

**Northern Arizona University**

**Project Requirements**

**Bit Tag**

**Temitope Alaga, John Dance, Josh Frampton, Jun Rao**

**CS 476**

**Version 1.0**

## Table of Contents

[Table of Contents](#)

[Introduction](#)

[Problem and Solution Statement](#)

[Functional Requirements](#)

[Environmental Requirements](#)

[Non-Functional Requirements](#)

[Potential Risks](#)

[Project Plan](#)

[Glossary](#)

[Appendix](#)

## Introduction

Bit Tag Technologies has sponsored our team to develop a mobile app in order to extend their current functionality to further platforms. Their current platform is limited to a Chrome extension and so they are trying to build out their platform to include a wider audience. Building the Bit Tag application into a mobile app will allow for more people to easily access the technology. Since the current desktop application is built into an extension, it isn't currently easy or possible to use the application outside of Chrome. This mobile application will show that it is possible to port this technology to a variety of platforms and show that the technology can be viable. The mobile app should exhibit the same functionality of tagging and viewing tags that is currently available through the extension. This includes being able to view tags on the Bit Tag tag repository, being able to navigate to the area of a target page that has been tagged, and being able to create new tags on a desired web page. In developing this, we need to keep in mind that web technologies in this context are limited to a single thread so the application needs to be lightweight. If the app does not perform as well as the current desktop platform, users will be less inclined to be interested in the mobile app.

## Problem and Solution Statement

The sponsor of this project is Bit Tag Technologies, a local Flagstaff startup run by Wyatt Brown. Bit Tag Technologies main product currently is called Bit Tag. Bit Tag is a technology that allows users to share specific content with other users across the web. This is unique because it allows for tagging and linking of specific content, as opposed to the standard sharing of links on the web. This is significant because with this technology, users are able to share exactly what they want, rather than sharing a link and having to describe where on a page this content is located. The benefits of this are seen most immediately when thinking about sharing a 100 section Wikipedia article. Rather than asking someone to find the nth paragraph, now we can simply link that paragraph to the user.

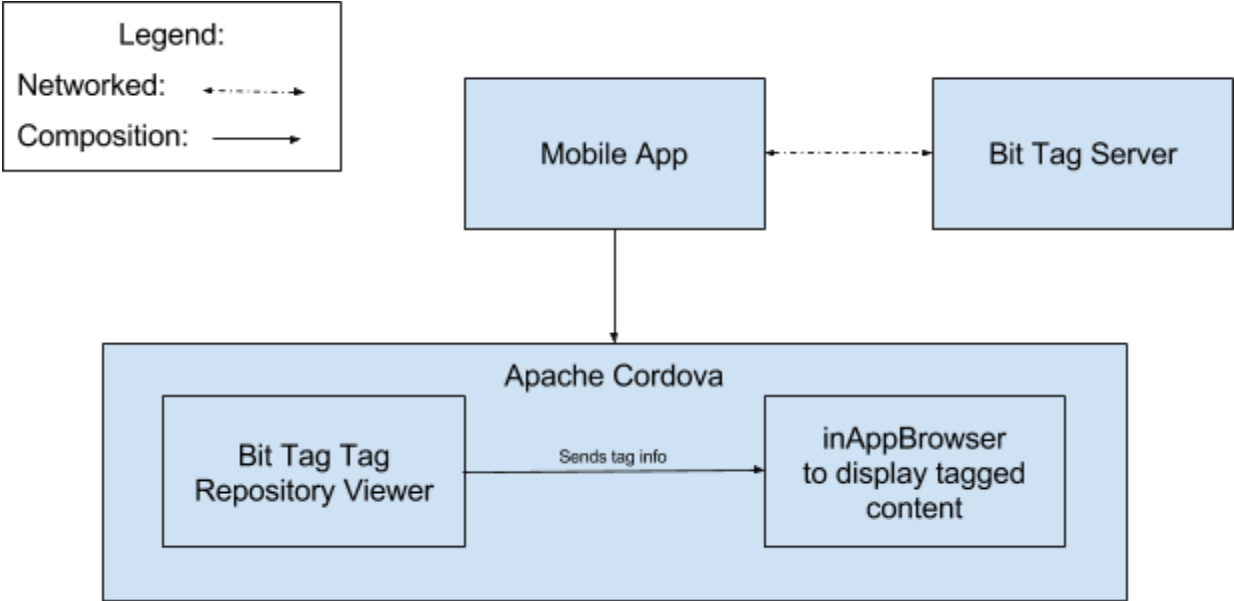
There are several challenges that Bit Tag Technologies is currently facing. Since the company is a startup, they are still trying to find a niche for their content to fit. They currently have a backend powerful enough to make this tagging process possible, it's just a matter of finding exactly where and how they want to deploy a front-end solution. Currently the solution is a more social media-oriented platform that allows for the sharing of content in a public collection on their [website](#). While this is a novel idea, they want to explore other possible use cases for their technology. Next they face a large limitation on their front end deployment. Currently, the core functionality (tagging and following tags), is currently limited to a Google Chrome extension. The issue here is if a user does not have this extension installed, they cannot use the product. From here the issue becomes that on a mobile platform, browser extensions are not supported so this method is not viable. Bit Tag Technologies wants a way of tagging and viewing Bit Tags on a mobile platform.

The software system we are developing is designed to alleviate the issue of mobile incompatibility, while at the same time defining a meaningful front end experience for users. Our system will utilize modern Android<sup>4</sup> development technologies accompanied by the use of several web technologies to create a convincing proof of concept application. Because this is a proof of concept, we have the freedom to take creative liberties to show that this deployment of Bit Tag is viable.

Our system will address and solve the main issues that Bit Tag has with their current limited product and allow for a meaningful proof of concept that will demonstrate viability as a mobile platform. The key pieces of functionality in our planned system include, access to the current Bit Tagged content available on the Bit Tag Box website, the ability to follow tags to their intended destination and highlight the tagged content, the ability to tag web content in the same way as available on Google Chrome, and to be able to access all of this on a mobile platform. The data used in the application will be provided by the preexisting Bit Tag back end system via an API as provided by Bit Tag Technologies. The data then generated by the system will be stored on the Bit Tag servers the same as tagging on Google Chrome currently works, again via an API as

provided by Bit Tag Technologies. The main data transfer will be done here via an internet connection to and from both the Bit Tag servers and from the requested websites. Both of these types of requests will be made via the HTTP protocol. With this system, Bit Tag Technologies will have further proof of product viability, whilst also having another means of interaction with customers. While the application will not use a native or widespread browser, it will be clear that expanding Bit Tag to further platforms will be a viable option and a possible option.

Our technology integration stack can be seen here:



### Functional Requirements

**User Functionality:**

- This application shall allow users to click on tags that will display the website with that tag.
- This application shall allow users to navigate websites with or without tags.

- This application shall allow users to create tags that will be saved to Bit Tag's servers.
- The application shall allow users to view tags on web pages without needing to login into their account.
- The application shall allow users that have logged in to create tags that contain text and point to elements on a web page.
- The application shall allow users the ability to create Bit Tag accounts with their Google accounts.
- The application shall allow users to login to their Bit Tag account.
- The application shall allow users to logout of their Bit Tag account.
- The application shall allow users to enter a URL into a text field and go to the corresponding web page.
- The application shall allow users to disable Bit Tag functionality if they choose to.
- The application shall allow the user the ability to delete their browsing history.
- The application shall allow users the ability to delete their cookies and clear their web cache.
- The application shall allow users to view all of their created tags on their Bit Tag account.
- The application shall allow users to ability to delete any tags they have created on their Bit Tag account.
- The application shall allow users the ability to edit any tags they have created on their Bit Tag.
- The application shall allow users the ability to view all tags created by other users with Bit Tag accounts.
- The application shall allow users that are logged in the ability to comment on the other users' tags.
- The application shall allow users that are logged the ability to flag other users' tags.

- The application shall allow users that are logged in the ability to save other users' tags.
- The application shall allow users that are logged in the ability to order their saved tags into a collection.
- The application shall allow users that are logged in the ability to view their saved collection of tags.
- The application shall allow users that are logged in the ability to view their saved tags that are uncollected.
- The application shall allow users the ability to send feedback to the Bit Tag development team.

**Application Specific:**

- This application shall be able to display all tags from the Bit Tag servers.
- The application shall be capable of functioning like a regular mobile web browser without using Bit Tag's functions.
- The application shall allow display tags that contains text and point to elements on a web page.
- The application shall be capable to displaying any and all tags on a web page.
- The application shall be capable to display web pages without any tags.
- The application will save whether or not the user has disabled Bit Tag functionality.
- The application shall save the user's browsing history.
- The application shall open the lasted visited web page by the user if the app is initialized with previous browsing history.
- The application shall display Bit Tag's website every time when the app is first initialized if their is no previous saved browsing history.
- The application shall save the user's cookies and web cache.
- The application shall save the user's information onto his or her device once the application is exited or paused.

- The application shall save the user's information onto their Bit Tag account.
- The application shall display which version of Bit Tag is being used to display tags.
- The application shall save the user's login information when he/she exits the application.

## Environmental Requirements

- The application shall be capable of being ported to a variety of different mobile environments.
- The application shall connect to the Bit Tag servers to achieve functionality.
- The application shall require an internet connection to send and receive data.
- The application shall require the user to login to his or her Bit Tag account in order to use Bit Tag functionality.
- The application shall be developed for a mobile environment.
- The application shall conform to the App Store and Play Store acceptance requirements as follows:
  - The application will only use public APIs.
  - The application will not download any code.
  - The application must adapt and display correctly on any device flavor.
  - Any web content must use the WebKit/WebView provided by the mobile platform.
  - The application must not arbitrarily restrict access.
  - The application must comply with the iOS Data Storage Guidelines
  - The application must notify and obtain user consent before collecting, transmitting, or using location data.
  - Any push notifications must be implemented using the Push Notification API of the given provider(better language should be used here).
  - The application must comply with the user interface guidelines provided by the mobile platform.



- The application will only transmit data about a user after obtaining the user's prior permission.
- The application must comply with all legal requirements in any location where they are made available to users.

## Non-Functional Requirements

- The application shall display the tags on a website consistently in the same way.
- The application shall display tags on a page in such a way that it is identifiably the same as the Chrome extension.
- The application shall display tags from the Bit Tag tag repository in such a way that it is identifiably the same as the website.
- The application shall reflect stylistic changes to Bit Tag UI.
- The application shall provide a easy transition between the repository page and the tag page.
- The application shall provide the user with an intuitive user interface to tag various forms of media from across the web.
- The time it takes for the user to create a tag and then display on the tag page should be no slower than the desktop version.
- The user shall be able to add a new tag with a recently taken immediately after that picture is taken.
- The application will not store any of the tagging information on the actual mobile device.
- The application shall run very similarly on all target mobile platforms.
- The application will work on all versions of Android starting at Honeycomb (3.0)
- The application will require an internet connection (3G, 4G, WiFi)
- The application will provide the Bit Tag backend with data such as new tags, geotagging, and account information

- The application shall only need as much storage as the application plus potential temporary memory will take up.

## Potential Risks

Our sponsor, Bit Tag, is a new company so this project has the potential to not be the optimal solution to some of their problems that they are looking for which would in turn cost them time and research which is something that is very valuable to them.

One of the major risks of this project, specifically for Bit Tag, is that this mobile application turns out to be something that users find redundant and useless compared to what they can do within the browser or that this application just doesn't find its place in the mobile space. This could be very problematic because it could cost a large amount of the current user base to be turned off to the mobile side of this company or just lose potential people that would be interested in only the mobile application of Bit Tag.

Another possible risk is that mobile users find no interest in this application because the application would not be able to support your normal mobile browsers like Chrome, or the default web browsing applications. This could be impactful for similar reasons as before. You may have some people completely turned off to the mobile experience.

One of the possible minor risks we see as a possibility is parts of the functionality of our application having no usage whatsoever. This could be whether no users use the web searching often, or even visit the Bit Tag homepage at all. This could be a problem because it would not keep users using our application. The more functionality that they are not using means the less they are using Bit Tag's application.

A more technical-related risk could be that WebView only has access to a single OS thread. If you want to want to do a lot of tasks with it you are going to see that the application will not be very smooth. This could easily detract from the user's experience when using the application and will make the overall experience less than great. One way to combat this would be to offload as much processing to the backend as possible.

## Project Plan

The project plan thus far is as follows.

1. Tech Feasibility Report - This document will outline the technologies we intend to use for the project and show to a degree that these choices are viable.
2. Requirements First Draft - This document will outline the project and its requirements in detail. This is subject to change.
3. Initial Design Review - In this we will present our project so far, outlining the details, the technology, and the requirements of the project.
4. Requirements Final Draft - This document will outline the project and its requirements in detail. This is the final copy of the document.
5. Initial Learning of Technologies - In this phase we want to make sure the whole team understands what is required to use the decided technologies
6. Request APIs and Libraries From Bit Tag - In this we need to get all the documentation and libraries required of the project from Bit Tag.
7. Pull/Generate List of Tags Into a View - In this we will pull required data from Bit Tag servers and populate our app.
8. Implement Viewing of Bit Tags - In this we will implement the extension functionality of Bit Tag.
9. Product Integration - In this phase we will integrate both sides of the app into a functional app.
10. Product Styling - In this phase we will ensure the stylization of the app is valid and follows designated design principles for the selected platform(s).
11. Product Delivery - In this phase we will deliver the product to Bit Tag.

## Glossary

### **Chrome Extension:**

Extensions are small software programs that can modify and enhance the functionality of the Chrome browser. You write them using web technologies such as HTML, JavaScript, and CSS.

**Push Notification:**

Push notification, also called server push notification, is the delivery of information from a software application to a computing device without a specific request from the client.

**Front End:**

In this context it is our application.

**Back End:**

In this context it is the Bit Tag servers.

**Appendix****Apache Cordova:**

<https://cordova.apache.org/>

**App Store Review Guidelines:**

<https://developer.apple.com/app-store/review/guidelines/>

**Bit Tag:**

<http://www.bittagbox.com/>

**inAppBrowser:**

<https://www.npmjs.com/package/cordova-plugin-inappbrowser>

**WebView:**

<http://developer.android.com/reference/android/webkit/WebView.html>

[https://developer.apple.com/library/ios/documentation/UIKit/Reference/UIWebView\\_Class/](https://developer.apple.com/library/ios/documentation/UIKit/Reference/UIWebView_Class/)