REQUEST FOR COMMENTS

Apps Flagship Deeplinking

Problem & Context

When the Flagship apps launch, CRM emails triggered from user actions will contain links to the Photobox website.

These links will open a web browser when clicked, even if they are opened on a device with the Flagship App installed. This leads to a broken experience as the web won't support app creations at launch.

An example of a CRM email that is triggered is the "basket abandoned" email which is sent to users after adding creations to their app basket but aren't checked out, after a period of time.

Considered, but out of scope

We are not proposing to change the email text or links for the app right now. The logic for this, in the UK specifically, resides in the legacy CRM system that is in maintenance mode.

We can only open the app on devices where the app is already installed.

Solution

Architecture

Android and iOS have a mechanism to register URLs or patterns of URLs to open specific apps. These are defined within the app package.

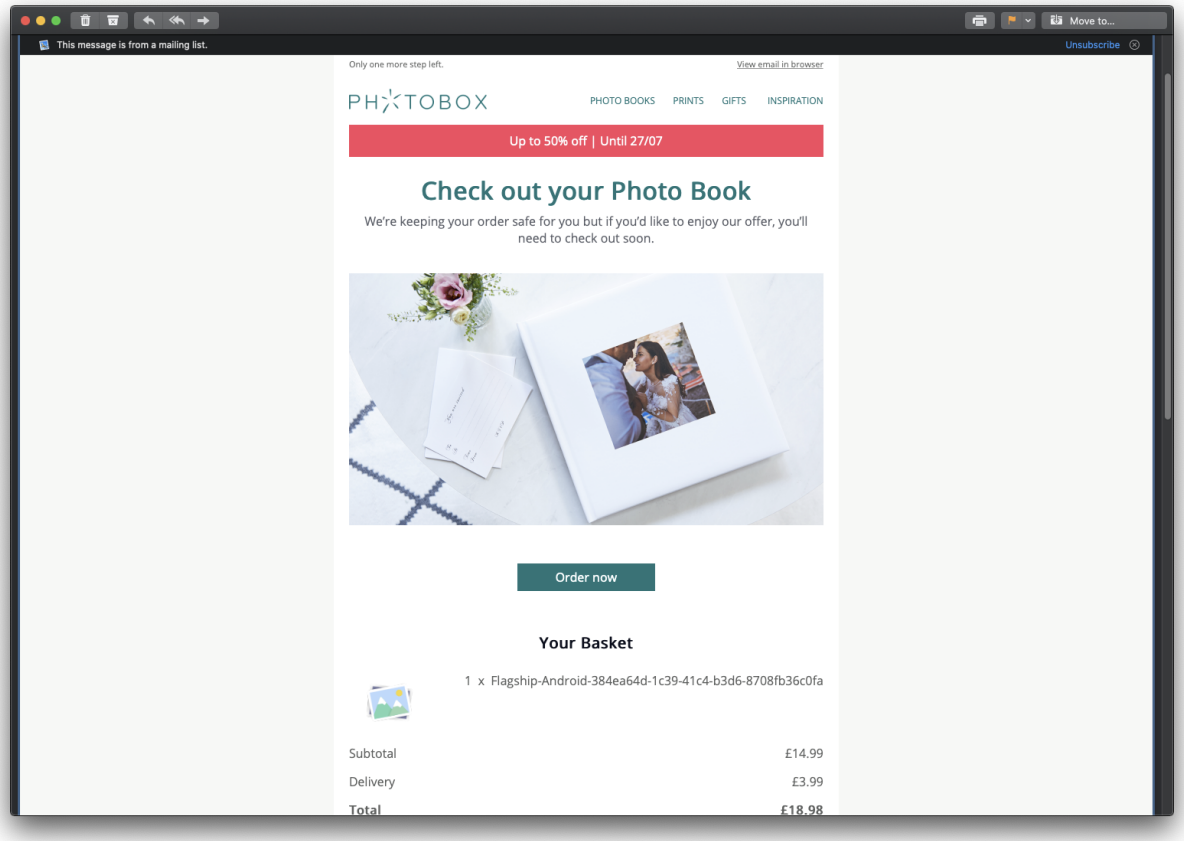
There is not enough granularity to match past host and path components in the URL to prevent the app from being opened for all unsupported links, so we plan to:

- Register supported Photobox domains to open the app when entered or part of a link that has been tapped in an email
- Handle supported URL paths in the app to deeplink into the appropriate app screens
- Redirect to the user's default browser for links that aren't supported by the app

The defined list of domains and paths will be shipped with each app binary, and configured based on what the app will support, but initially we will register the following domains to be handled by the app:

- pbox.photobox.co.uk (https)

Example (supported link):



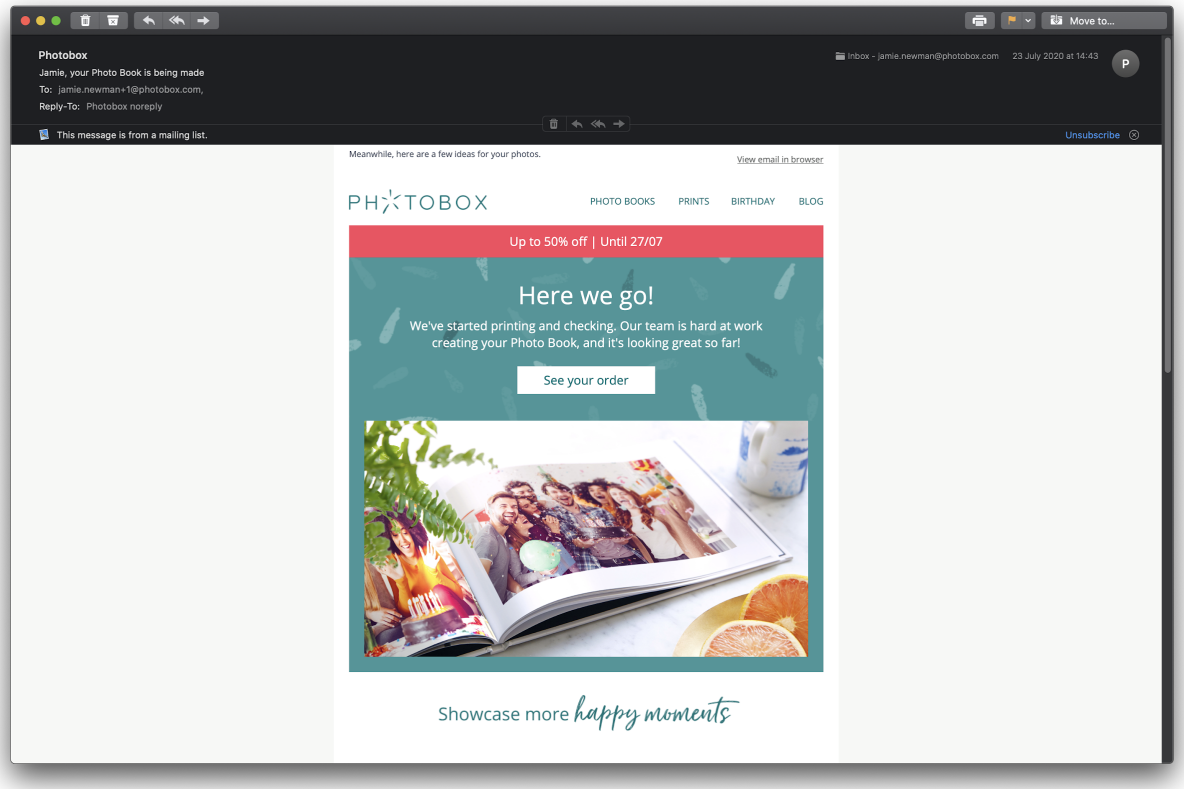
The **Order now** button when tapped opens a web browser with the following URL protocol and domain: https://pbox.photobox.co.uk/ (1) and the path

dynclick/photobox-uk/ (2) and a query strings which contains **p1=www.photobox.co.uk/checkout/basket** (3)

(1) When opened on a mobile device with the app installed, the app will have registered the https://pbox.photobox.co.uk/ protocol and domain as one that it supports in its metadata
(2) And (3) The path and querystrings will be matched in-app for supported routes using a regex matcher to look for the basket path in the query string value

The app will then open and land on the Basket tab

Example (unsupported link):



The **See your order** button when tapped opens a web browser with the following URL protocol and domain: https://pbox.photobox.co.uk/ (1) and the path

dynclick/photobox-uk/ (2) and a query strings which contains **p1=www.photobox.co.uk/my/account/history** (3)

(1) When opened on a mobile device with the app installed, the app will have registered the https://pbox.photobox.co.uk/ protocol and domain as one that it supports in its metadata
(2) And (3) The path will match but the querystring will not

The app will open the default device web browser with the unsupported link for the web browser to display.

Risks & Open Questions

This is a blunt tool to provide a better experience for app users, but it comes with the following trade-offs:

- Supported links will not make a network call, which has implications for tracking and cookie setting

Alternative Approaches

Alternative options are:

(1) Support app creations properly on the web. This was discussed with the Editor team and there isn't currently sufficient resource available to do this in the near term.
(2) Don't send CRM emails from app actions (requires custom CRM development, which was ruled out)
(3) Send specific emails for app that can be handled separately (requires custom CRM development, which was ruled out)