Clemson Bucket List

Introduction
Clemson Bucket List is a responsive web application that provides an interactive experience for students to create and complete their very own “bucket list.”

This document is a technical reference manual, meant to explain in detail the work that went into creating this web application. Information regarding web development using HTML, CSS, Javascript, jQuery, PHP, mySQL, and more can be found on the following pages. If this document is lacking, please send inquiries to the authors, Anna Kutch and Kirsten Erich (contact information can be found in the Development Team section).

Our application will be available online and is designed to be responsive. This means that all devices with a web browser should be able to visit and interact with our application, and the layout will adapt to a wide variety of device widths. The embedded ability of the application to adapt to different widths ensures a positive user experience on a vast range of devices.

The functionality is geared towards Clemson students. The concept of a “bucket list” implies that there are tasks to be completed before a certain deadline (in this case, graduation). Clemson students will be able add tasks to their own bucket list and check them off (when complete) during their time at the university. It is our hope that this application will allow Clemson traditions to continue and encourage Clemson students to be more involved on and off campus.

<table>
<thead>
<tr>
<th>Date</th>
<th>Task</th>
<th>Person(s) Responsible</th>
</tr>
</thead>
<tbody>
<tr>
<td>30 March</td>
<td>1. Set up a meeting with Greg concerning DB constraints</td>
<td>Both</td>
</tr>
<tr>
<td></td>
<td>2. Fine-tune storyboard</td>
<td></td>
</tr>
<tr>
<td>6 April</td>
<td>1. Make final design decisions</td>
<td>Anna: #1</td>
</tr>
<tr>
<td></td>
<td>2. Finalize database(s) and functionality</td>
<td>Kirsten: #2</td>
</tr>
<tr>
<td>13 April</td>
<td>1. Progress on UI</td>
<td>Anna: #1 and media</td>
</tr>
<tr>
<td></td>
<td>2. Begin adding advanced features (geolocation, media)</td>
<td>Kirsten: #2 and geolocation</td>
</tr>
<tr>
<td>20 April</td>
<td>First Draft of Software and Documentation</td>
<td>Both</td>
</tr>
<tr>
<td>27 April</td>
<td>Final Draft of Software and Documentation</td>
<td>Both</td>
</tr>
<tr>
<td>30 April</td>
<td>Final Presentation</td>
<td>Both</td>
</tr>
</tbody>
</table>
Features
The features of the application center around a basic Add, Remove, Update, Delete scheme. Primarily all of the data in the application is changed via these methods. Additionally, geolocation and media (e.g. photos) features of the device that is being used to access the application is leveraged for a more cohesive, community experience while using the application. HTML5 has built in functionality for both geolocation and media features, thus making the process of accessing those feature quite a bit more straightforward. Below are the desktop and mobile layout for each of the pages of the application which highlight both the design and the functionality of the application at each stage.

Figure 1a: Home Page (Login/Sign Up)  
Figure 1b: Home Page Mobile  
Figure 2a: Tasks  
Figure 2b: Tasks Mobile
Home
When students first visit the website, they will be required to login or sign up. They will need to enter the necessary credentials - name and CUID - in order to create or login to an account. Credentials will be checked using a Javascript function that parses the strings and looks for a “valid” CUID (8 numeric digits preceded by a ‘C’ character). This page is minimalistic with a strong Clemson background image, so as not to overwhelm the user from the beginning. After an account has been set up, the user will be create their own bucket list.

Tasks
This tab will contain the universal task list (one of the tables in our external database). The user can browse through the picture and description of each task in order to determine which ones to add to their own bucket list. They can add tasks to their bucket list by click on the addition symbol on the right side of each task. In addition, users can add their own unique task to the list. At the top of the page, there is space for a new task to be submitted. Once submitted, the new task will be reviewed by the authors of the website for approval. It at the author’s discretion whether or not to be marked “approved” and added to the Tasks table in the external database.

My List
This tab contains a user’s personal bucket list. This includes any tasks that they have chosen to add from the Tasks tab to their bucket list. From here, the user is able to mark a task “completed.” In addition, the user has the option of uploading a photo of the completed item or checking in via geolocation to prove that they’ve completed said task. These features are built in using built-in functionality from HTML5 and Javascript. We referenced similar projects to learn how to integrate these features.

Photos
This tab displays the photos from the entire Clemson Bucket List community. There will be an option for a user to “hide” his/her photographs if he/she does not want the pictures to be shared amongst all users. Users can chose to add descriptions to their photos which provides a more social aspect to the application. Photos will be stored in an external database (more details to follow).

Settings
This final tab will allow users to change certain settings associated with their account. They will be able to pick a color scheme for their view of the application. This will be accomplished by using PHP scripts to call different CSS style sheets depending on the theme the user has chosen. The color scheme can be toggled between a light and a dark theme. The darker theme may be more appropriate for users with poor vision or users accessing the application in a very bright setting, as there will be higher contrast. Users may also disable media which is a helpful setting if they are running on a slow connection. Thus, without having to load the media, the application will still have all relevant text information and load at a much faster pace. Also, the user may not want to share their photos with the community, so they can change their photo settings to only allow their photos
to be seen on their account. This will be moderated by setting a flag on the photos that indicates it will not be shared. The user can fine-tune the images that he/she would like to share with the community on each individual photograph as well. Another option that we have provided is whether or not to use the “Activities” that have not yet been approved. This setting might help to filter out inappropriate or irrelevant activities to be chosen to add to the users Bucket List.

Databases
The external database for the application will be composed of three main tables - the user table, the task table, and the bucket list table. It will be hosted on the Clemson server and accessed using PHP script calls within Javascript functions. The user table will contain a primary key, the user’s login credentials, and any additional profile information they decide to include. A foreign key will link the user to their bucket list. The task table will also contain a primary key, the title and description of the task, and the GPS location (if applicable). Again, a foreign key will link this table to the bucket list table. The bucket list will be linked via a foreign key to the users table and the tasks table, it will also contain a field that shows if a task is completed or not.

In addition to these three main tables, there will be a photos table that will be associated with the bucket list. Photos and captions can be uploaded in conjunction with a completed task.

Figure 6: External Database Schema
The internal database structure will be used for the user settings. There will be a single table three columns. The first column will be an autoincrement key, the second column will be the setting, and the third column will be an integer to indicate the users preference(s) for the given setting.

Technical Implementation Details
In building a responsive web page, the bare bones requirements are HTML5 and CSS/ CSS3. In order to make the application responsive, the use of media queries is required. Further, the use of JavaScript and jQuery is needed for any web page interactions as well as more complex plugins. Finally, the PHP scripts must be called within the JavaScript files in order to make calls to the Database(s) and update relevant information on the page*.

*In later versions of the Technical Reference Manual we will include relevant code excerpts for further clarification.

Development Team
Anna Kutch
Senior, Computer Science
E-mail: akutch@clemson.edu
Website: http://people.cs.clemson.edu/~akutch

Kirsten Erich
Junior, Computer Science
E-mail: kirstee@clemson.edu
Website: http://people.cs.clemson.edu/~kirstee

Acknowledgements and References
Thank you, Dr. Pargas, for allowing us to work on this unique project and for giving us the tools and advice to be successful.

Thank you, CPSC 4820 classmates, for offering your constructive criticism and sharing your projects with us.

Thank you, w3schools, for your extensive tutorials on the languages of web design and development.
http://www.w3schools.com

Thank you, Aquent Gymnasium, for your class on responsive web design.
http://gymnasium.aquent.com

Thank you, David Walsh, for your guide on integrating the camera functionality into a web application.
http://davidwalsh.name/browser-camera