

Christopher Soto

Apple-tite

Academy of Scientific Research

Lincoln High School

Dr. Saghafi

2015 - 2016

Title of Project

Appetite

Field of Study

Invention

Abstract

The purpose of my study was to create a solution to help benefit the world's older population that is projected to be 1.6 billion globally by 2050. In the United States alone from 2013 to 2014 government spending on health care services increased by 5.3 percent and reached a cost of \$3.0 trillion dollars on health care services. As a solution to help benefit the world's older population in the field of health care services, I developed a mobile application called Apple-tite. Apple-tite is a mobile application that provides healthy food recipes and financial information for each recipe, as well as connects healthcare professionals and patients through a system that I created called the CareID system. Apple-tite was created using the software Intel XDK and the programming languages of HTML, CSS, Javascript, PHP, and SQL. Apple-tite is also connected to a MySQL database using Ajax which allows for communication between Javascript and the backend of the mobile application (PHP, SQL, and the Database). Apple-tite is unique because it has multiple functionalities that most health, financial, and medical applications on the Google Play store and on the iTunes store don't offer. Apple-tite also has various functions such as allowing users to add recipes to their favorites list, connect with caregivers / professionals, and have the ability to regulate their own medical information. If the user is connected with a caregiver / professional, the caregiver / professional has the ability to regulate the user's medical settings. Apple-tite also works on all mobile operating systems such as Android, iOS, and Windows.

Introduction / Background

According to the United States census from 2025 to 2050, the world's older population is projected to double to 1.6 billion globally [1]. The increase in the older population throughout the world, has led to an increase in government spending in health care services. The Centers for Medicare & Medicaid Services in 2014 stated the cost of health care for the population of just the United States alone from 2013 to 2014 increased by 5.3 percent and was projected to reach \$3.0 trillion dollars [2]. This problem is not an issue that the United States alone is facing. There are many other countries that are also dealing with this problem as well as countries that have a population that are underrepresented and don't have access to health care services. As a solution to this problem I created Apple-tite. Apple-tite is a mobile application that provides healthy food recipes, financial information for each recipe, and as well as connect healthcare professionals and patients through a system that I created called the CareID system. Apple-tite is unique and different than any other mobile applications on the Google Play store and on the iTunes store because Apple-tite has different functionalities such as the CareID system that allows healthcare professionals to regulate the recipes that are shown within the mobile application depending on the patient's medical record. Apple-tite also has other functionalities such as allowing patients to regulate their own health care information within the mobile

application if they are not already connected to a health care professional. Apple-tite also allows users to add recipes to their favorites list and has a registration system built within the mobile application that runs on a MySQL database. To create Apple-tite I used Intel XDK which is a software that allows individuals to create mobile applications using the programming languages of HTML, CSS, and Javascript. The process is very similar to creating a website. Intel XDK has a compiler in which it compiles the source code and converts to run on multiple mobile operating systems such as Android, iOS, and Windows, because of this Apple-tite is compatible on all mobile platforms.

Experimental Description

As mentioned above I used the software Intel XDK to create Apple-tite. When creating Apple-tite on the Intel XDK platform, I later realized that the software does not allow for compatibility with any other programming languages besides HTML, CSS, and Javascript [3]. The design and structure of Apple-tite revolves around using a database to save the user's information on multiple devices and as well as allow for the connection of healthcare professionals and patients through the CareID system. I decided to use Asynchronous Javascript (AJAX) as a solution to my problem. AJAX allows for the requesting of information from the server without having to reload any pages and as well allows for Javascript to receive and work with the data from the server [4].

I established a connection between AJAX and a PHP document hosted on my own server that would allow for the receiving and sending of data through this PHP document.

```
function sendPostData(action, data, callbackFunction){
    var xhttp = new XMLHttpRequest();
    xhttp.withCredentials=true;
    xhttp.onreadystatechange = function() {
        if (xhttp.readyState == 4 && xhttp.status == 200) {
            callbackFunction(xhttp.responseText);
        }
    };
    xhttp.open("POST", "https://christophersoto.me/app/php/portal.php?action="+action, true);
    xhttp.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
    xhttp.send(data);
}
```

As shown above I am establishing the HTTP connection between the two files.

AJAX is awesome and also allows the ability to specify what is being sent, as shown below.

```
function toggleDisease(careid, disease){

    var toggleResponse=function toggleResponse(data){
        refreshDiseases();
    }

    sendPostData("toggleDisease",
        "id="+encodeURIComponent(careid)+"&disease="+encodeURIComponent(disease), toggleResponse);
}
```

By using AJAX I now had the ability to go around Intel XDK's limitation of only allowing HTML, CSS, and Javascript within the software. AJAX now allowed for the ability to use programming languages such as PHP and SQL, thus now allowing for the use of the MySQL database which will save the information from the mobile application.

Registration / Login

As mentioned in the introduction, Apple-tite has many different functionalities which are built within the mobile application. One functionality is the registration and login within the mobile application which allows for users to log onto any device and have their information saved and as well as their favorite recipes and as well as their health care information.

```
function register($db, $username, $email, $name, $password, $password2, $caregiver){
    $errors=array();
    if(!ctype_alnum(str_replace(array('-', '_', '.'), '', $username))) array_push($errors,"Invalid Username! Use A-Z,a-z,1-9,-,_,. or .");
    if($username<=20&&$username>=5) array_push($errors,"Username must be between 5 to 20 characters");
    if(username_exists($db, $username)) array_push($errors,"Username already is registered to someone else.");
    if($password!=$password2) array_push($errors,"Passwords do not match!");
    if(!filter_var($email, FILTER_VALIDATE_EMAIL)) array_push($errors,"Invalid Email!");
    if(email_exists($db, $email)) array_push($errors,"Email already is registered to someone else.");

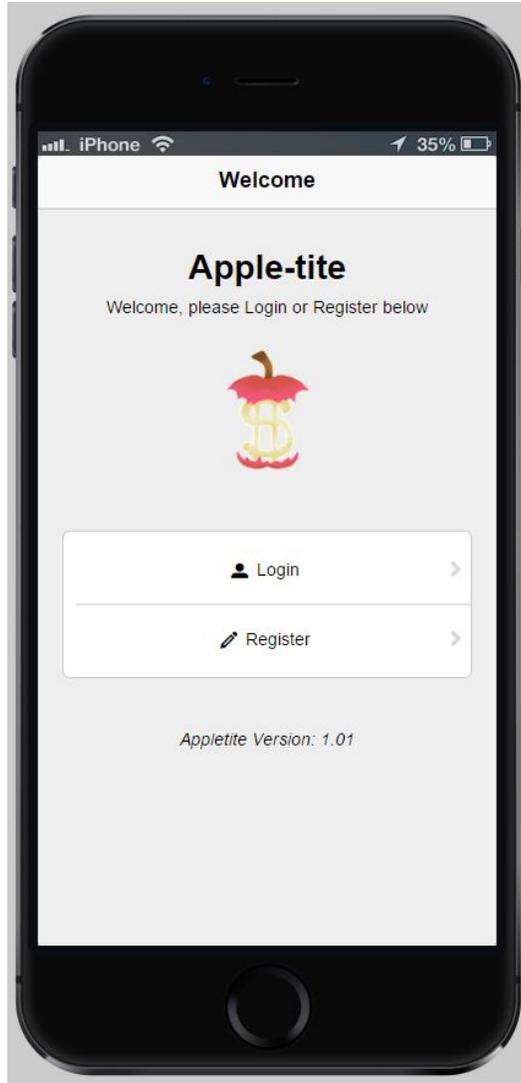
    if(count($errors)>0){
        return $errors;
    }

    $caregiver=($caregiver=='yes'?1:0);

    $stmt=$db->prepare('INSERT INTO users SET username=?, email=?, name=?, password=?, ip=?, caregiver=?, careid=?');
    $stmt->bind_param("sssssii",$username, $email, $name, password_hash($password, PASSWORD_DEFAULT), $_SERVER['REMOTE_ADDR'], $caregiver,generateCareId());
    $stmt->execute();
    return true;
}
```

As shown by the code above, this is just a small part that went into creating the backend of the registration for the mobile application. The programming language shown above is PHP.

The example of PHP code above is just showing the criteria that I took when creating the mobile application. Since the target market of the mobile application is to individuals who are at the age of 65 or will be 65 or older, I incorporated different errors that they may experience within the registration page. Throughout the design of the mobile application I also took this into consideration of emphasizing the design of the mobile application and making it as simple as possible to use. Within the function register, it is checking if the parameters are met and has different criteria and conditions to make sure that the email is valid, if both the passwords match, and also checks if someone already has the same username or email as the individual who is currently registering.



Above is the login and registration page of Apple-tite. When a user downloads Apple-tite and first opens Apple-tite, the user is brought to the page above.

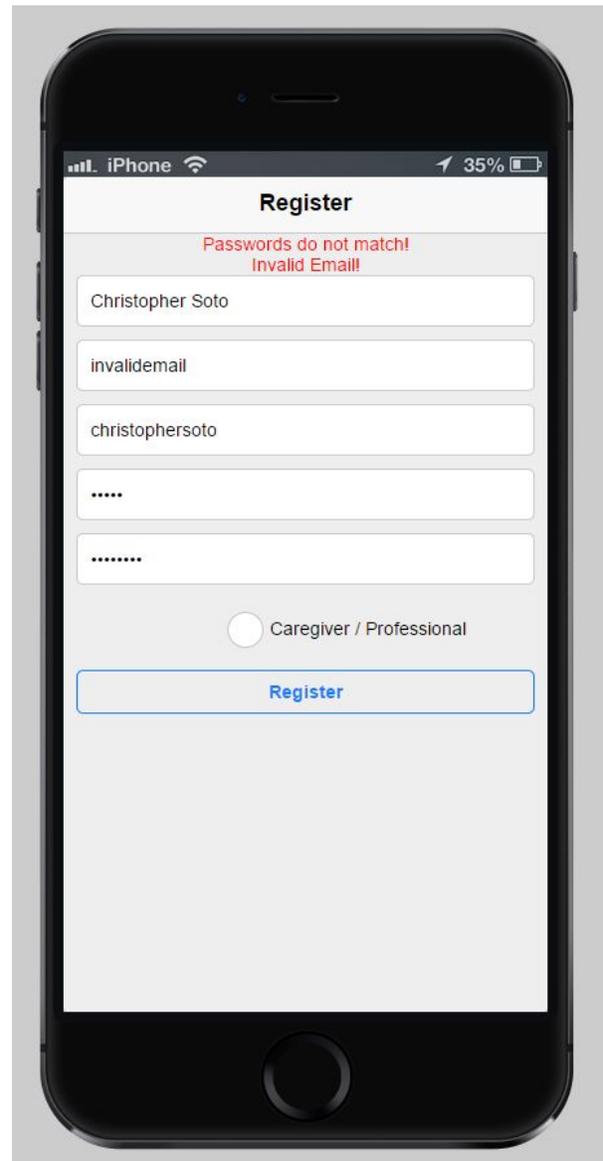
The Registration Functionality Displayed in the Example Below:

As shown by the example on the right, this is the registration part of Apple-tite. Registration consists of your name, email, username, and two password checks to make sure it is correct (as described in order). The example on the right is showing the functionality as mentioned above in a scenario in which a user might not have the same matching passwords or a valid email.

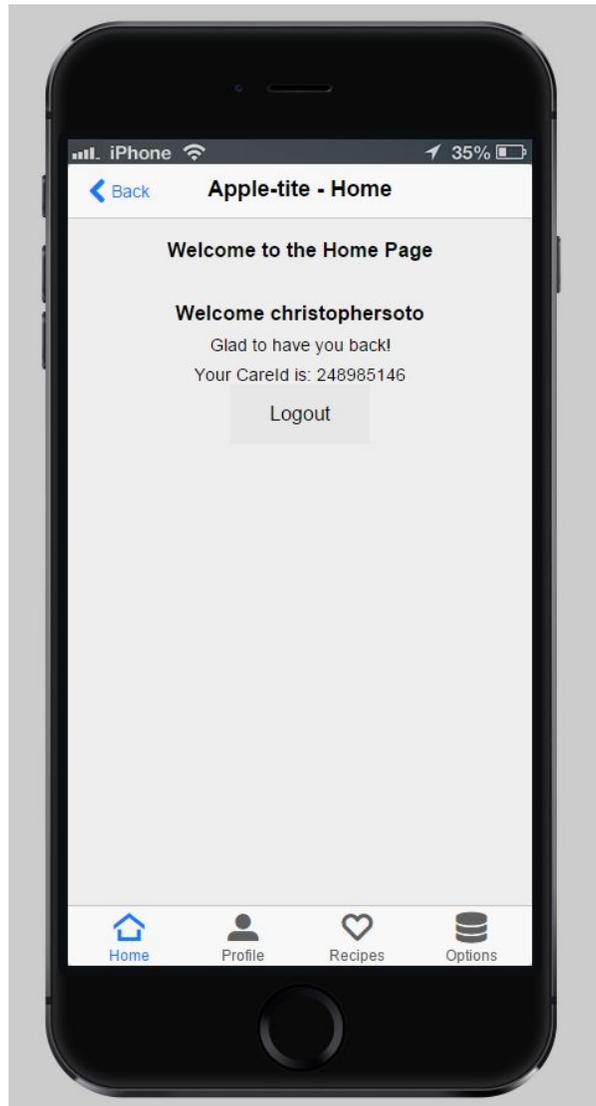
An email is considered valid if it has an at symbol (@) and as well as contain no invalid characters.

At the bottom of the registration the user is also asked whether they can check if they want to be registered as a caregiver / professional. Enabling this setting would allow the user to see the CarePage in the mobile application which allows the caregiver / professional to then enter the CareID of the user and a message will be sent to the user (If the user allows the caregiver / professional access or not), if granted the permission the professional has the ability to modify the user's health care information.

The ability to be a caregiver / professional is not only shown within the registration. The user also has the ability to switch their account status from user or caregiver in the options page as well.



CareID System



Once the user is logged in or registered, they are then brought to the homepage as shown above. Within the home page the user is greeted and their username is shown. But more importantly the user's CareID is displayed on the homepage.

When I originally designed the CareID system I designed the CareID to be a 9 digit randomly generated integer. No two users can have the same CareID and a CareID is created and assigned to a user when they first register.

```
function generateCareId(){
    return rand(100000000, 999999999);
}
```

As shown above the function generateCareId is returning a random value. This function is also used in other functions within the PHP document.

Profile Page and the UserID System

Normal User	CareGiver
<p>As shown above my account status is a user and I have the ability to switch my own health settings.</p>	<p>As shown above my account status is a care provider and I have access to go into the CarePage.</p>

Notice as well that the care provider can still modify his own health care settings if the caregiver / health professional wanted to, allowing them to still use the mobile application and have the ability to modify their own health settings.

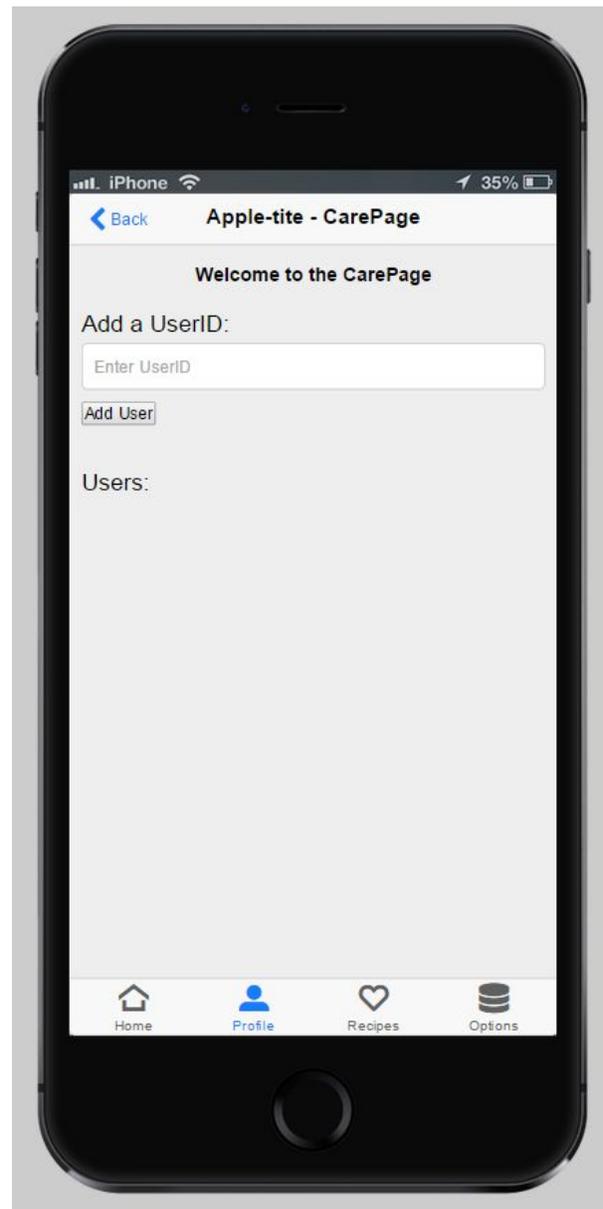
CarePage

As a caregiver I have access to the CarePage which is only restricted to users of the account type of caregiver.

Within the CarePage I can add a UserID and I can also manage and see the users that are associated with my account as a caregiver / health professional.

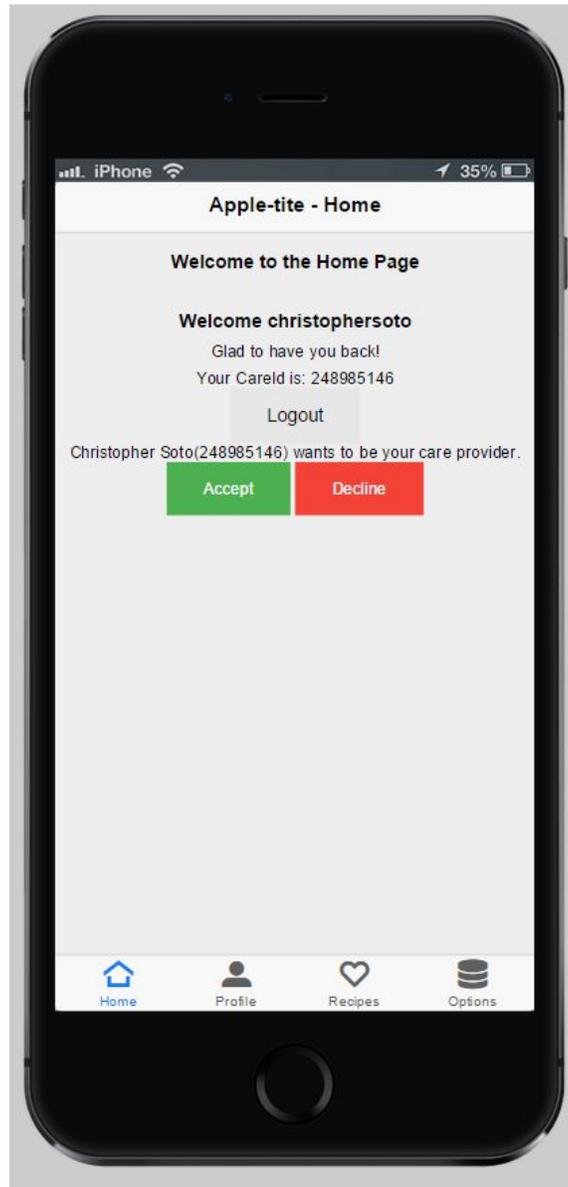
The UserID that is entered within the UserID box is located on the user's homepage.

In the beginning of the care system we see that the user christophersoto was given the CareID of 248985146.



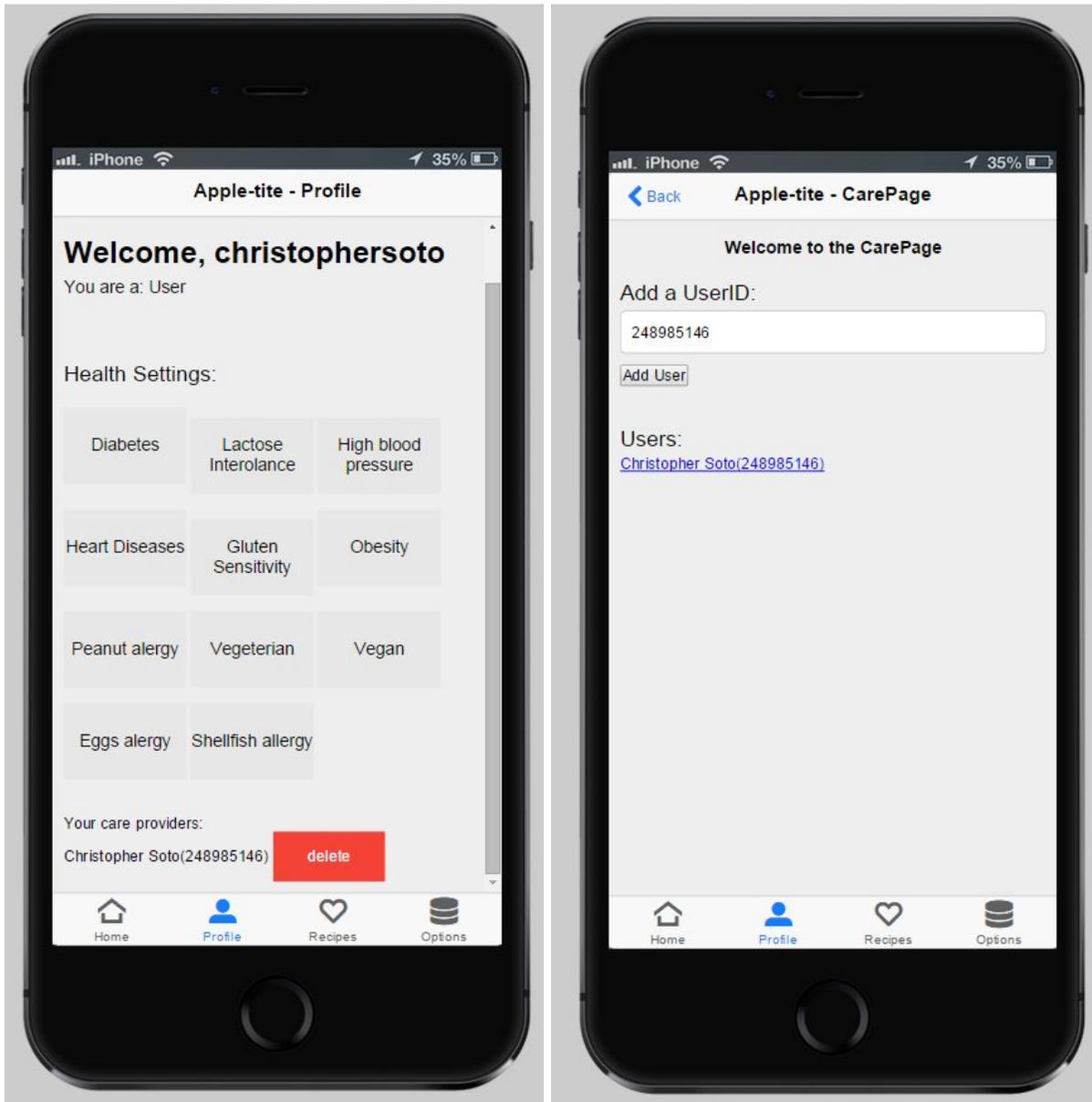
Once the caregiver / professional enters the UserID code within the CarePage, the user is then prompted to the following:

Invitation from the Care Provider to the User on the Homepage



As shown above, after the caregiver / professional enters the CareID of the user. The user is then prompted in the homepage whether the user accepts or declines that their account is than associated with the care provider.

What Happens After the User Accepts the Invitation



On the left image you can see that the user's page is updated and it now shows their new care provider. The user at any time has the ability to remove or keep the caregiver / professional as well.

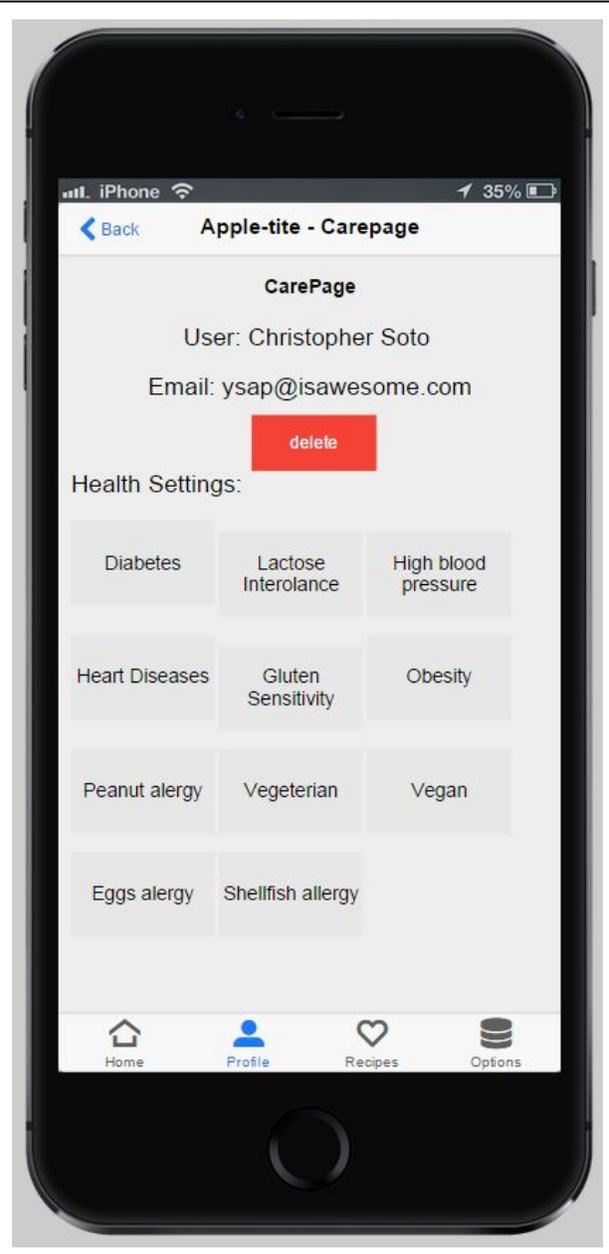
As shown on the image on the right, this is within the CarePage and now the caregiver / professional can see that the user Christopher Soto is now added to their user list. Having a user list allows the caregiver / professional to manage the users they have associated to their account much more easily.

Selecting a User from the CareGiver / Professional User List

Once the caregiver / professional selects the user of their choice from the user list on the CarePage the user's individually generated CarePage is opened.

The caregiver / professional now has the ability to see the user's name and email.

They also have the ability to now regulate the health settings of the user Christopher Soto and at any given moment have the ability to remove the user, if the user goes to another caregiver or professional or for any other reason.



Recipes

For the purpose of the demo, concept, and creation of the mobile application I only incorporated one recipe to the mobile application. The system in which the recipes are displayed within the mobile application and the pages associated with the recipes are randomly generated using PHP and SQL. By allowing the recipes to be created only on the MySQL database and having them automatically created within the mobile application allows for the recipes to be mass produced, if desired at a later point.



As shown in the image above, these are just some of the tables within the MySQL database that allow for the data from the mobile application to be sorted within the appropriate tables and then within the appropriate columns.

name	picture	category	html
Banana Split Oatmeal	food-photos/banana-oatmeal.png	BREAKFAST	<div id="center"> <h3>Banana Split Oatmeal</h3></div>

Within the table Recipes there are four columns:
Each column as well within the database has a specific type.

Name - The name of the recipe.
Type - integer(11)

Picture - Picture of the recipe.
Type - varchar(40)

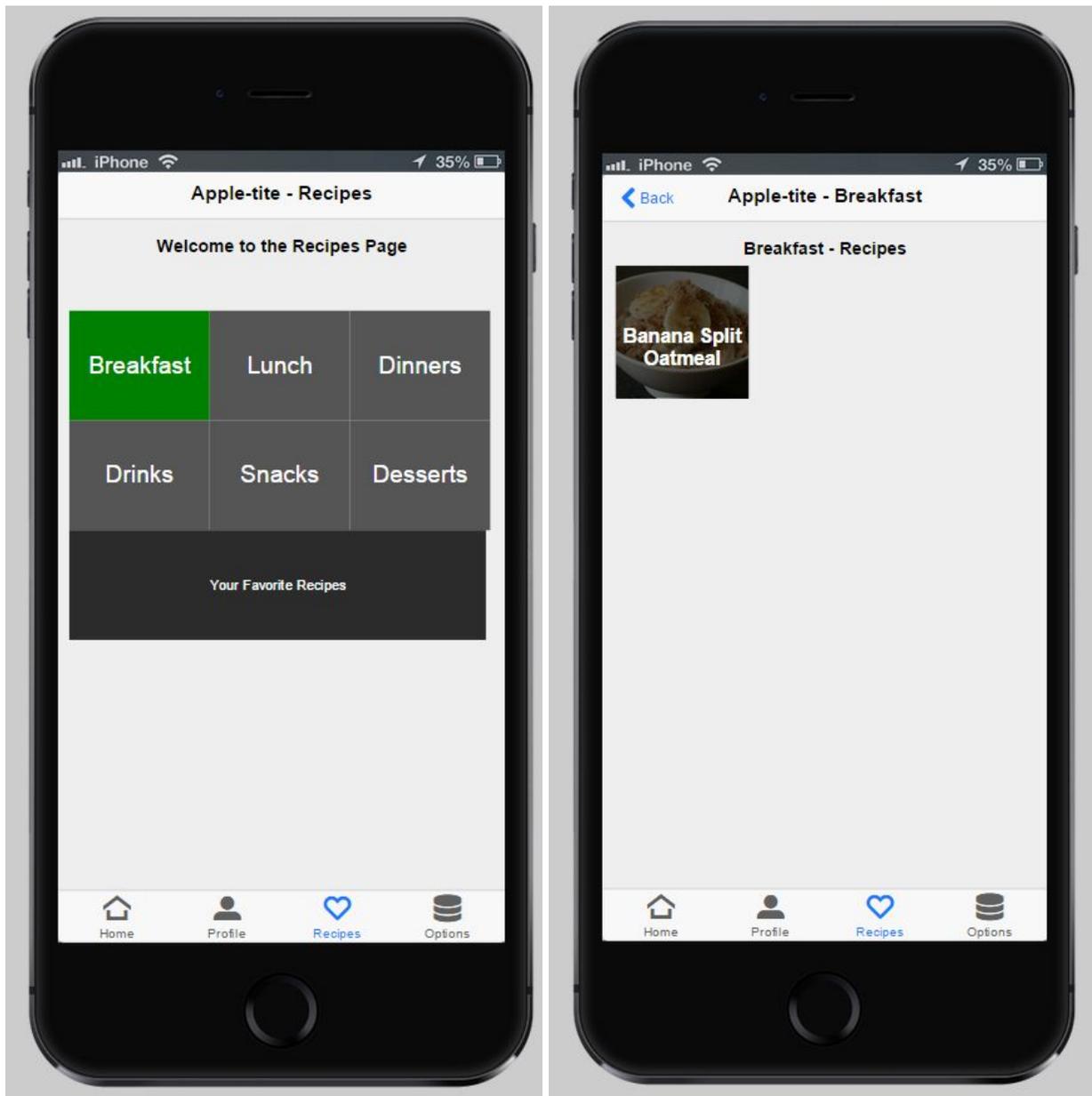
Category - The certain category that the recipe will be sorted into.
Type - varchar(100)

HTML - The HTML code for the Recipe page within the mobile application.
Type - longtext

Type - enum('BREAKFAST', 'LUNCH', 'DINNER', 'DRINK', 'SNACK', 'DESSERT');
Above shows the different categories that each recipe can be placed within.

Each column must be matched with a specific type as it will be storing data of the selected type.

Displaying Recipes



As shown on the left, these are the different categories that recipes can be sorted into.

On the right displays an example of a recipe within the breakfast category.

Recipe Page

Each recipe page contains:

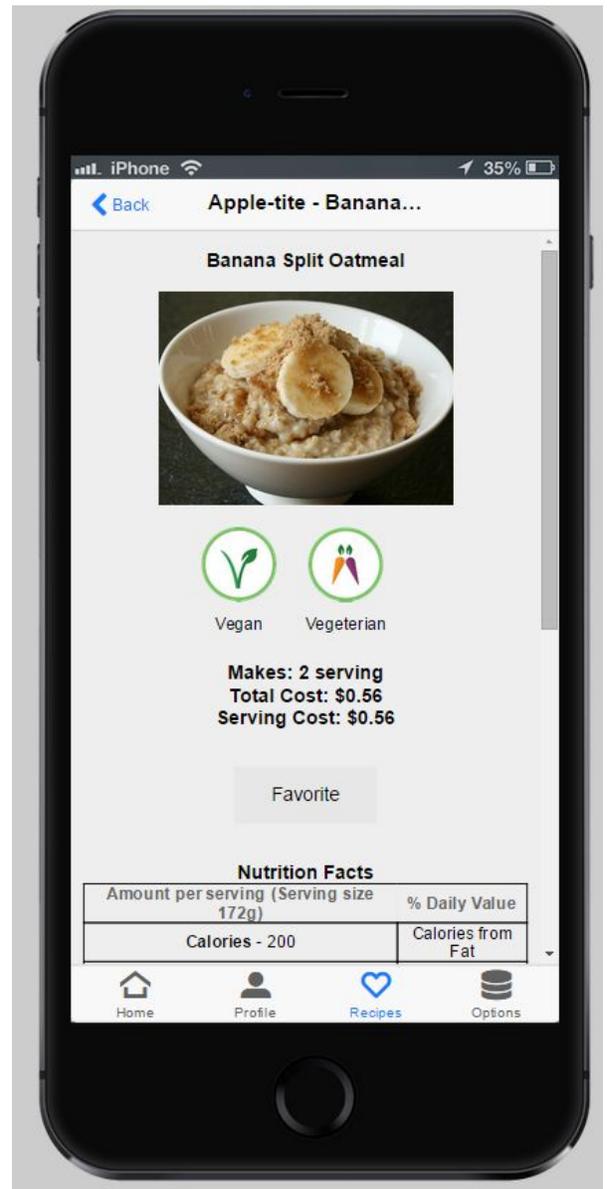
- The title of the recipe
- A photo of the recipe item
- Information about the serving size, total cost, and serving cost
- The ability to favorite the recipe
- Nutritional Facts
- Ingredients
- Directions

Within the recipe page as well the user can favorite the recipe as shown.

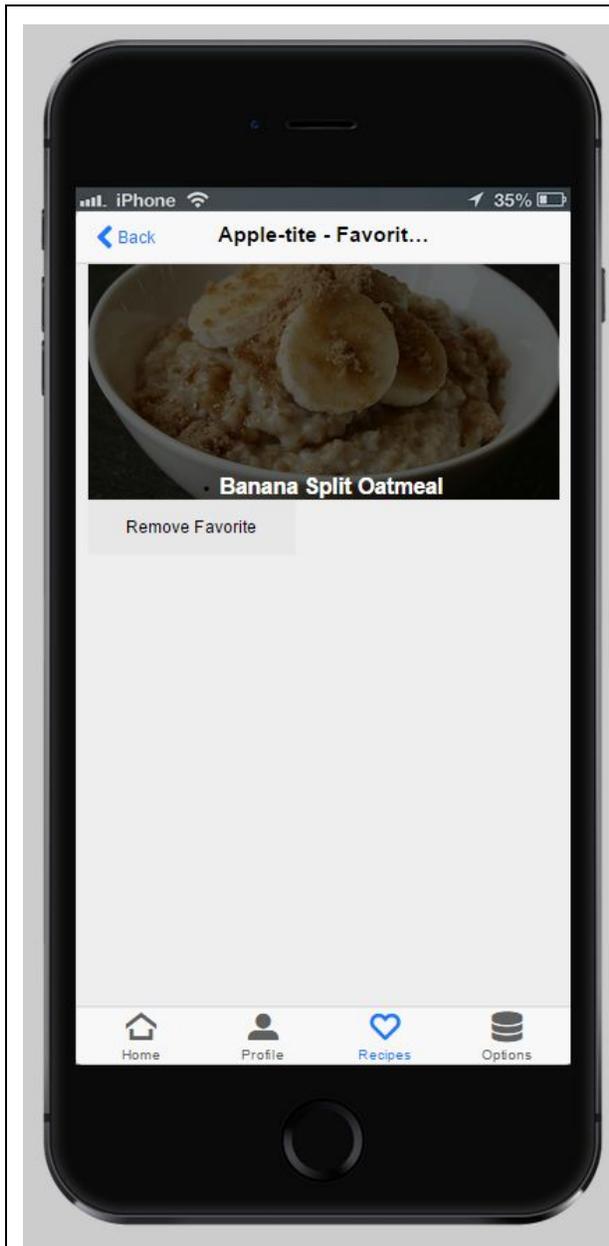
The Nutrition Facts displayed on the recipe page is designed and based on the U.S. Food and Drug Administration Food Label [5].

The ingredients displayed on the recipe page display the ingredient and the cost of the ingredient.

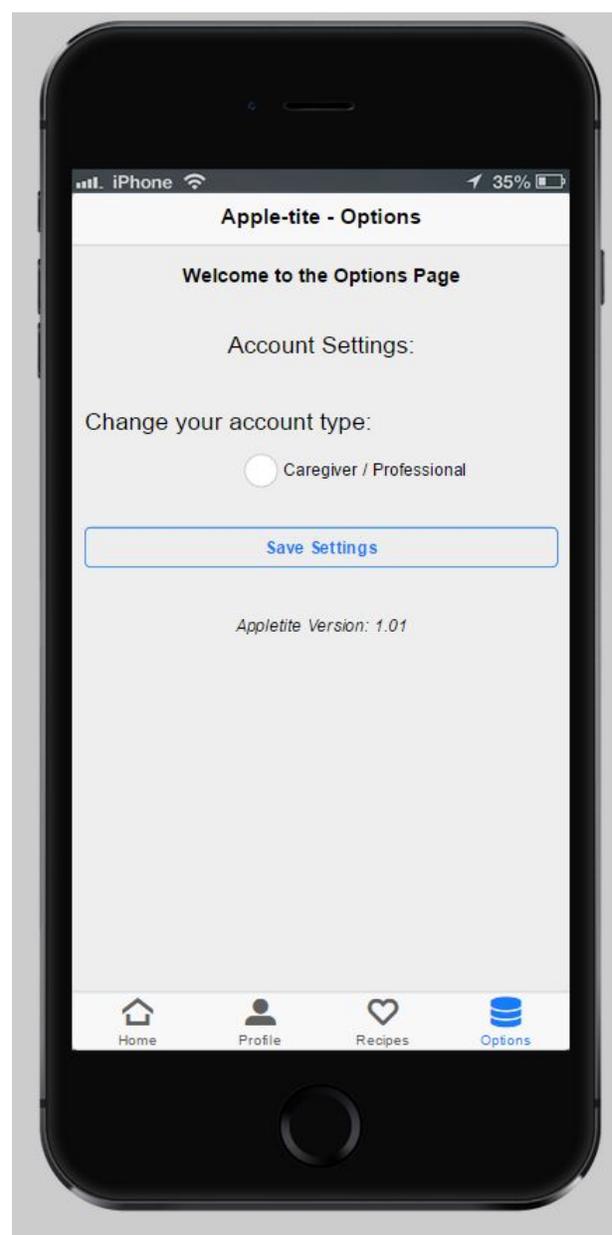
The directions page also displayed on the recipe page shows numerically step by step on how to create the recipe based off of the ingredients displayed within the recipe.



Favorites and Options



On any recipe page, once the user favorites a recipe it will appear at the your favorite recipes page which is on the recipes page, under the different categories of recipes. The user can always remove favorites and add them back as well.



As previously mentioned before, there are two times in which users can become a caregiver / professional. During registration if the checkbox is selected and as well as in the options page.

Results

id	username	email	name	password	careid	caregiver	last_login	register_date	ip
27	testnu	aksjd@ksjd.com	kadsj	\$2y\$10\$b7GjpnftZIsRIOq	387103087	0	2016-04-15 10:06:15	2016-04-15 10:06:15	108.46.184.94
28	testye	aksjd@ksjd.com	kadsj	\$2y\$10\$1lgZQ0Jw5Oivgl	433513203	1	2016-04-15 10:07:10	2016-04-15 10:07:10	108.46.184.94
29	suhsuihs	hiuhuih@yahoo.com	ujhuihiu	\$2y\$10\$laeT..luj33b.YsF	933571586	0	2016-04-15 11:15:03	2016-04-15 11:15:03	108.46.184.94
30	Dean001	Dsaghafiez@mac.com	Dean	\$2y\$10\$yiLthD8mwzY3li	161482130	0	2016-04-15 11:42:55	2016-04-15 11:42:55	66.87.9.210
31	Hshdjzj	Shdhdh@yahooshsj.com	Sherman	\$2y\$10\$pNn.IGsUp/0Uur	814022578	1	2016-04-15 11:46:46	2016-04-15 11:46:46	66.87.9.210
32	DJ	email@email.com	Mobile app	\$2y\$10\$WXFODQAYcy7	508974282	1	2016-04-15 13:16:00	2016-04-15 13:16:00	198.105.45.134
33	niel	niel@niel.com	Niel	\$2y\$10\$EX4d05rxUm6M	350360873	1	2016-04-15 13:58:47	2016-04-15 13:58:47	198.105.45.134
34	login	login@login.com	login	\$2y\$10\$Zsjh/Fh5bXtyNL	850226886	1	2016-04-15 14:18:27	2016-04-15 14:18:27	198.105.45.134
35	James	james@james.com	James	\$2y\$10\$AHN6zinnupjvVr	142421024	1	2016-04-15 14:25:57	2016-04-15 14:25:57	198.105.45.134
36	christophe	ysap@isawesome.com	Christopher Soto	\$2y\$10\$xmp65oicyR1ix9	248985146	0	2016-05-20 04:14:34	2016-05-20 04:14:34	108.46.184.94

The image above is displaying the table Users and the various columns within the Table.

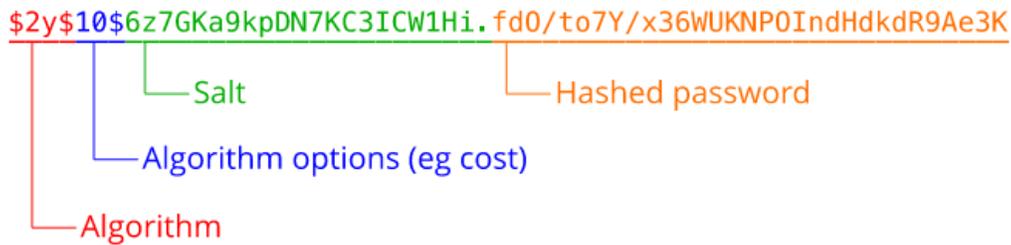
The results after a user is registered on the mobile application is that their information is sorted within the users table.

Example:

Table - Users

id	username	email	name	password	careid	caregive r	last_log in	register_d ate	ip
1	christopher soto	ysap@isa wesome.c om	Christ opher Soto	\$2y\$10\$x mp65oicy R1ix9ws ZCxU5u2 k.VFpW5 wVZ1NQL w6ln9l...	24898 5146	0	2016-0 5-20 04:14:3 4	2016-05-2 0 04:14:34	123.45 .678.9 10

Once a user registers on Apple-tite they are automatically assigned an id within the database and within the table users and column id. Information such as the user's username, email, and name are also inserted within the table users in their respective columns. You may notice that from the image above and from the example of the table users above that the password of users 27 - 36 all start the same \$2y\$10\$, and after that point they are all different. This is because each password received from registration are put into Salt and Hash.



Salt and Hash which is also known as Password Hashing is one of the most basic security considerations that must be made when designing *any* application that accepts passwords from users [6]. Without hashing passwords it is very easy to simply decrypt the passwords, if the database was compromised.

Salt and Hash prevents this by using a hashing algorithm which as seen by the diagram above adds the algorithm, the salt which is data applied during the hashing process and as well as the hashed password.

Here's an example of how easy it is to apply the Salt and Hash Algorithm in the programming language of PHP.

```
string password_hash ( string $password , integer $algo [, array $options ] )
```

The example below is from how it is used within Apple-tite.

```
$stmt=$db->prepare('INSERT INTO users SET username=?, email=?, name=?, password=?, ip=?, caregiver
$stmt->bind_param("ssssii",$username, $email, $name, password_hash($password, PASSWORD_DEFAULT),
$stmt->execute();
return true;
```

After the password the user is given a randomly generated CareID which is associated with every account during registration. If the user is a caregiver or a professional the value of the caregiver column would be equal to 1, otherwise it would be equal to 0 which simply means that the account is a user. Lastly the last_login, register_date, and IP are all saved within the table users as well. Last_login being the last time that the user logged onto Apple-tite, registration_date being the timestamp of when the user registered on Apple-tite and the last IP address that the user had when logging into Apple-tite or completing registration.

Here are some other tables that were not mentioned, that are also used for many different functionalities within Apple-tite.

MySQL Database Tables

careid_link			recipes_favorite		user_conditions		recipe_conditions	
userid	providerid	status	userid	recipeid	userid	usercondition	recipeid	recipecondition
15	17	ACCEPTED	34	1	35	DIABETES	1	DIABETES
32	32	ACCEPTED	36	1				
33	33	ACCEPTED						
34	34	ACCEPTED						
0	35	PENDING						
35	35	ACCEPTED						
36	36	ACCEPTED						

Within each of the following tables they are calling to the userid or recipeid. The table careid_link is responsible for the connection between the user and the caregiver / professional link within the mobile application.

Recipes_favorite is used within the mobile application for the functionality of allowing users to have the ability to favorite recipes and generate and display them on the mobile application.

Both user_conditions and recipe_conditions work together in the way that if a user has a condition in this case the disease diabetes and a recipe has too much sugar or is not healthy for an individual who has diabetes to consume than it will determine whether if the user has the same condition as listed on the recipe and this it will restrict the user from having the ability to see the recipe.

Conclusion

In conclusion Apple-tite as a mobile application has the ability to make an impact on our society and within our planet. Individuals who don't have access to medical services, a nutritionist, or are just looking for a way to lose weight and stay healthy now have a solution. There are other recipe applications on the Google Play store and on the iTunes store which provide steps to individuals on how to make healthy recipes but there are no mobile applications that focus on the target market of individuals who are apart of the older population. Many applications are badly designed and don't really take into consideration that there mobile application may be used by people who are not as experienced with technology as the developer or the company might be. These are the reasons of why Apple-tite has the ability to make an impact on society. Apple-tite was designed with the full thought in mind in creating a mobile application that is easy to use and as well as allow individuals to not only have the ability to find healthy recipes but also have the ability to have access financial information for each recipe, and as well as connect caregivers and professionals with the users.

Acknowledgements

For my acknowledgements I would like to thank the Young Science Achievers Program for funding my project. I would also like to thank Dr. Dean Saghafi for his mentorship and constant support throughout my entire project.

References

[1] *Humanity's Aging* [Internet]. National Institute on Aging; [cited 2016 May 15]. Available from: <https://www.nia.nih.gov/research/publication/global-health-and-aging/humanitys-aging>

[2] National Health Expenditures 2014 Highlights [Internet]. Centers for Medicare & Medicaid Services; [cited 2016 May 15]. Available from: <https://www.cms.gov/research-statistics-data-and-systems/statistics-trends-and-reports/national-healthexpenddata/downloads/highlights.pdf>

[3] *Getting Started Tutorial* [Internet]. Intel; [cited 2016 May 15]. Available from: <https://software.intel.com/en-us/xdk/docs/intel-xdk-guided-tutorial>

[4] *Ajax* [Internet]. Mozilla Foundation; [cited 2016 May 15]. Available from: <https://developer.mozilla.org/en-US/docs/AJAX>

[5] *How to Understand and Use the Nutrition Facts Label* [Internet]. U.S. Food and Drug Administration; [cited 2016 May 15]. Available from: <http://www.fda.gov/Food/IngredientsPackagingLabeling/LabelingNutrition/ucm274593.htm>

[6] *Safe Password Hashing* [Internet]. The PHP Group; [cited 2016 May 15]. Available from: <http://php.net/manual/en/faq.passwords.php>