Young Chefs

Experiment HTML Specification

Intro

All experiment content for **Young Chefs** is loaded dynamically. No experiment in the app is "hardcoded" in. Creating a new experiment does not require changes to the source code of **Young Chefs**.

This document is a living specification of the details that the app requires from all Experiments. This includes how to create and properly package an entire Experiment from start to finish.

Experiment Package Overview

Experiments are constructed from the following assets:

Asset	Type	Naming Convention	Purpose
Name	Text	example	Every experiment is required to have a unique name. This is used to group together and classify all the associated files. This is also the name presented to the user.
Content	Multiple HTML Files	example-#.html	Each HTML file defines a single page of the experiment's walkthrough. The screens are ordered according to the # in their filename. Numbering can start at 0 or 1.
Icon	Single Image File *(optional)*	example.png	Image to display on the home screen along with the name of this experiment. Can be *.png* or *.jpg*
Styles	CSS File	styles.css	This is a standard file automatically copied into every experiment. It defines the style and formatting of the HTML.
Code	Javascript File(s)	example.js	These are standard files automatically copied into every experiment. It should not be modified. It defines how the app and the experiment communicate.

Experiments should hold all their assets in a single folder, and that folder's name should be the experiment name.

Name

The experiment name does not need to be stored in a file, but must remain consistent throughout an experiment. We recommend naming the folder the experiment name.

This name defines how the app searches for and loads files associated with the experiment.

Content

The HTML which defines each page, or screen.

Every HTML file associated with an experiment must be named `[experimentName]-[#].html`, where `[experimentName]` is the name of the experiment and `[#]` is the order of the screen. *(ie. example-2.html will be presented third, because counting starts at 0).*

An example of content, with inline comments can be found in the demo experiment.

HTML Structure

Every HTML file which defines content must adhere to the following principles:

The HTML should be structured like this:

```
<!DOCTYPE html>
<html lang="">
<head>
<meta charset="UTF-8"/>
<meta name="viewport" content="width=device-width, initial-scale=1.0"/>
<script src="jquery-2.1.4.js"></script>
<script src="fastclick.js"></script>
<script src="javaswift.js"></script>
<link rel="stylesheet" href="styles.css"/>
</head>

<body>
HTML CONTENT
</body>

</html>
```

Providing for in-app Navigation

You should include buttons for navigation to other screens.
To include these, first define a navigation block as such:

```
<div class="navigationBlock"> […] </div>
```

Inside the navigationBlock div, include this for backward navigation:

> <div class="back block">Previous Screen…</div>

Inside the navigationBlock div, include this for forward navigation:

> <div class="next block">Next Screen…</div>

Important: If you include both a forward and a back button, make sure the back button comes first, or else the spacing of the buttons may become clipped.

Note that these navigation buttons will function inside the **Young Chefs** app, but not in a test browser.

Creating Questions

To create a question, simply include this:

> <textarea class="input" rows="4" id="[unique question identifier]"></textarea>

[unique question identifier] is a unique string of text. If this is non-unique, user responses will not save correctly.

Icon

The icon image which should be presented along with the experiment name when a user is selecting which experiment they wish to do.

Icons should be 400px by 400px so they look good on iPads of higher resolutions.

Styles and Code

These files are not intended for editing by non-developers.