verbally discussing it, where it is easy to misunderstand your fellow group member or forget what was said at the last meeting about the user story and what in particular needed to be accomplished, and in what order. We plan to continue to utilize UML Sequence diagrams in the future. This week, our group discussed and began diving into developing a spike for seeing how difficult it would be to create and integrate an algorithm for giving each user a specific and personalized health and nutrition information based on their profile.

Diagrams Needed

Were there any diagrams that you wish that you had? Why or why not? (approx. 1 page)

Before answering this it is important to note that since that after a meeting with our customer, we agreed that implementing this in the form of a mobile app was unrealistic for the time and resources we have available between now and the deadline. Instead, we collectively chose to make a website, making the selected user stories far easier to complete. With that being said there were a few documents that would have made the implementation process a little bit easier.

- **Paper Prototype**

  Once we had the user stories selected, and a general team-wide approach planned out, we jumped right into coding the website. Peter had a well-fitting template from a web-site he built in the past which took care of standard page structure, but without paper prototyping it was a lot of back and forth "guess and check" work. If we had a nicely organized paper prototype to begin with, formatting the layout of button, links, entry fields etc... on the pages that we added would have been a bit more seamless. These details were more cosmetic than functional however, so it did not get in the way of us finishing our user stories.

- **More Sequence Diagrams**

  Our website draws heavily from back-end database access so knowing the sequence of events is important when linking the site to the database. In Homework 5 we specified a few simple sequence diagrams to accommodate our user stories, but it would have been beneficial to have a few more. I don't think that the absence of a few extra sequence diagrams slowed us down at all, however it would not have hurt to have some extra clarification.

- **ER Diagram**

  As previously mentioned, our website draws heavily from database access. We already have multiple tables being queried, and throughout our next week of implementation we may even add more. An ER diagram will be very beneficial if our database increases in complexity. The reasons for this are clear. A well-defined diagram showing the relationships between entities is critical for successful database management.

Refactoring

Briefly describe any refactoring that you did (approx. 1 page)

We have repurposed code from Peter's and Josh's past work. Josh has done some code cleaning on several occasions, making small changes to improve the aesthetic. When new php code is written, links are added to the existing page to make the code accessible. Specifically, that has been done at least twice so far: once for the food search php, and once for the food lookup. The registration php was refactored to include the entry of additional information. Related to that, the user info database object was refactored to include the new fields.

Our documents have seen some refactoring as well. The definitions of each of the user stories yet to be implemented have seen some small changes over the course of implementing the project thus far. With these changes, our project timeline has also been adjusted as necessary. When we get things done depends on the definitions and requirements of each one, but also it depends on what the customer has specified as the most important features of the project.

In the future we may continue to do more refactoring when needed. Some probable future refactoring may include: more links added to web page base as more functionality is added, as well as additional changes to user database entity, as related to goals and history.

Working with the Customer

If you had to ask the customer any questions, indicate what those questions were and what the customer's response was. Briefly describe any surprises or requirement changes that you encountered. (approx. 1/2 page)

Our team did have several questions for our customer, because we felt it was important to discuss our progress with our customer to make sure we were on track and creating the system that the customer was envisioning. Below are the questions that we asked our customer and his response to each.

1. Is it acceptable for implementation (or do you have any recommendations or comments about) having the user type in a food identification number instead of scanning a barcode/QR code? (Our team has discussed how difficult it would be to actually implement the code scanning and so we decided to assign each food item with a ID number.)

   Yes that works. I agree that the actual QR part is a lot of work for a 1 week team assignment. I think that's the most reasonable alternative.

2. How extensive/what do you want the food item attributes to be? Right now we have each food item having a name, a picture, # of calories, and a description.

   I was basically thinking of including almost all information that you'd find on a nutrition label. (Vitamins, ingredients, carbs, etc.) But the information you included works for the first implementation. The things you have are the most critical.

3. How extensive/what do you want the personal profile attributes to be? Right now we have last name, first name, gender, height, weight, and physical activity level (on a scale from 1 to 10).

   Those cover most of the attributes I could think of. Other options could be body fat % and blood pressure, but those are extras.

4. What steps do we need to take to make the system more personalized to each user?

   I logged in and the web page looks good! My only real suggestion is to have the links at the top pop out more. You could just make the background of the list a gray or something simple like that. Maybe the user could specify goals like a certain cap of calories for the day and it could display how close they are?

5. Based on what we have right now, what do you think would be the most important next step(s) for implementation?

   Maybe find a site or something that allows you to create QR codes and imbed the item id in it. That could be the user story for the restaurant people wanting to create the QR

codes. Another important step is to allow the food item to be added to your... cart? One of the most important features is for your nutritional information to be tracked over time so you can see your progress for the day / week / month. I think the easiest way to do this is to have a new table with the user ID, food ID and date/time food is consumed. Date/time as primary key probably? Then when a user wants to see their progress for the day, have it search the table for food items that match that user ID and window of time.

After reviewing the customer's responses, we adapted our requirements and project plan in accordance with them if needed. There were no real surprises; we generally expected these responses. Our customer was very helpful in giving us insight on his vision of the project which in turn helps us to sharpen our implementation even more.

Integration Tests

Briefly describe all integration tests that you did on the system, the test results, and any changes that you made (or will make) to the system as a result. (When you do these tests, don't be soft. Do a good job -- your goal is to build an awesome system!) (approx. 2 pages)

Integration Tests Notes:

Register form accepts numbers over 10 as an option and not all the fields are validated.

- The information is then saved as 11 in the personal information.

Updating personal information does not work - php error().

The title seems to truncate to a certain length, as does description, but the description goes way off the page if it's too long; it needs to be put into a div or some container to wrap it.

- There is no written limit on how much you can type;
- There should be validation for the limit and what you put into the fields. (I.e., I can write words into calories and it will take it just fine and put 0 since it is not a number.)

If the file is too big the site will just revert back to the "upload food" page and not actually upload anything. There is no error or warning and the site takes the file. It may have been uploaded or not, but there is no way to tell.

- It also accepts any file type as long as the file is not too big, which is probably the biggest
- Security hazard besides giving someone your password.

Important: Needs Form Validation!

Details:

We ran some integration tests on the system to see how it reacts to both a typical user and the errors they may make, as well as someone who was purposely trying to break the system. The biggest problem we encountered was form validation, which is lacking on many parts of the website. The first thing I noticed was validation on the registration form. When registering, the form accepts numbers over 10 as an option to the activity level, and saves the information as 11 in the personal information section. The form also accepts letters in the fields where there should be only numbers, in which case it either saves them as -1, the letter, or 0. Fixing this would simply require some more form and variable validation; I.e checking if the input is a number or not, and if it is in the range that is allowed for the given value. Another option would be to replace different input fields like activity level, height, weight, etc. with select tags (select tags: http://www.w3schools.com/tags/tag_option.asp).

Updating personal information just does not work. It fails with an error in the php code, so there was not much testing we could do on that front. Of course, this cannot be left in the production code/version, as an error like that would take our entire product offline or render it useless. We will have to go through the code and make sure everything is working and accessible, and perhaps even have some backups in case some connections go down.

The food search page seems to be working well, and will respond with a proper and meaningful error if the food is not in the database, so we don't think there is anything to fix here in our opinion.

The food upload page has several issues associated with it, some minor, some more serious. The first is that the title and description truncate to a certain length, which seems to be the size allotted for the entity in the database. However, no warning or instructions are given on how much a user is allowed to type in either field, and it simply accepts any type of input (no matter how long/large), and truncates when saving. The description also goes way off the page bounds because it is not wrapped in a div or wrapper, but that is an easily fixed styling error. To fix the input errors we again need to do some form validation and error or show a warning at least when the input is incorrect. We also should provide some instructions like "160 chars max" for description and "20 chars max" for title, just as an example. The bottom where it asks for calories also is not validated and takes letters, which are then interpreted as 0 in the table.

Now the major part of the upload page is more concerning and should be fixed before anything else on the site: the upload image. We were able to upload literally any type of file, with virtually any file size to the site/server. I attempted to upload a 90mb exe, which it says uploaded, but then revered back to the upload food page and did not actually upload anything (or so it seemed). The file may be on the server, but no error or warning was given, which leads us to believe it was just uploaded but not added to a database or parsed correctly. This is a major security risk as anyone can upload anything, thus it is leaving a wide open door for attackers and intruders. We MUST validate this form and check the extension of the file being uploaded, and then check the file size and only allow up to a certain number of kilobytes or so.

Overall the main errors we encountered with the system were due to validation not being implemented, and would easily be fixed and yield a stable and smooth system in the end.

New Schedule

What is your new schedule for what to complete next week? What user stories will be done, what is your time estimate for each of these user stories? (approx. 1 page)

Our first implementation went smoothly, and yielded a functional website with back-end database access. This was an important precursor to the bulk of the system's vision: to allow users to track their nutritional intake with an easy-to-use system. I am not certain there will be enough time to implement every detail that the customer had in mind, but we have selected the tasks that we (our group and the user) feel are critical to basic system functionality, and also reasonable. The tasks are as follows:

User can search food items table by food name:

The original vision from the user was to have a mobile app that could scan QR images and/or bar-code of any food item they eat, and it would instantly add the nutritional information that they consumed to a daily counter. Since that didn't seem realistic we instead created a table for food items whose primary key is a number. This number is synonymous with a QR code in the ideal implementation. Currently, you can only search by this unique food ID; however, since the user is not scanning something to produce a number for a desired item, we need to add an alternative way to fit the new system spec. Thus, we will need to implement a food search by name. (4 hour)

Function to calculate a user's basal metabolic rate for personalized diet suggestions:

Each time a user signs up, he/she enters personal information, all of which is stored in a table and a user profile is created. At this point, our user profile is not sophisticated enough to implement a BMR function, so we will first need to add fields for age, diet, and anything else that may contribute to someone's BMR to our users table. Then we will write a function that will add a BMR to each user's profile which can be used to calculate recommended calorie consumption. (3 hours)

Users and food items need to be connected so a user can "eat" and item and have its nutritional information added to user's daily intake.

The underlying purpose of this system is to allow users to track their nutritional intake. In order for this to happen we need to link the two tables and allow users to figuratively "eat" a selected food item. (3 hours)

In addition to implementing these new tasks, the customer has requested that we update that attributes of each user and food item. (i.e. adding more nutritional value attributes other than just calories to food items) (1 hour)

If we manage to finish this wave of implementation early, the user has directed us to libraries that generated qr codes, and suggested trying to fully implement a scanner somehow. This would be a tall order and its legitimacy is to be determined. (6 hours)

Final Notes

In one sentence, briefly summarize whether your customer was willing and able to talk with you by email in a timely fashion. If you do not receive a response within 24 hours, proceed as best as you can without customer input.

Yes, our customer Kevin Stephenson was willing and able to talk with our team via email. He was quick to respond, providing both a quick response and then a more in-depth response later that day. We appreciate his helpful feedback on our project, because it allows us to make sure we are on track with the system purpose and goals as we move forward with the implementation.

Briefly summarize the contribution of each of your team members.

- For pair programming/implementation contributions, see "User Stories" section above.

Homework Sections:

1. Project Information - Peter
2. User Stories Breakdown - Kyle and Josh
3. Diagram Usefulness - Taya
4. Diagrams Needed - Nick
5. Refactoring - Emerson
6. Working with the Customer - Taya
7. Integration tests - Colin and Tudor
8. New Schedule - Nick
9. Final Notes, PDF compilation/submission - Taya