User Testing:

Native Plant Field Guide App

for a

Phenology of Place at UVic

Awesome! This page is mobile-friendly.

By Wendy Anthony
for University of Victoria ES 490 Directed Studies pt. 2
April 2015

Supervisors:
Dr. Eric Higgs, Environmental Studies
Dr. Yvonne Coady, Computer Sciences

Introduction

This document represents a pictorial tour of the design development phases of the Native Plant Field Guide App. Each development phase started with a fully functioning app, which had some additional functionality added to it in increments, based on research, learning new code, user testing, and experience through trial-and-error.

In order for a web application to be useful to users of the target audience, user testing needs to be done on a regular basis, at all stages of design. By starting with a working prototype, and adding functionality in stages, the end product is always a functional version with either new features added, or with working code replaced with more elegant coding solutions to achieve a similar, but more efficient result.

As different users will likely also use different devices and screen sizes, using responsive design techniques will reduce the need to create a different version for each different screen size.

The following document summarizes the different versions, along with the design challenges presented. Test user suggestions have also been listed, as well as the design changes made as a result of user testing, and the wish lists for future design ideas.

Table of Contents

Table of Figures

I. Field Guide App Concept Idea & Background

This Native Plant Field Guide App is a result of research into methods to use seasonal phenological photo-based data, about native plants growing at the University of Victoria (UVic) campus, as a Directed Studies into Phenology of Place at UVic. The intent was to create a web-based application that can be used for both educational and research purposes.

The data gleaned from a collection of digital photographs of native plants were organized within a FileMaker Pro database, and then exported in code scripts to be used in web page templates, and using Java Script code, to create a functional web application design that was responsive to a variety of computer devices sizes, including smart phones, tablets, laptops, and desktop computers.

The code design development was based on incremental iterations, adding code in small increments to create a fully functioning version, before adding any further functionality. With each new iteration version, the design, function and use of the app would be tested by potential users, changed according to user feedback, and tested again.

The original target audience has been identified as being UVic students (Environmental Studies, Biology, Forestry, Environmental Restoration), restoration volunteers, outdoor recreational enthusiasts, and newcomers to the University or area.

The companion document *A Project Manual: Using Native Plant Photographs and Digital Technology as Tools for Education and Research* will provide more detail, and code examples, about the various design elements tested in this document.

II. Design Issues & Development Processes

User-Centered Design

- Critical to designing web applications that are under development is to know the audience who will use the app, and to solicit user feedback (Verba, 2008).

Prototyping

- Basic design wireframes will sketch layout and content but not interactivity. Prototypes are a working example of the design, which allow the user to interact with the design features and provide feedback (Verba, 2008). When the prototype is a fully functioning application, the test user will get a true feeling of how the various elements work during common usage, and allow for opportunities to test redesigns.

Agile, Incremental Development

- Each version is a functionally working app before adding any further code modifications. After research, recoding, and retesting, the next functional version is ready to adapt and change as future needs are addressed and added.

Responsive Web Design

- Different users will use different devices, at different locations. To avoid creating multiple versions of the same design, using a responsive design (e.g. using proportional units instead of absolute units) will allow the app to adapt to different screen sizes, and allow for a more user-friendly experience.

Testing

- Invite users to test and make suggestions early, and regularly throughout the design process, if a web application is to be truly useful (see Appendix I). This allows for opportunities to adapt and change designs based on real user needs and experiences. Future changes and additions also need to be tested to have a better chance to provide the functionality that actual users will find useful. Invite test users to presentations, and thank them for their participation (see Appendix II).

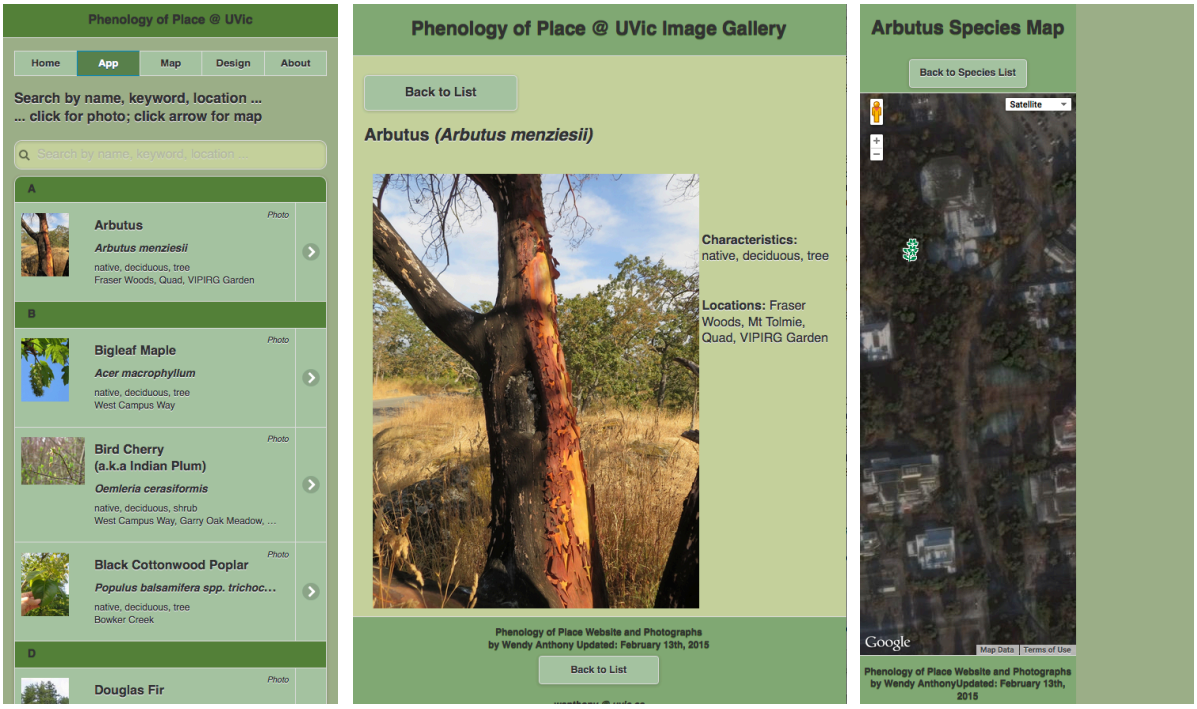
Versioning

- Each working version is saved in an archive before any further changes are made, to prevent new changes from breaking working code, or to save simpler versions to revert back to when additional functions are added that may need to be changed or removed at some future time. Using on-line versioning software (e.g. Git) to host each of the versions, can create a handy repository, as well as an opportunity for other developers to make contributions to the app's code.

III. Design Versions

Design Version v. 3.4 - Best Current Working Version

- The current, most stable working version includes a search bar, a link to an image, species info (which can be searched), and a location map (see Figure 1).
- This design is Mobile-friendly, by being responsive to different screen sizes, as well as using JQuery Mobile List View and its search function, along with large button sizes for use with fingers as well as a mouse.
- The first species image (Arbutus) is actually loaded as part of the original page, which reduces page load time, though each of the other species are linked to images which are hosted on a separate webpage for each species (see Figure 1b).
- The map is linked to a separate file, which is populated by a GeoJSON file, exported from the database, for coordinates and species information (see Figure 1c).
- **Challenges:** The image and map pages are all individual files, which would need to be maintained separately. Ideally code would populate the image and map pages automatically based on the choice of species from the List View.



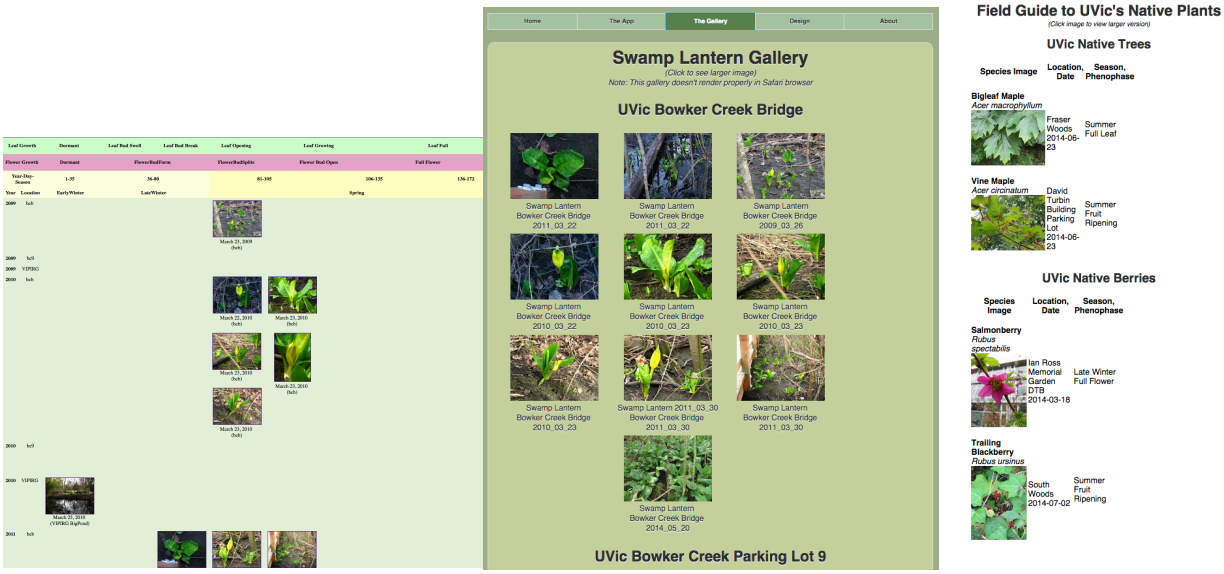
***Figures 1 a,b,c*:** v. 3.4 best, most recent, stable version.

Design Version v. 1.01 to 1.03 – Web Image Galleries

- Developed as a way to showcase native plant photos in some systematically organized seasonal time-line to enable any patterns to be seen.

- These image galleries were seeds for the idea to create a database to easily sort plant images by species, date, season, location, and a web-base application for purposes of identification of species and phenophase growth stages.

- **Challenges:** The first version resulted in a very wide web page that needed scrolling even on a large screen (see Figure 2a). Further attempts used a flexible table (see Figure 2b and 2c), and a JSON data template (see Figure 2c), which were more responsive to different screen size without resorting to a sideways scroll.



***Figures 2 a,b,c*:** v. 1.01 to 1.03 – Native Plant Web Photo Galleries.

Design Version v. 1.1 to 1.3 - inside "Lively" developer environment

- Versions 1.1 through 1.3 were developed as an off-shoot of a CSC130 Fall 2014 course project Music Player app, by adapting the code to develop drop-down lists which retrieved species image and information (see Figure 3).

- These versions were originally programmed inside the "Lively" programming environment, and after considerable re-tooling, the JavaScript code was designed to exist outside "Lively" as a stand-alone webpage.

- **Challenges:** Coding and saving inside a programming environment allows access only through a login procedure resulting in an awkward public access. The working code could not be exported, and thus needed to be completely rewritten for use outside the coding environment.

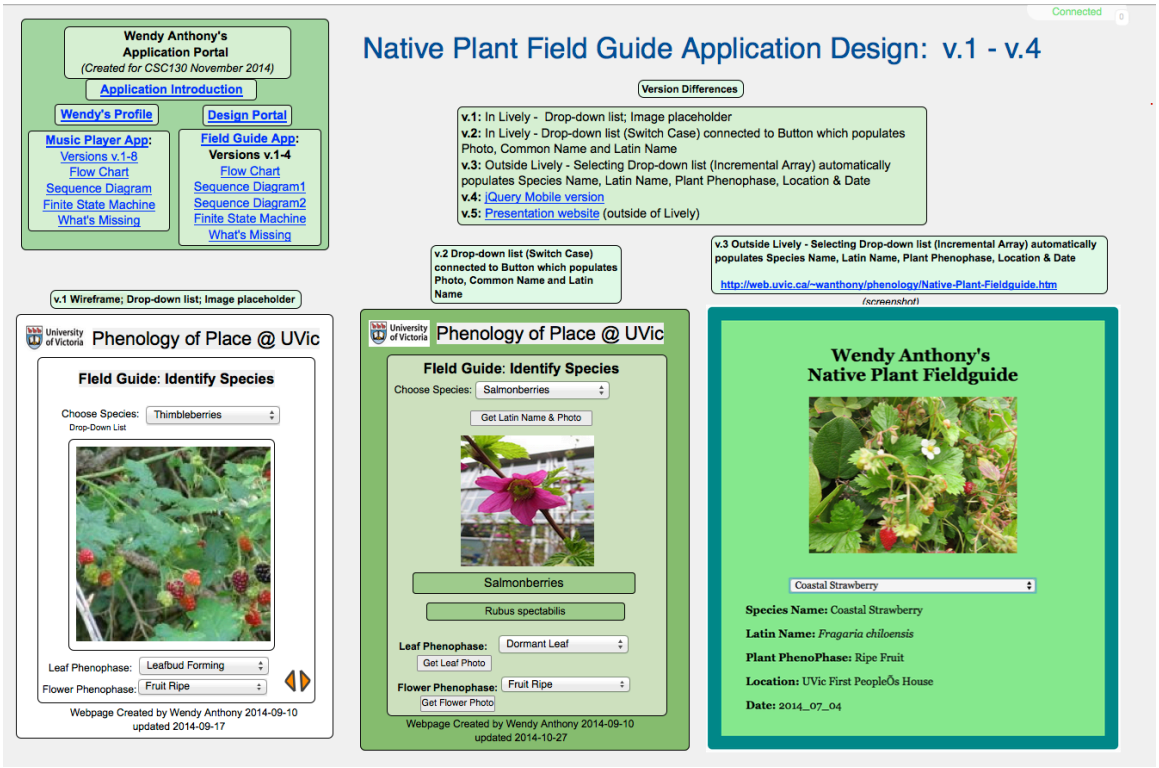


Figure 3: v. 1.1 – 1.3 in Lively programming environment, v. 2.1 Outside of Lively.

Design Version v. 2.1 & 2.2 - On the Web

- Though the field guide app Java Script code was eventually recreated outside the "Lively" environment, v. 2.1 was not responsive to changing screen sizes (see Figure 4).

- After researching responsive web design, v. 2.2 was redesigned to be responsive to changing screen sizes, though the image was very large in a full-sized web browser, and the drop-down lists and font sizes were quite small in a smaller browser screen sizes (see Figure 5).

- **Challenges:** Though a prototype app had been coded within the "Lively" environment, the code could not be exported and required an entire re-write. Responsive changes need to address image size and placement as well as font size.

Figure 4: v. 2.1 App not responsive to screen size.

Figure 5: v. 2.2 Responsive to screen size.

Design Version v. 2.3 - Responsive To Screen Size

- After more code research and testing, the responsive design was tweaked so the image stayed a reasonable size, and the font remained legible.

- **Challenges:** Though the drop-down lists worked fine when using a mouse to navigate, they were much too small to be used by a finger in mobile devices. The images also ended up too small when responding to smaller screen sizes (see Figure 6). Research into different ways to present lists for users to choose from will be necessary to make the application user-friendly on different devices.

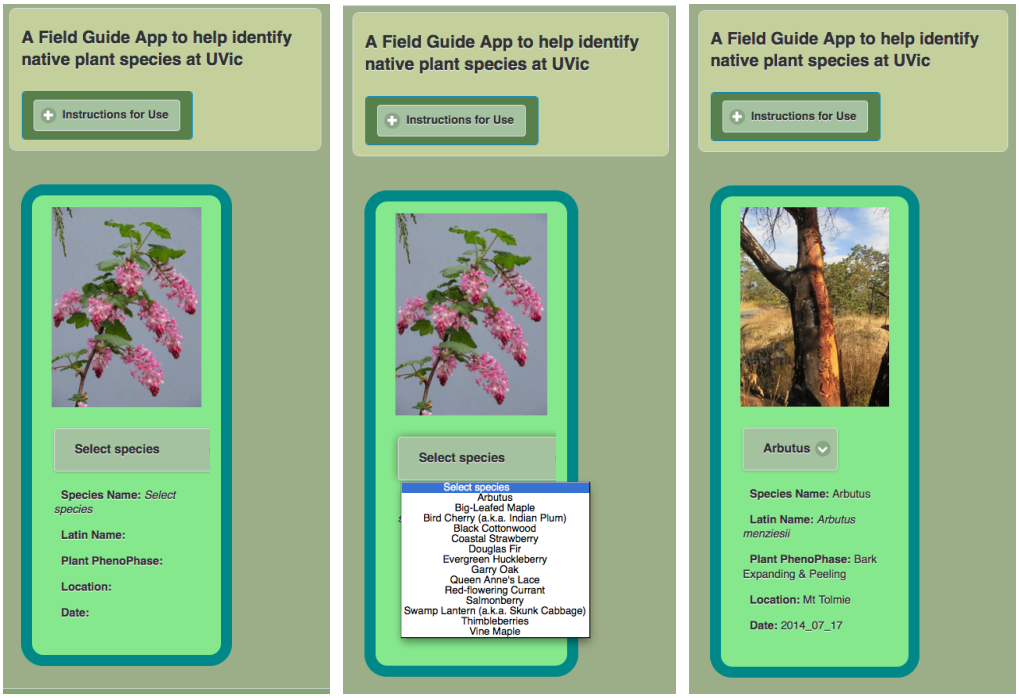


***Figure 6**:* v. 2.3 Closed state; select list open; selected species image & info.

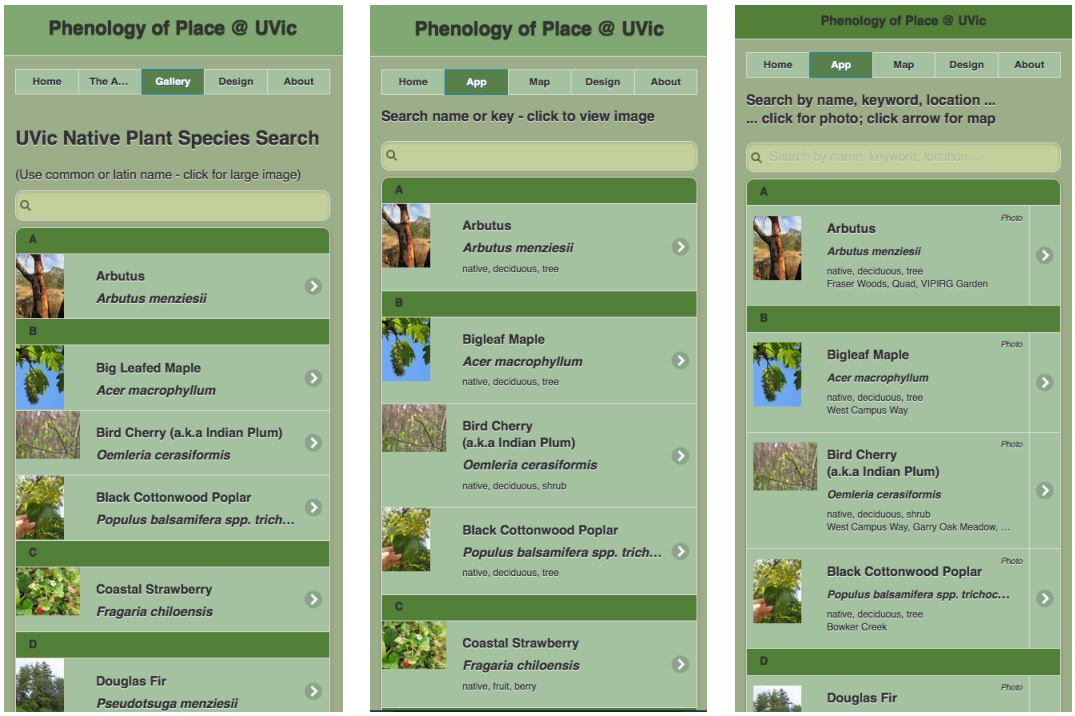
Design Version v. 3.1, 3.2, 3.3 - using jQuery Mobile JavaScript format

- Version v. 3.1 was the first attempt using jQuery Mobile library List View format to make the app mobile-friendly. The Common Name and Latin Scientific Name are listed, and linked to the species image file when the List View item is clicked (see Figure 7a).

- The search feature can be added to the mobile list view, activated by searching any key words appearing in the list view. Version v. 3.2 added more species information to make the Search function more helpful when identifying different species (see Figure 7b).

- Version v. 3.3 added the species location to expand the Search function, and adding an `li-aside` provided another link option, from the List View to a map of the where a species photo is located (see Figure 7c).

- **Challenges:** In order for the species information to be searchable, more details need to be added to the List View, expanding the section to be larger than the thumbnail image.



***Figures 7 a,b,c*:** v. 3.1, 3.2, 3.3: jQuery Mobile List View, species info, photo & map.

Design Version v. 3.1 - links to image file

- Version v. 3.1 is directly linked to the image file, with no need to create a separate webpage for the image (see Figure 8).
- A thumbnail version of the same image file is place in the List View.
- **Challenges:** As more species information is added, the list view container increases in height though the thumbnail image remains the same size. The thumbnail version of the image will need some formatting to establish a thumbnail size that is appropriate to the size of the list view height.

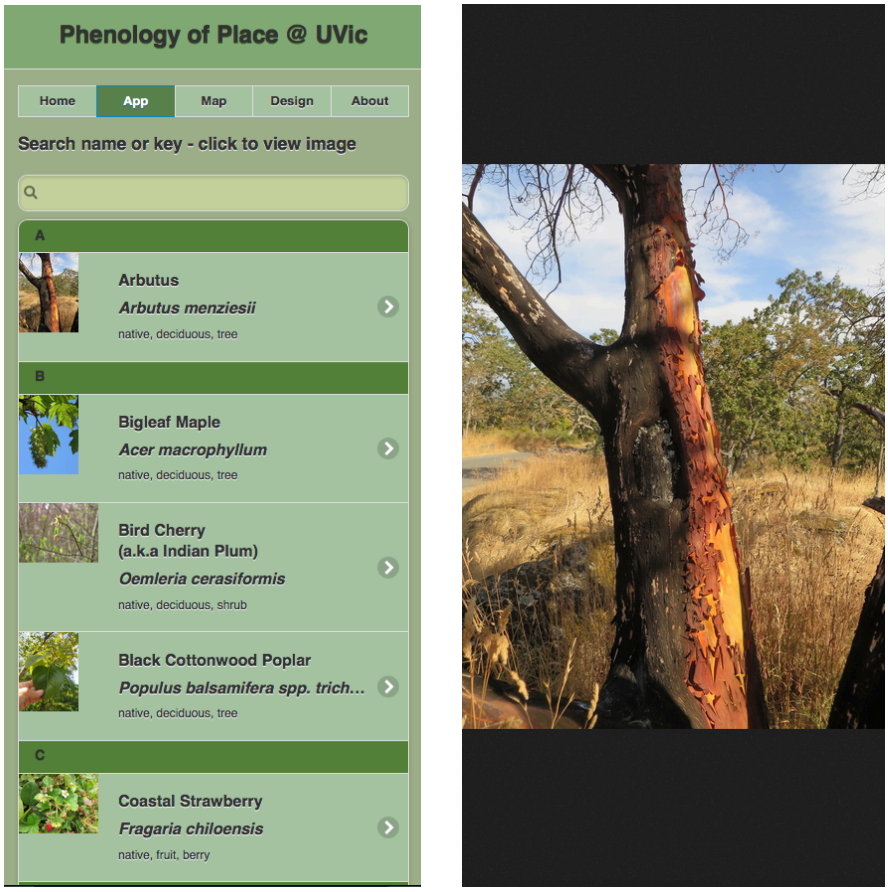


***Figure 8*:** v. 3.1 Each list item links directly to image file.

Design Version v. 3.2 – No Back Button

- Version v. 3.2 adds a `li-aside` to the List View, enabling a secondary link to a species map page (see Figure 9).

- **Challenges:** Neither the linked image nor map have back buttons, which makes navigation difficult for the user to easily return to the List View page.

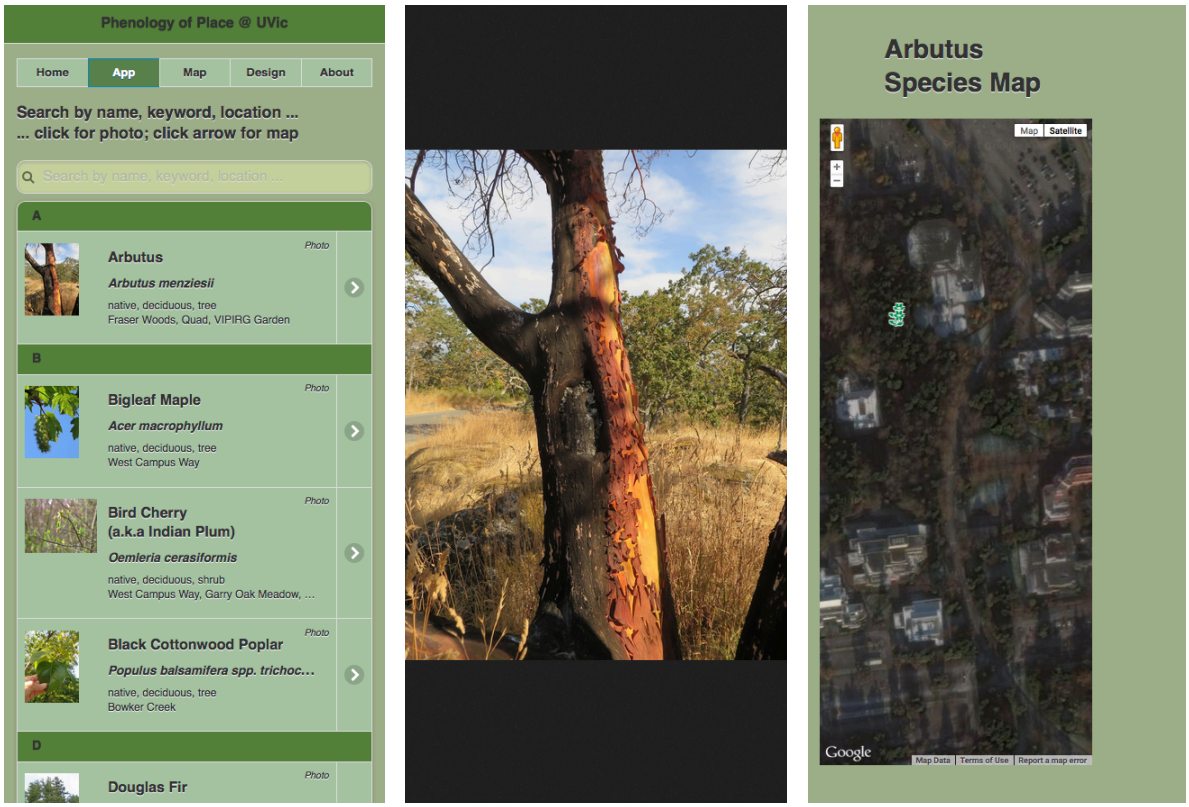


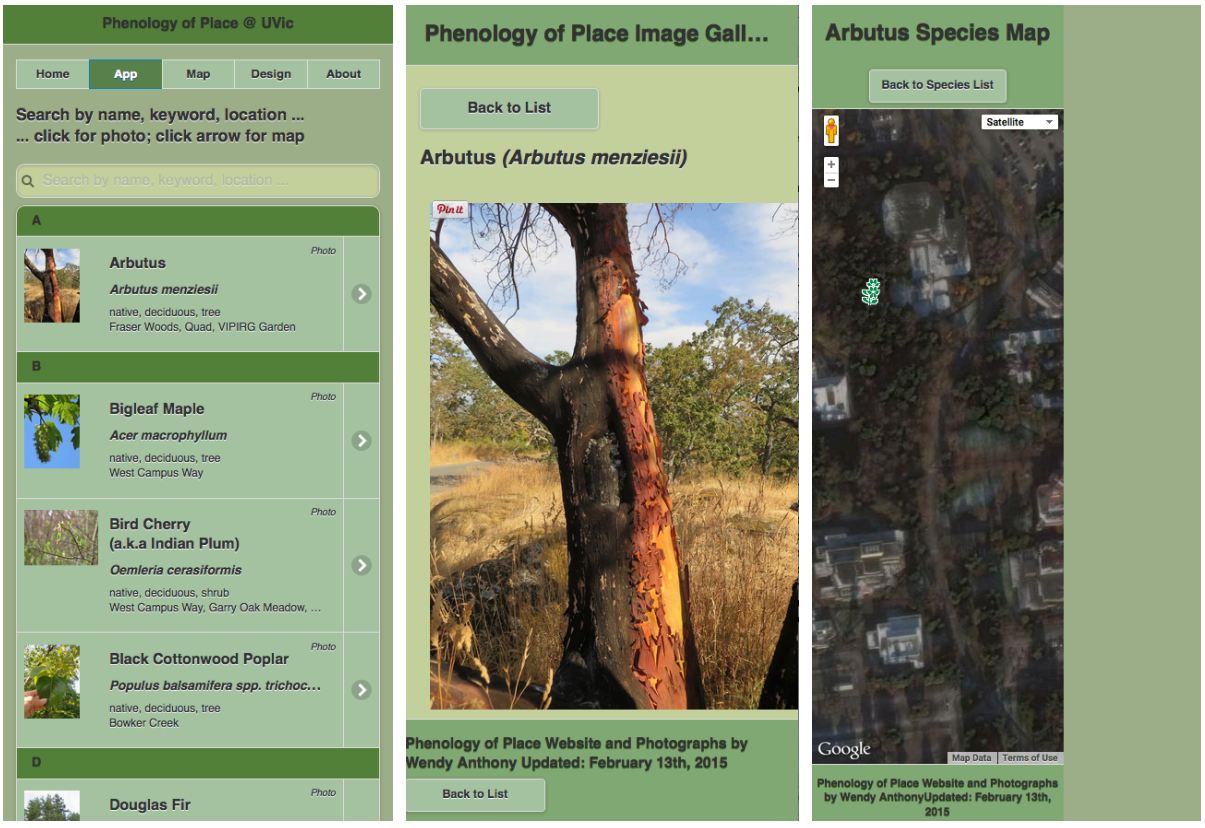
***Figure 9*:** v. 3.2 No back buttons to return from the image or species map.

Design Version v. 3.3 - with Back Buttons

- After viewing user frustration trying to return easily to the List View from both the image and the map, a Back Button, large enough to be easily used on a smaller mobile device, was included on both the image and map pages (see Figures 10a,b,c).

- Only one of the Image pages is actually included on the original List View page as a second page, while all other images have a separate webpage (see Figure 10b); the Map page is a separate webpage linked to from the side link on the List View (see Figure 10c).

- **Challenges:** JavaScript (jQuery Mobile) code is needed to allow different images to be linked from the same second page to enable the images to load faster. Using separate webpages for each of the images, and each of the maps results in a longer time to download each individual page. Each individual page would also need to be maintained separately if any changes were made to the website design.



***Figures 10 a,b,c**:* v. 3.3 Back Buttons return to List View from Image and Map pages.

Design Version v. 3.4 – more detailed species information

- After user requests to include more species information, which would overwhelm the small List View section, more species information was included on each species image webpage (see Figure 11).

- **Challenges:** The information on this page cannot be searched from the Search tool on the main List View page.

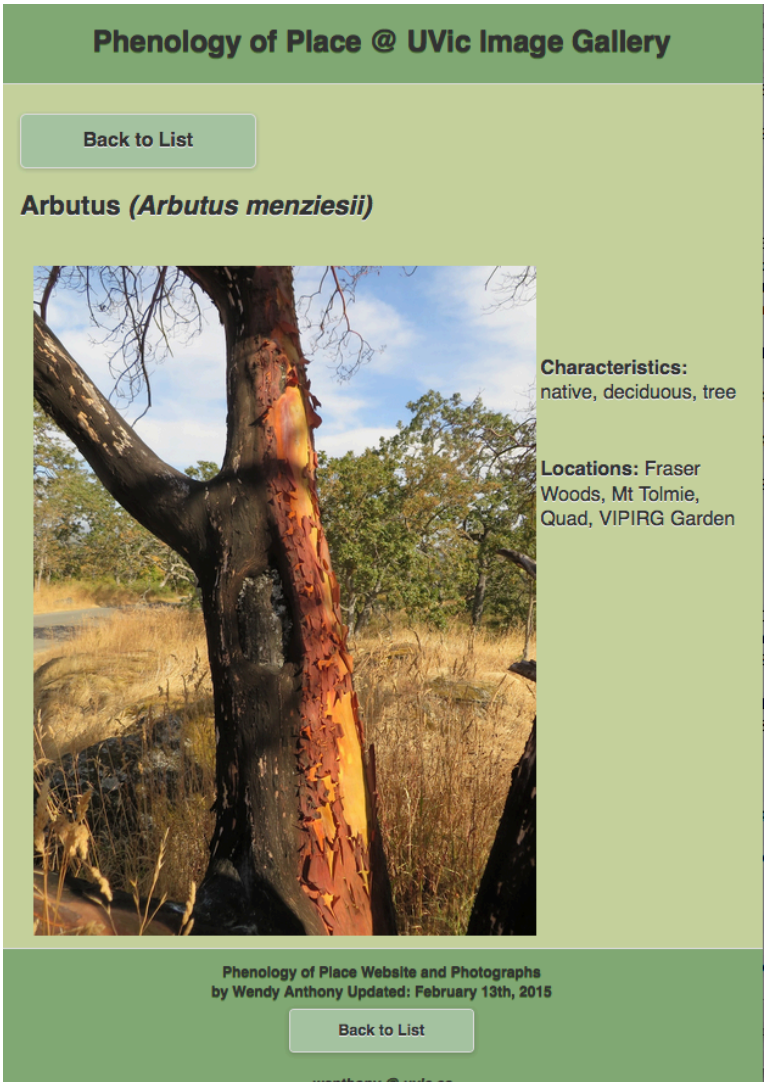


Figure 11: v. 3.4 Photo page with species info.

Design Version v. 3.5 – Phenophase Slide Show

- In keeping with the Phenology theme, users requested to see images of the species in different leaf and flower phenophase growth stages, in order for the plant to be recognized in different seasons.

- Swamp Lantern is the only example where the Image webpage contains a slideshow of different phenophases, (see Figure 12).

- Left & Right Arrows are used to navigate between the phenophase photos; the Back Button returns to the List View page.

- **Challenges:** Finding photos of different phenophases for each of the species included in the field guide. Web page loading time will be affected by the number of photos to retrieve.

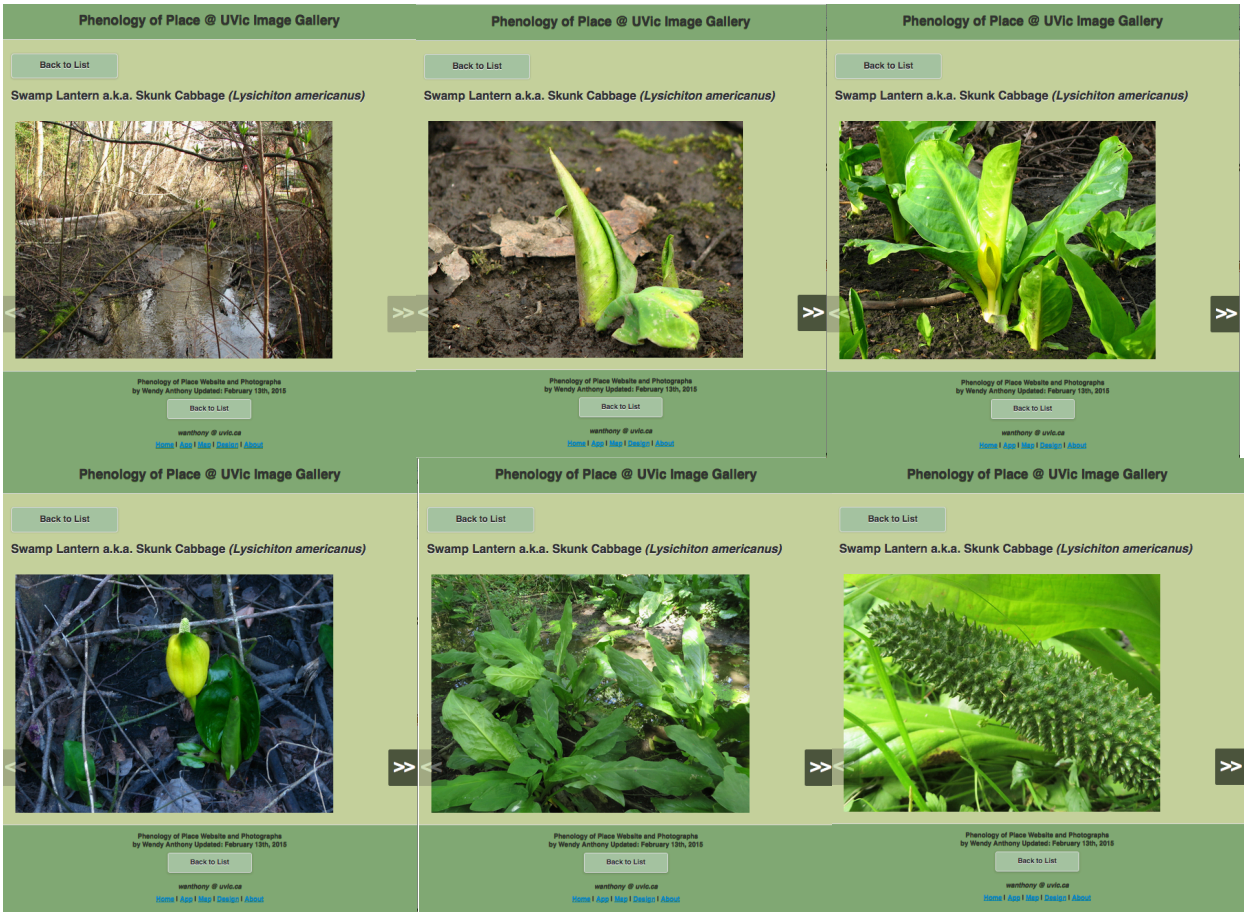


Figure 12*:* v. 3.5 Slide Show of Swamp Lantern phenophase photos.

Design Version v. 4.1 and 4.2 – JSON populates List View

- In order to find a way to populate the List View automatically from the database, a JSON file, consisting of data exported from the database, was used to populate the Species List View, instead of having to separately type in the information for each species in the List View.

- The first version v. 4.1 ended up being a run-on list of each of the species as found in the JSON file, making the user scroll to see beyond the first screen, without enabling a search feature (see Figure 13).

- Version v. 4.2 uses a JSON file to populate the original List View, including the image thumbnail, and though the species information in the list is not responsive to changing screen size, the information is searchable (see Figure 14).

- **Challenges:** Including enough information to be searched in limited space.

Figure 13: v. 4.1 JSON populates webpage.

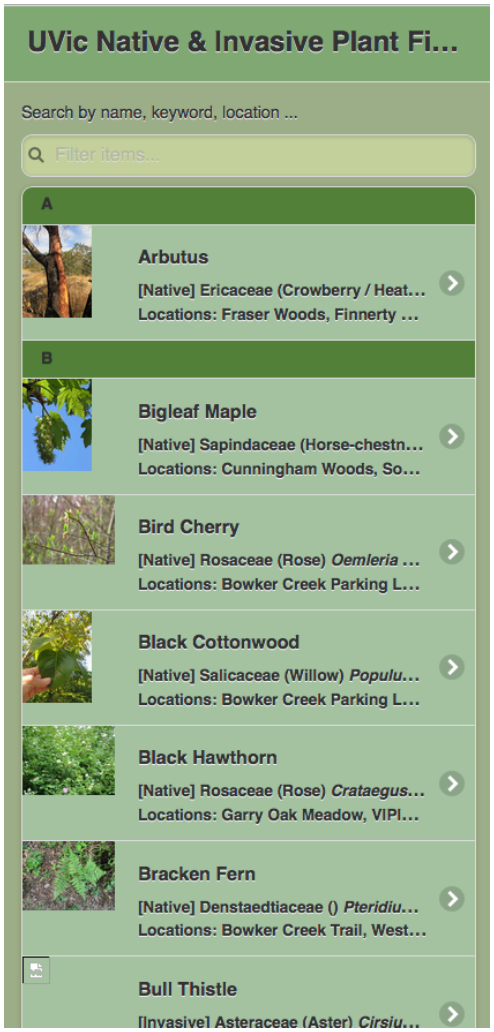


***Figure 14*:** v. 4.2 JSON populates List View.

Design Version v. 4.3 - JSON populates List View – using a different database

- The code proved to be transferable when used to create a similar searchable List View in version v. 4.3 for the UVic Herbarium database (see Figure 15).

- **Challenges:** As only a small amount of information can be included on the initial List View page without breaking the screen responsiveness, finding a way to format the information in the List View, as well as how to search the information on the details page would further enhance the usability of this app code.

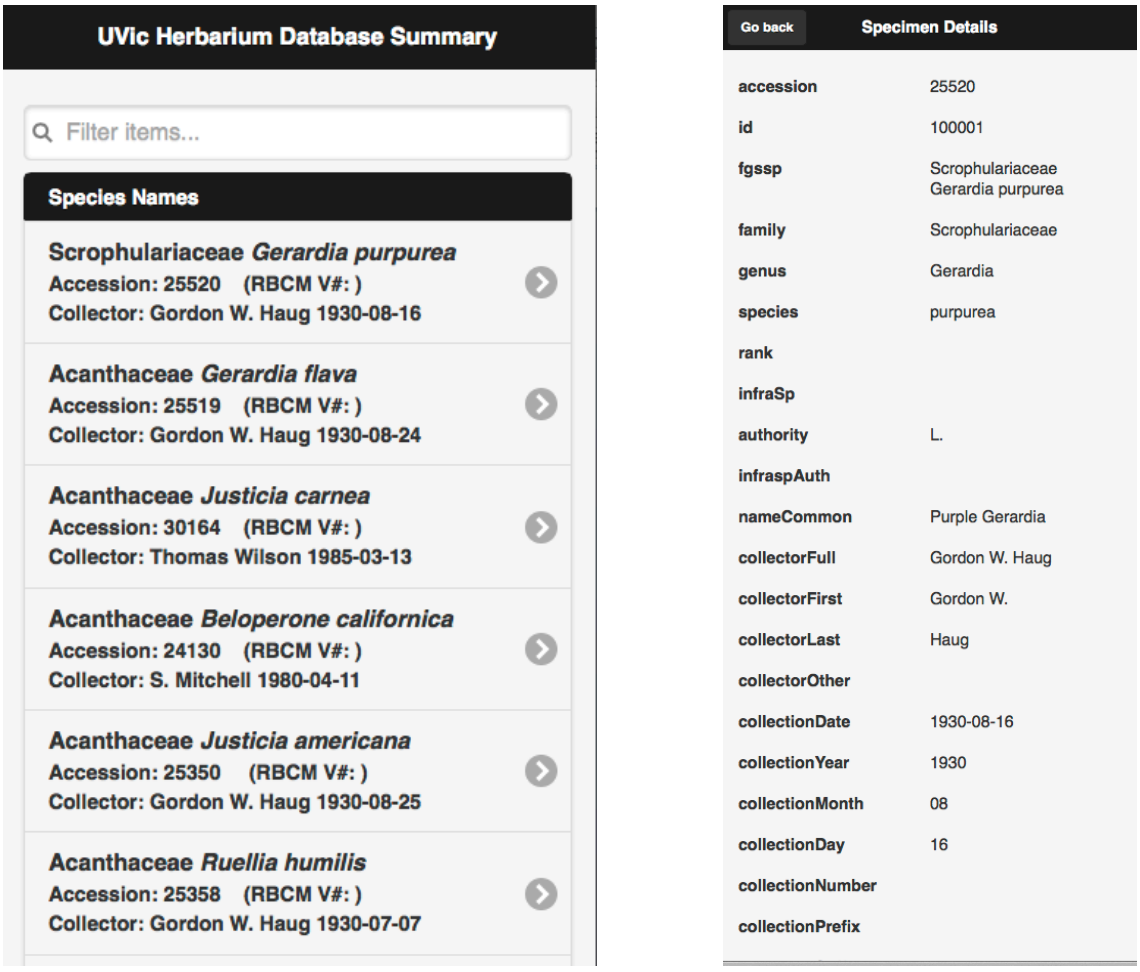


Figure 15: v. 4.3 Similar coding scripts re-used to create an online List View for UVic Herbarium.

IV. Spatial Components

Species Location in Google Maps

- The first working code version imported location data for the 900+ photos in the database (see Figure 16) into one map; a separate map was eventually created for each species (see Figure 17). A GeoJSON file, with detailed information including lat/long coordinates, was exported from the database, and used to populate the map location Markers and Info Windows.

- An image of a flower was used to replace the default Google Maps Marker icons.

- **Challenges:** Multiple location markers proved too difficult to determine which marker to click; using cluster markers for each species and location would limit the visual clutter.

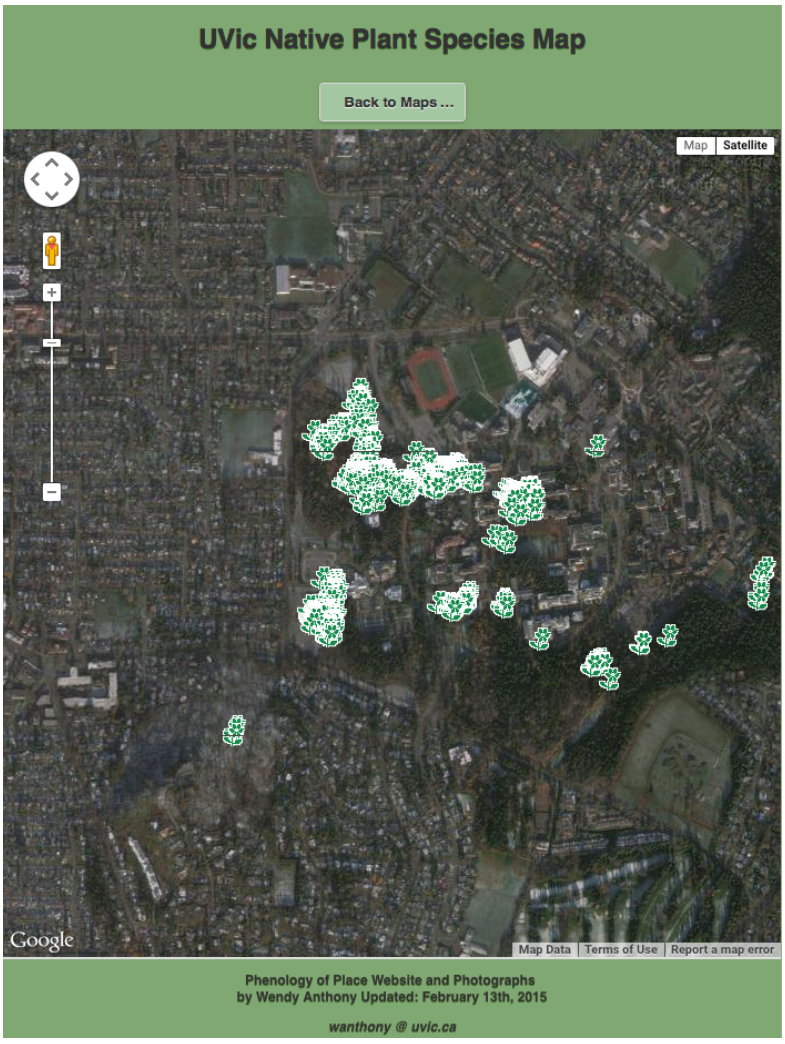


Figure 16: Location map for all species in database.

Info Window for Species Location in Google Maps

- Each species has a Google Map showing plant photo locations (see Figure 17).
- When a location marker is clicked, an Info Window opens, with details populated from a JSON file exported from the database (see Figure 18). Each Info Window is a template containing the following information: common & Latin name of species, location of photo, image thumbnail with link to larger image, date, season, flower & leaf phenophases, and minimum and maximum temperature, precipitation, and sun rise and sun set for the date of photograph.
- **Challenges:** Finding a way to make species information in the Info Windows relevant to the users of the field guide app. The weather information and day length data that is included is based solely on the date of the photo. Some Info Windows won't show an image as not all images have been uploaded to the server.

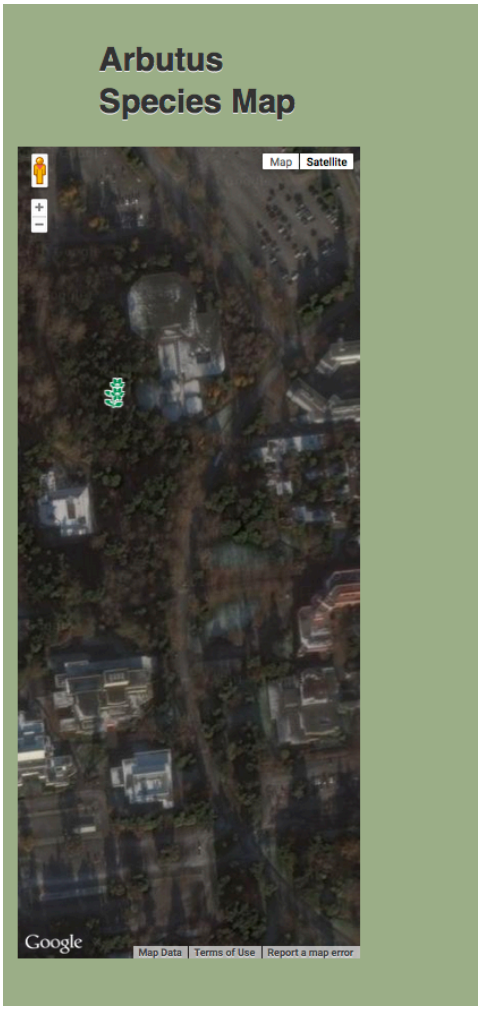


***Figure 17**:* One Species Location Map.

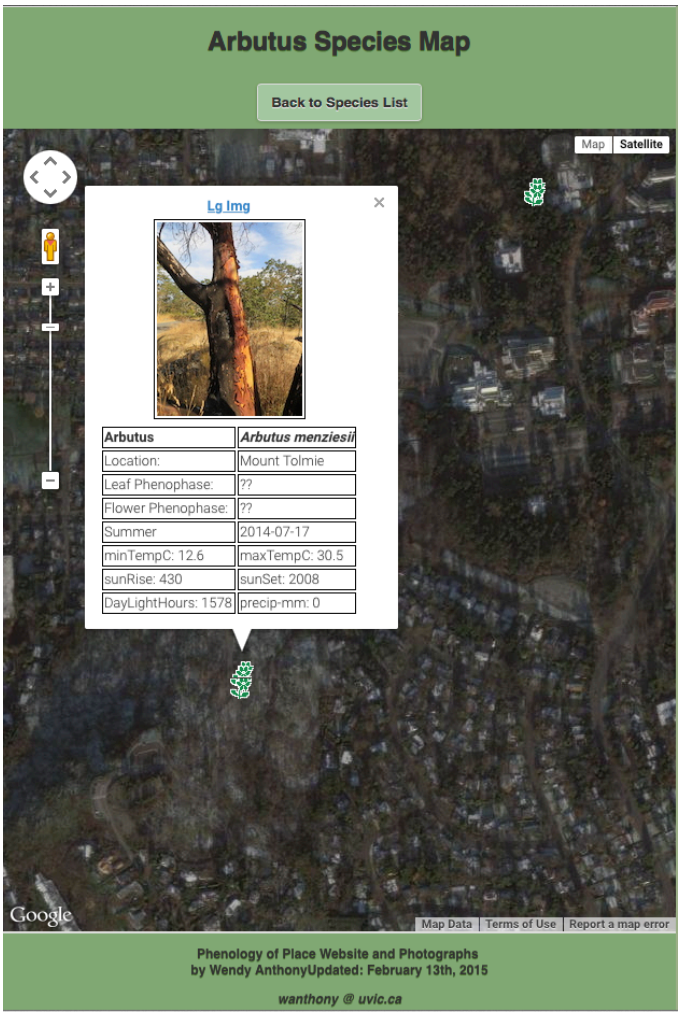


***Figure 18**:* Species location map with open Info Window.

Finding User's Location

- Through the browser, HTML5 coding allows the user's location to be shared after asking user permission (see Figure 19).

- Once consent is given, and the location is found, the latitude and longitude are placed with a marker on a Google Map embedded in the webpage (see Figure 20).

- **Challenges:** As users will need to trust the app before allowing their location to be shared, linking to a privacy policy can detail what the location data will be used for. Finding a way for the user's moving location to be mapped in relation to a species' location would further augment this app on mobile devices (see Figure 21).

***Figure 19**:* User permission required to share location.

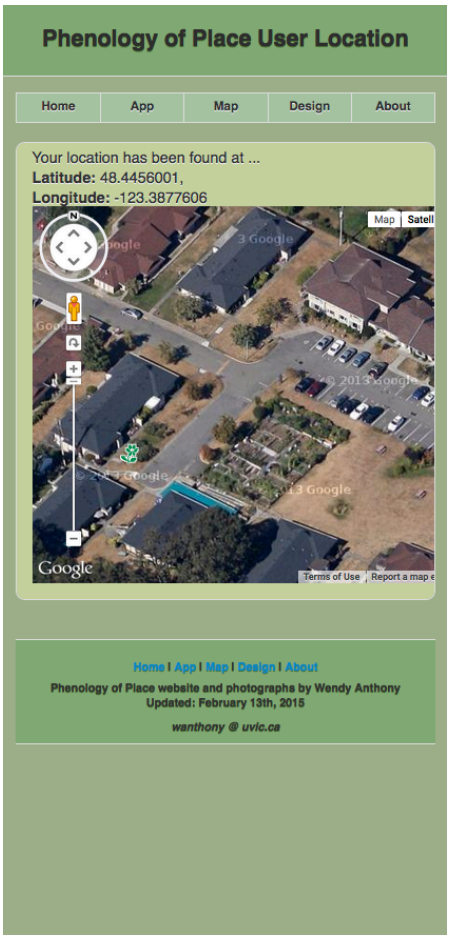


Figure 20: User's location found.

V. Testing the App

Purpose of User Testing

- View, use, provide feedback
- Class Field trip to South Woods - plant ID
- Test on iPhone 6
- Consider how to make useful for other user groups
    - e.g. adapt JSON populated List View to Herbarium data

User Testing Suggestions

- Be sure to correctly label Native and Introduced (non-native) plants
- Make note if plants have edible or poisonous parts
- Add more photos of different phenophases
- Order List View by common name, adding Latin scientific name for extra info
- Add Back Buttons to photo & map pages to make navigation easier to return to Species List View
- Get lots of Testing done by lots of Users
- Use versioning control software e.g. Git
- Using List View to link to photos of different phases and plant parts, and different levels of explanations, e.g. like iBird
- Don't show everything on map – let user explore to find species
- Add section "What to expect to see When/Where", using a seasonal chart and map
- Add Glossary – don't use technical terms (e.g. phenophase)
- Explore potential for possible use as framework/template by other user groups
    - UVic Herbarium database (see Figure 15)
    - WSANEC Tribal School
    - Sea Change Restoration
    - Pacific Northwest Plant Knowledge Cards
- Add crowd-sourced, user uploaded photos of different phenophases
- Add other info
    - Traditional use
    - Compare to similar species, especially Native vs Invasive
        - e.g. Oregon Grape vs Holly, Trailing vs Himalayan Blackberry
- Make downloadable app to use in field without Wi-Fi
- Include other locations with similar species
- Include other locations with native species found locally, but not at UVic
- A list view of records in database, aggregated by species
- Give users app's webpage URL address to check the app out more thoroughly
- Make app accessible through UVic website to be available for student use
- All users appreciate that the app is responsive to different screen sizes
- Add watermarks to photos, and if considering the future marketing potential of the app, don't offer all photos in the pilot project version

- Use proximity of user's changing location on Google Maps in relation to species location marker when user is moving along a trail (see Figure 21)

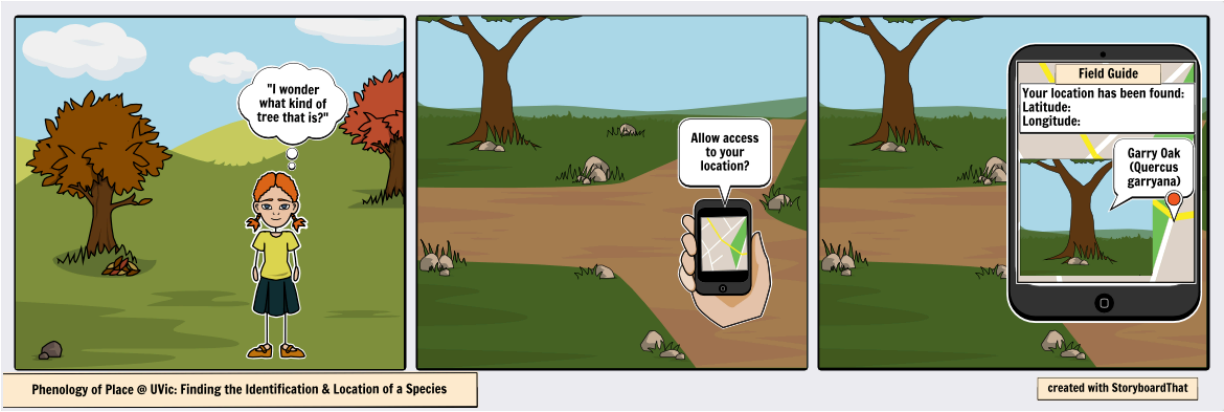


Figure 21: Wish List: utilizing user's location to find proximity to species using Google Maps.

Changes Made After User Testing

- Removed Coastal Strawberry (introduced, not native) from app species list; added a new field in FileMaker Pro database Species table to differentiate between native, non-native (introduced) and invasive
- Swamp Lantern phenophase slideshow is linked to the list view photo link
- Got lots of user input, using Agile Development to make changes and get other users to test the incremental changes
- Added back button to image and map webpages
- Created a responsive webpage to request & show user location on Google Maps
- When automated code was used to populate species data information, it was formatted for each List View item to not take up too much room on the screen
- Added a web user statistic counter to each of the app pages to help determine use by test users

Testing Devices and Browsers

- iMac, MacBook Pro 13" laptop; Windows 7 PC
- iPhone 6; recent model Android smartphone
- Chrome, Firefox, Safari, Explorer 13

Testing for Mobile-Friendliness

- Using Responsive Web Design, only one website needs to be developed that will create a responsive website design which can be viewed on desktop computers, tablets, and mobile smart-phones.

- When searches are made from mobile devices, Google Search is adding the label "Mobile-friendly" for websites which have optimized their content to be viewed on both desktop and mobile computers, with minimal scrolling (Selby, 2014).

- After submitting each of the URL links for my Native Plant Field Guide App list view, photo and map pages to Google Developers' Mobile-friendly test webpage, I received the message *"Awesome! This page is mobile-friendly"* (see Figure 22) (Google Developers, 2015).

- **Challenges:** Though the app elements size properly for the smart phone display size, the font size is too large in the header, the search text, and the Common and Scientific names. These font sizes could be adjusted using different CSS media rules for different device sizes (see Figure 23).

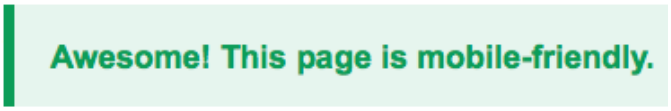


Figure 22: Google Developers' mobile-friendly message.

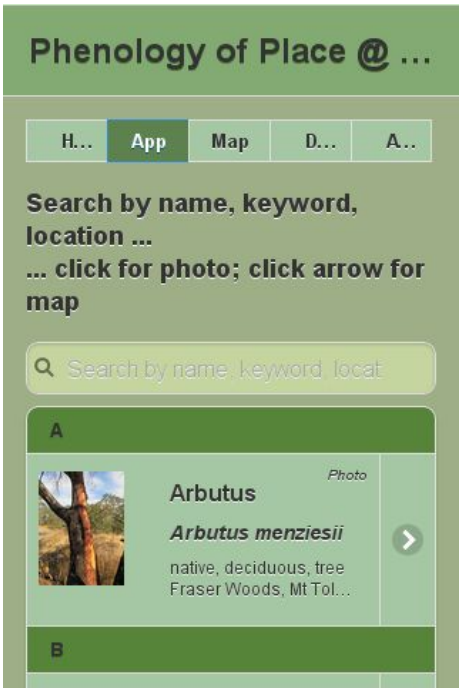


***Figure 23*:** How Google Bot sees the app.

VI. Challenges & Wish Lists

Challenges Resulting From User Testing

- Not enough photos in database for separate phenophase photos for each of the species on list
- Learn how to use versioning software repositories, like Git Hub
- Returning to previous page after viewing image file
    - Image was no longer responsive after adding back button
    - Separate web page needed for each species image - any change requires changing each page
- Still looking for code for getting location to respond to user movement
    - Would like to show user's proximity to species
- Get valid JSON format
    - Add JSON script to database so each record is automatically converted to JSON object format
    - Use JSON to populate species photos
    - Use JSON to populate species data, formatted to not take up too much room
- Aggregate common species name in accordion list showing number of items nested within
- Creating methods and protocols for adapting field guide app code for other users e.g. UVic Herbarium (see Figure 15)

Challenges Resulting From Design & Code

- Different devices
    - Responsive design
        - Images, Maps, Webpages
    - CSS Media – create screen size breakpoints for style changes
    - Handles drop-down lists differently
- Adding new features from user test feedback
    - Before asking for more user testing
- Code
    - Getting the code to work for the features I want to add
    - JavaScript/JSON use to populated one page versus one-page/species
    - JSON file needs to be on the same server for JQuery Mobile & AJAX to work
    - Pages won't work without JavaScript enabled
- Finding a way to visually portray phenological data, along with weather and day length data, in a meaningful way

Wish List Resulting From User Testing

- Add more species photos
- More phenophase photos
- Photos from different locations at UVic
    - Mystic Vale, Mt. Tolmie, UVic Arbutus – Finnerty Gardens parking lot
- Photos from different locations with similar species
    - Beacon Hill Park, Uplands Park, Royal Roads University
- Photos from different locations with species not found at UVic
- User input
    - Use current location for Google Maps trail guide
    - Upload crowd-sourced photos from smart phone to contribute to database
- Add Map for What to expect to see When/Where
- Google Maps location proximity based on movement of user and device
- Add other info e.g. Trad use, compare to similar species
- Make downloadable app to use in field without Wi-Fi
- Use JSON to populate species photos
- Add Glossary – don't use technical terms (e.g. phenophase)

VI. Conclusions

Repeatedly requesting user testing, as each additional function is added, will help ensure that the app will be meaningful, usable and effective.

Keeping track of design versions, at each stage of the development process creates an effective way to keep track of design changes, as well as providing working copies to fall back on if future versions don't work as planned.

Designing and developing this mobile-friendly web application may end up being a long, continuing process as new functions are developed and added, new user tests are conducted, and the app is redesigned, retested, and repeated until all currently desired functions are included. The designs and code for future wish-lists can be added in small incremental changes as ideas and code are researched, experimented with, and tested again, and again.

VII. References

Google Developers. (2015). *Mobile-Friendly Test.* Retrieved March 28, 2015 from https://www.google.com/webmasters/tools/mobile-friendly/?url=http%3A%2F%2Fweb.uvic.ca%2F~wanthony%2Fphenology%2FPoP-app.htm

Selby, Ben. (2014). *Website Mobile Friendly? Test it out here!* Retrieved March 28, 2015 from http://www.jbidigital.co.uk/mobile-friendly

Verba, David. (2008). *Sketching in Code: the Magic of Prototyping.* Retrieved March 31, 2015 from http://alistapart.com/article/sketchingincode

Appendix I: User Testing – User Contacts

UserContact	Users	Users#	Position	Date	Place
Dr. Nancy Turner		1	UVic Professor; Ethnobotanist		UVic Environmental Science
Kinga Menu	Uvic Geog 101A Students	20	Senior Lab Instructor		UVic Geography
Dr. Yvonne Coady		1	UVic Instructor; Coding Advocate		UVic Computer Science
Dr. Eric Higgs	ES 482 Novel Ecosystems Seminar	20	UVic Professor; Novel Ecosystems		UVic Environmental Studies
Dr. Brenda Beckwith		1	UVic Instructor; Undergrad Advisor; Ethnobotanist		UVic Environmental Studies
Ken Josephson		1	UVic Technical Services; Designer; Community Mapping		UVic Geography
Dr. Cam Owen		1	UVic Instructor		UVic Geography
Dr. Ian O'Connell			UVic Professor; mapper		UVic Geography
Dr. Bob Bandringa		1	UVic Instructor		UVic Environmental Studies
Robin		1	UVic I.C.E. Innovation Centre For Entrepreneurs		UVic School of Business
David Palmer-Stone		1	UVic Career Counsellor; Mt. Tolmie restoration volunteer		UVic Counselling
Dr. Gerry Allen		1	UVic Herbarium Curator		UVic Biology Herbarium
Carol Cutforth		1	UVic Herbarium Volunteer		UVic Biology Herbarium

Appendix II: Letter to Invite Test-Users to App Presentation

April 1, 2015

Dear Test-User of Wendy Anthony's Phenology of Place at UVic Native Plant Field Guide App:

Thank you for participating in the beta testing of my Phenology of Place Native Plant Field Guide App during the period of January to April 2015. Your interest, usage patterns, and feedback have provided me with invaluable ideas with which to further develop the functions and usability of the App.

As you know, this Field Guide App is a prototype design based on the Native Plant photographs I've taken at the UVic campus between the years 2009 to 2015. After learning more about database management skills and web-coding techniques, I have created a number of different versions of the app based on user requests and functional code, as well as wish lists for new code snippets to learn.

I created a series of Directed Studies projects, starting May 2014, for which I will receive academic credit for two Environmental Studies ES490 courses, with Dr. Eric Higgs, of UVic Environmental Studies, supervising both, along with Dr. Yvonne Coady, of UVic Computer Science, offering supervisary assistance for part 2. In the first part, I created a database from my Native Plant photographs, and developed naming, date and location protocols from which to research seasonal patterns in the growth stages of native plants at the UVic campus. To complete part 2, I created a functional prototype of the app, and a project manual to document the instructions to produce the code for the application design, as well as a report documenting the various versions created, the test user suggestions, changes made, challenges encountered, and code still needed to add the desired functionality.

I will be pleased to present my Phenology of Place project with the resulting database and mobile-friendly web app, on a date still to be determined in April 2015, and would like to invite you with the hope that you may appreciate the changes made as a result of some of the user suggestions received, and may offer feedback and suggestions that might arise from viewing this version of the prototype.

Thank you once again for your encouragement and feedback suggestions, which have all helped to make this challenging project both interesting and useful.

Wendy Anthony