

# HTML, CSS, JAVASCRIPT IN-APP DEVELOPMENT WORKFLOW

Date      September 2020

Author: Calvin Parks: [cparks@laughlin.com](mailto:cparks@laughlin.com)

## Overview Discussion:

History: The Laughlin Email development strategy was created by Dariusz jarosz and perhaps other Dev's that I am not aware of.

I started developing In-Apps for Laughlin when Dariusz left the company in October of 2019.

## Observations:

Braz's In-App development is pretty simple to understand. They are HTML content served in a Webview in the Hoosier IOS and Android native app.

## Dev Team

Laughlin Email Team members

- Jenny Maglio: Marketing Automation Associate Director- [jmaglio@laughlin.com](mailto:jmaglio@laughlin.com)
- Cathleen LaFond: Marketing Automation Specialist - [clafond@laughlin.com](mailto:clafond@laughlin.com)

## Documentation Links

VIDEO: Learn How to Use Appboy's In-App Messages

<https://www.youtube.com/watch?v=nwp7prafy3Y>

In-App Messages using HTML5

<https://github.com/Appboy/appboy-custom-html5-in-app-message-templates>

HTML In-App Messages

[https://www.braze.com/docs/user\\_guide/message\\_building\\_by\\_channel/in-app\\_messages/customize/](https://www.braze.com/docs/user_guide/message_building_by_channel/in-app_messages/customize/)

BRAZE SDK for IOS and Android Development

??????????????

Hoosier Lottery website

<https://hoosierlottery.com>

The official IOS Hoosier Lottery App

<https://apps.apple.com/us/app/hoosier-lottery/id894092490>

The official Android Hoosier Lottery App

[https://play.google.com/store/apps/details?id=com.gtechg2.mobile.indiana&hl=en\\_US](https://play.google.com/store/apps/details?id=com.gtechg2.mobile.indiana&hl=en_US)

## SETUP TOOLS

**PhotoShop** - Raster graphics editor

- Website <https://photoshop.com>

**TinyPNG** - Image compression site that uses "smart lossy compression techniques" to make your images smaller.

- Website <https://tinypng.com>

**Laughlin In-App Repo** - Repository for the final email that has been launched.

- [https://github.com/laughlin/lc\\_client\\_emails/tree/master/clients/Hoosier%20MyLottery/In-app](https://github.com/laughlin/lc_client_emails/tree/master/clients/Hoosier%20MyLottery/In-app)

**Braze** - Braze is a customer engagement platform used to create email campaigns

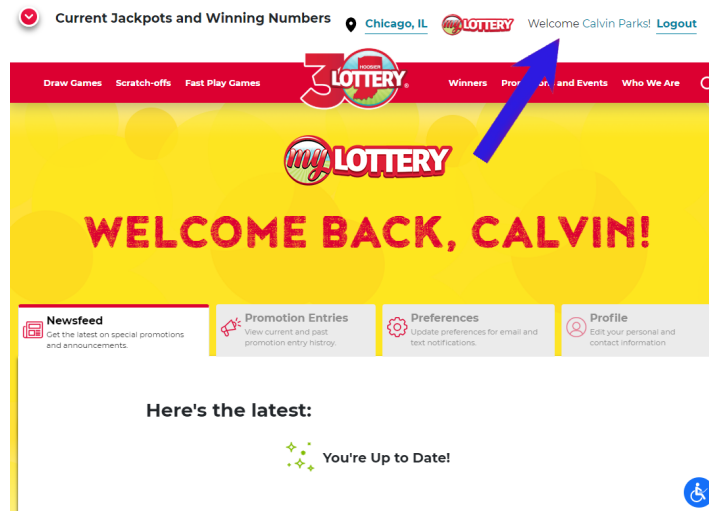
<https://www.braze.com>

- Login: hoosierbraze@laughlin.com - Griffin2018

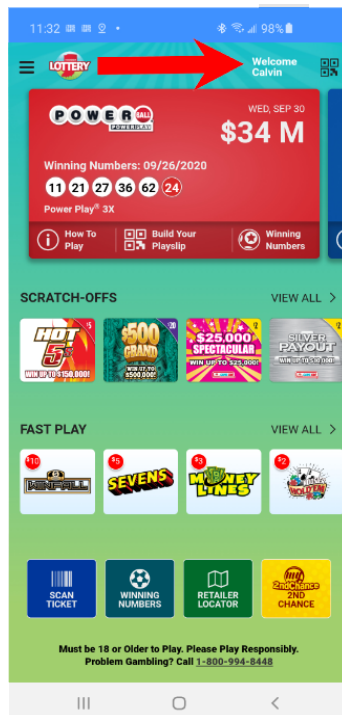
## Step 0 Setup

Download apps and set up notifications

Log into <https://hoosierlottery.com/> site and signup and sign-in the set up notifications



Download hoosier lottery app and sign-in to it using the username and password that created on the website



## Step 1 In-App Construction

There are 3 ways you can begin your coding

1. You can start your code completely from scratch
2. You can start your code using In-App code from Laughlin Repo
3. You can use a Braze In-App template from the Braze(formally Appboy Repo)

Code can consist of the following:

- Javascript
- HTML
- CSS
- Supports local and external code libraries
  - GreenSock
  - CreateJS
  - jQuery
  - Slick
  - Pagination
  - etc
- InApp Links
  - Open the News Feed **<a href="appboy://feed">View Feed</a>**.
  - Close the in-app message **<a href="appboy://close">Close</a>**
  - Open link using mobile browser and not webview
    - **<a href="http:www.site.com?abExternalOpen=true"></a>**
  - etc
- Constraints
  - All Code and assets should be placed in the same directory
  - width: 370px;
  - height: 601px;
  - App weight is 400K

## Step 2 testing

When building the bulk of your code it is convenient to:

- Test locally on Chrome browser
  - Note: Android webview are using Chrome
- Test locally on Safari browser
  - Note: IOS webviews are using Safari

## Step 3 Upload to Braze

The screenshot shows the Braze Campaigns interface. On the left sidebar, the 'Campaigns' menu item is highlighted with a red arrow. In the main content area, the 'Drafts' filter is selected from the 'Show' dropdown, also indicated by a red arrow. A red circle highlights a list of drafts, with a red arrow pointing to the draft titled 'Hoo Knew Trivia In-App Screen 1'. Another red arrow points to the 'Choose an in-app' button in the top right corner of the draft list.

Group By:	Name	Sent	Last Edited
[No Grouping]	30th_in-app_3and4	N/A	Aug 24
Only Show Mine	30th_in-app_4	N/A	18 days ago
All Active	Learning in-app	N/A	8 days ago
One-Time Sched...	Hoo Knew Trivia In-App Screen 1	N/A	7 days ago
Recurring Sched...	Hoo Knew Trivia In-App Screen 2	N/A	7 days ago
Triggered	Cheer Challenge Test 4	N/A	7 days ago
Multivariate Tests	Cheer Challenge #2	N/A	7 days ago
Drafts	Cheer Challenge #5	N/A	7 days ago
Standard	2020 - Midday WN Alerts - Email	0	Feb 4

The screenshot shows the 'Edit In-App Screen' interface. A red arrow points to the 'Custom Code' dropdown menu. Another red arrow points to the 'Test' button. A third red arrow points to the 'Upload Zip' button. A fourth red arrow points to the 'Paste Your HTML Here' text area. Red text annotations provide instructions: 'After your assets and code is in the system click here to send in-app To devices' points to the 'Test' button; 'Upload only CSS, Javascript & Images' points to the 'Upload Zip' button; 'Place your Code here' points to the 'Paste Your HTML Here' text area. A warning message states: 'We do not support previewing Custom HTML messages at this time.' and 'We did not detect a close button in your In-App Message. Include a link with a Close Button action in your HTML to allow users to dismiss your message:'. Below this, a code snippet is shown: `href="appboy://close"></a>`.

## Step 5 Test inside of the Hoosier App

Android and IOS

- Send Test Notifications
- Send test email and check results on a limited number of Devices
- Debug any issues and resend test until all bugs are fixed

Campaigns > Edit 'Hoo Knew Trivia In-App Screen 2'

Resources ▾ Get Help ▾ Calvin Parks 👑 🔔

Campaigns Edit 'Hoo Knew Trivia In-App Screen 1' ✕ Edit 'Hoo Knew Trivia In-App Screen 2' ✕

Custom Code  
Use HTML, CSS, Javascript, and assets

Preview

Compose

Settings

Test

DEVICE TYPE

Phone

We do not support previewing Custom HTML messages at this time.

TEST RECIPIENTS

Select at least one Content Test Group or individual user to receive this test message. Messages will be customized with recipients' attributes by default.

Add Content Test Groups ⓘ

✕ Hoosier Test Group ✕ LC Seed

Add Individual Users

Add external ids, emails, or phone numbers

✉ Send Test

PREVIEW MESSAGE AS USER

Random User

Get Random User

**Remove ALL Test Groups**

**Add email addresses or phone numbers that you want to send the in-app to.**

◀ Back Compose > Delivery > Target Users > Conversions > Confirm Forward > Save as Draft



## Step 6 Email Marketing Specialist and Internal team review

### Save as a draft when done.

Inform the Marketing Specialist that the email is ready for the internal team to review.

The team might request updates to the in-app. If so then complete the change request and **repeat step 5**.

## Step 6 QA Testing an Debugging

Once the internal team has reviewed the email and is satisfied, they will it send it to the QA department for full device/client testing.

Any bug found will be for a specific device/client combinations



You should then fix any bugs reported. Then test then on the specific device/client where the bug was reported. Then you should **repeat step 2**

**If no more bugs are found, then the Internal team will inform you that the email is completed and will soon be approved.**

### **Step 7 Upload it into REPO.**

After the email is officially approved add it to the repo.