**Documentation for ‘Xertion’ app.**

* All external resources you used / consulted for your project.

All codes and images used for the app are in the zip folder.

Google Maps Javascript API is used to implement Geolocation service for the app.

Google API Key = AIzaSyD6WZaKd3LyB55lHNk3aRyGs7\_iDrqO-i4

A few images were downloaded online. The rest were self-drawn. Below is the reference section for images used in the app.

References

*Fisherman, Fishing Png*. (n.d.). Retrieved from http://www.freeiconspng.com/img/41473

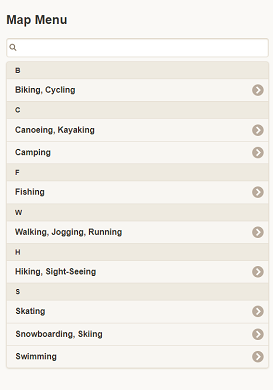
*Kayak on Water transparent PNG*. (n.d.). Retrieved from http://www.stickpng.com/img/sports/kayak/kayak-on-water

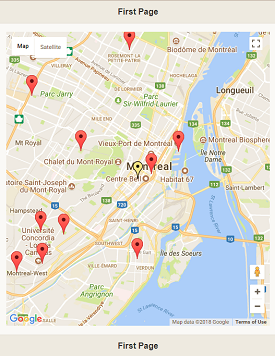
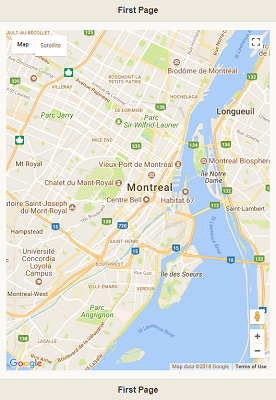
*Person, Biking, Riding, Travel, Bike Icon*. (n.d.). Retrieved from https://iconscout.com/icon/person-biking-riding-travel-bike

*Skiing Transparent Background*. (n.d.). Retrieved from http://www.pngmart.com/image/46911

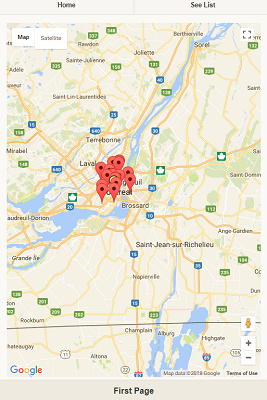
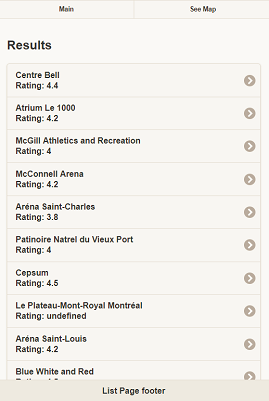
*Trekking Icon*. (n.d.). Retrieved from https://visualpharm.com/free-icons/trekking-595b40b85ba036ed117dacbb

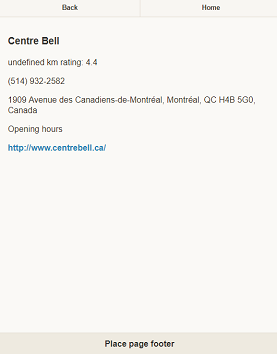
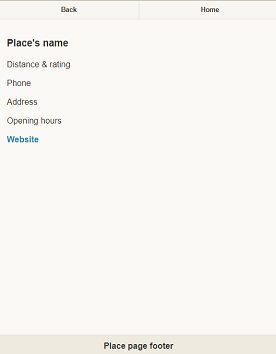
*World Map transparent PNG*. (n.d.). Retrieved from http://www.stickpng.com/img/world-landmarks/global/world-map

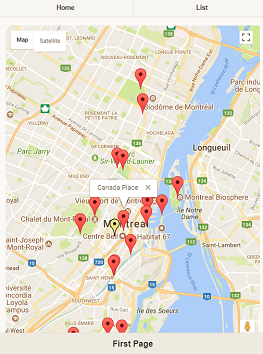
* Documentation of the different stages of your design (e.g. sketches, mockups.)
* First, menu page (‘mainPage’ which list 9 activities) was created and desired page of activity was navigated by clicking on each item on the menu.
* Then, Skating page was developed as a startup for all the activities. Google map was added to the page using Google Maps Javascript API.
* Map was made to fit on the browser of any devices. HTML geolocation was added to display a default location (Montreal, Canada) of the student.
* Using geolocation, user’s current location (latitude and longitude) was calculated. User’s current location was displayed on the map of skating page.
* Skating arenas nearby the user current location was added to the map using Google Places Services and Text Search.



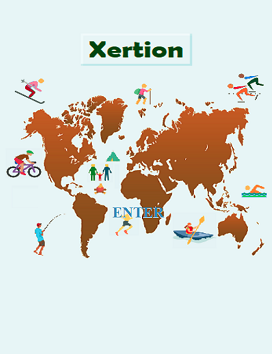
* Places Library was required with a valid API Key.
* A page (‘activityListPage’) to show the list was added. List is shown by filter items. Place search was reimplemented using ‘nearbySearch’ instead so that places are ranked by distance.
* Place’s name and rating were added to be displayed on each list item on the place list.







* Clicking on a list item will open another page which will display detailed informatiom about the place.
* function ‘getPlaceDetails(i) was added. Purpose of ‘getPlaceDetails’ is to do ‘Places Details Search’ to get additional information about each location returned by the‘nearbySearch’. Each result is stored in ‘arenasDetails[]’ array and this array will be used to display additional information of the location on ‘placePage’.
* ‘placePage’ was implemented to add information of each location.
* Distance of a location was fetched from calling ‘getDistanceMatrix’ function and this distance will be displayed to show under place’s name on the map and the ‘placePage’.
* After implementation of the ‘Skating’ activity, the other activities were added. It was ensure that these activities were displayed correctly on the map and listed correctly on the ‘activityListPage’.
* Adding conditions to display the related data appropriately on the ‘placePage’. This page display information about each location.
* How to open url of the location was figured out.
* ‘openNow’ status and operating hours of the locations were fetched from the results and displayed on the ‘placePage’.
* Phonegap and bluestacks are experimented and application was tested on the student’s mobile phone (Galaxy S7).
* On the ‘placePage’, actual URL was hidden and ‘Website’ text was shown instead.
* ‘infowindow’ on Map was upgraded to display name, distance, and open or close status.



* Cover page was created and added in the app. From the cover page, the following ‘mainPage’ was implemented to slide up at transition.
* Color theme was added. (Cover page background: #E2F5FA. Main page background: #045266 (dark greenblue), header background : #def2f7 (very pale blue))
* Main page was updated, and filter feature was removed.
* Thumbnails for each activity were added to enhance to the design.
* OVER\_QUERY\_LIMIT from getDistanceMatrix request was solved. When it occurs, distance is saved as ‘undefined’. In this case, distance will be saved as an empty string ‘’.
* ‘activityPage’ (Map page) was styled and upgraded. Appropriate activity logo was added to the page footer. That way, user can easily remember which activity was chosen.
* ‘activityListPage’ was styled using css.
* Navar icons were added above the text.
* ‘placePage’ was styled using css.
* Due to recurring of ‘OVER\_QUERY\_LIMIT’, a decision was made to give 10 locations. The reason is that the Google API throws a OVER\_QUERY\_LIMIT exception after sending more than 10 geocoding requests in a loop.
* Add photo of the place in placePage if there is a time for that.
* Evaluation results (qualitative, quantitative). There should be a description on when and how evaluation took place, and a presentation of the results. These evaluations should be self-tests and should not involve testing with family/friends/third parties.
* ‘Exertion’ was tested on mobile phone (Galaxy S7) at student’s home first.

|  |  |
| --- | --- |
| **Action** | **Results** |
| App was launched through Phonegap | Cover page displayed. |
| ENTER' was pressed. | main page was navigated. |
| Main page was scrolled up and down to view from top to bottom of the list. | OK |
| Biking Cycling' page was chosen. | A map page opened with user's current location (yellow marker) and 10 searched locations (red markers). |
| a red marker of near by location was touched. | A window (Mount Roya Park, 2.3 km, *closed*) was shown. |
| Other red markers were experimented. | Window for each location was shown with appropiate information. |
| +' | Map view out |
| -' | Map view in |
| square box (top right) | Map expended across the screen. Header and footer bars  were hidden |
| Satellite view was switched | OK |
| Home' was touched. | Main page was navigated. |
| Camping' was chosen | Map opened with 2 locations. |
| Location List' was touched | A page which lists 2 locations opened. |
| First locationed ('Montreal South KOA') was chosen  by touch. | A page which displayed detailed information opened.  Information shown - Name, open, Rating: 4.2, 27.8 km, 29 mins by driving, (450) 659 - 8626, Address: 130 ….., Opening hours: ……, Official website. |
| Official website' was touched. | a window opened for 'Montreal South KOA' webpage. |
| Back' button of the phone was pressed. | Previous page (Montreal South KOA) was navigated. |
| Official website' was touched again. | same information displayed. |
| Close the website window by touching 'x' on top right.  (for iphone) | Back to previous page. |
| Back' button (header bar) was touched. | Back to the list page. |
| Location 2 (Camping Amerique…') was chosen. | Camping Amerique Montreal' page opened with related information displayed on this  page. |
| Home' button (header) was pressed. | Back to main menu page. |
| Canoeing /Kayaking' was chosen. | A map with 3 locations opened. |
| Location 3 ('Depot Du Plein Air') was chosen. | A page with detailed information for 'Trans Canada Trail'  opened. |
| Offical Website' was chosen. | A google search result shown up for this location because  returned URL didn't work. |
| Back' ,'Home', then 'Skating' was chosen. | A map was shown with 10 locations. |
| Location List', and scrolled down to the bottom to 10th location  (Mont Royal Park') | A page for Mont Royal park opened. |
| During the evaluation, phone was rotated several times. | Design displayed OK on both portrait and landscape. |

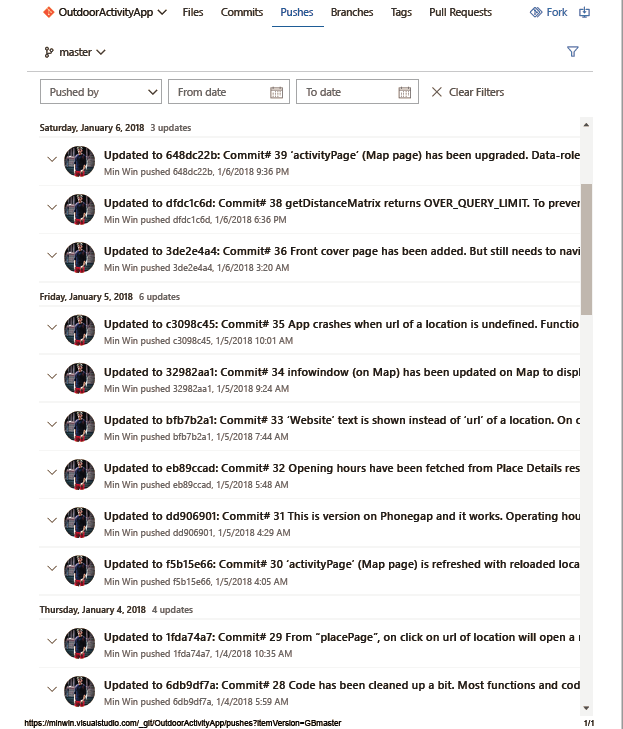
* A summary of your design decisions and how you arrived at them.

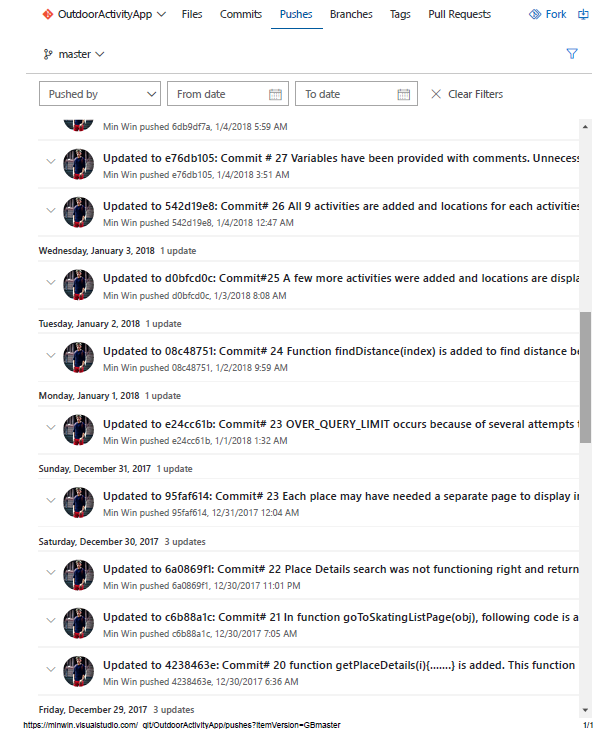
The goal of the final app is quite different from the previous design submitted for the first assignment. The initial goal of the app was to develop a social app for individuals who like to do outdoor activities together or as a group. But, after getting approval from the professor, the goal of the app was changed to become a GPS based app which provides locations of the outdoor activity to user. User should be able to choose a variety of physical outdoor activities from a menu. Activities must include Spring, Summer, Autumn and Winter activities so that this app is usable all year around. ‘JQuery Mobile List Views’ was used to implement the menu page. On click on each menu item, a map page should open and show user with the locations of the activity that user just chose. To get the location results, ‘nearbySearch’ function (Google Places API Web Service) was used over ‘textSearch’ function because the latter doesn’t give some useful information that the nearBySearch returns. From the map page, user should be able to go back to the main menu page and forward to the page which lists all the locations shown on the map. For this reason, home and location list buttons were added on the header bar. For the user to be able to recall what activity was just chosen, an icon of the chosen activity is displayed in the footer bar. On the location list page, locations are numbered so that user can easily find out the number of returned locations. On each location on the list, ‘name, rating and distance’ are displayed. From this page, navigational buttons were added on the header bar so that user can navigate to the home page or the map. ‘jQuery Mobile List Views’ was used again to display the list. A page to display detailed information of a location was added so that user can find out more useful information (duration by driving, business phone number, address, operating hours and official website) of a place. This page is easily navigated from the list. From this page, user can easily navigate to the home page or the list page. ‘Xertion’ logo was drawn in green to represent nature and freshness. A light blue theme is used to represent calmness and loyalty.

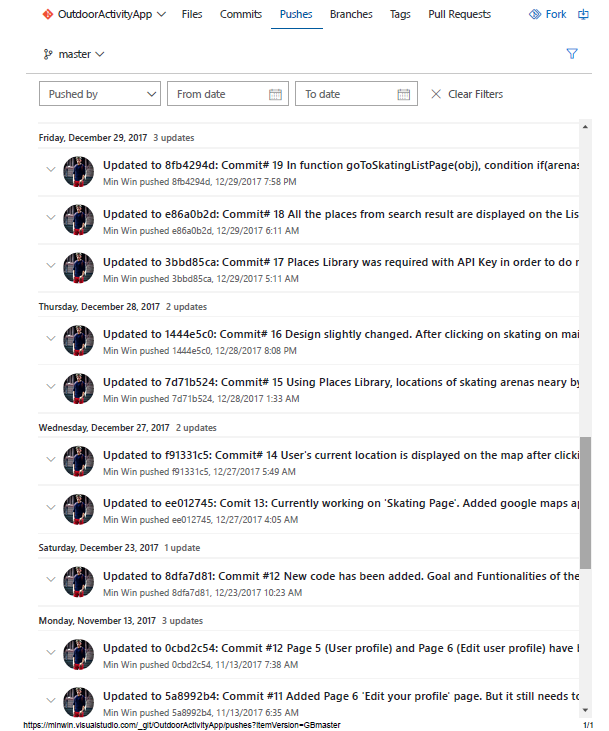
* Any implementation difficulties you encountered and how you addressed them.
* To display the map correctly on the page was troublesome. ‘getRealContentHeight’ function was implemented to solve this issue.
* Global variable inside a function didn’t get updated correctly. Using ‘var’ keyword solved this problem.
* ‘Place Details Search’ did not work correctly inside a ‘for’ loop. A separate function for this search is implemented and this function is called with an argument (i) within ‘goToActivityListPage’ function.
* There was code redundancy to update information of each location on the ‘activityListPage’ and ‘placePage’. ‘For’ loop is used to update each location with the ids “#place\_0” to “#place\_19”. Javascript substr() is used to solve this problem.
* Separate pages may have been required to display information. To solve this problem, In ‘goToPlacePage’ function, place number is added and this number is matched with index variable ‘i’ inside the ‘for’ loop so that when place number and ‘i’ match, place information is retrieved from ‘arenasDetails’ array. Then, only one page was needed for all 10 places.
* Paragraph to display a place’s information in ‘placePage’ was right next to the border. Margin was added to the ‘placePageContent’ to solve this problem.
* OVER\_QUERY\_LIMIT occurs due to several attempts to access ‘Place Details search’.

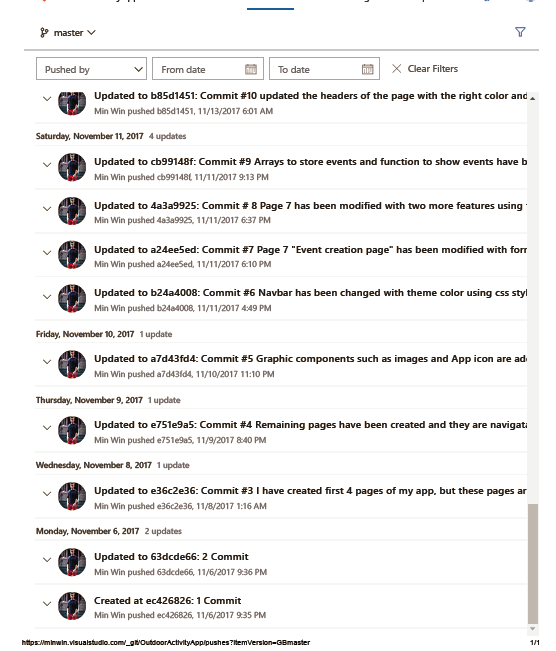
A pool of 9 or 10 requests exhaust the function, and an error is shown after 10 requests. From 10th request, execution is delayed 1 second to fetch the next request.

* There were code repetitions for activity pages. Each activity is navigated to one page only, ‘activityPage’.
* When there were less than 10 locations, extra navigational buttons on ‘activityListPage’ must be hidden. Those hidden buttons must be displayed again when there are 10 locations. ‘Else’ condition is added to solve this issue.
* App crashed when URL was launched because URL of a location is undefined.
* A couple thumbnails were not displayed. Student’s phone had to reset.
* getDistanceMatrix returns OVER\_QUERY\_LIMIT. To prevent this, (function (i) { setTimeout(function() {} was added. For 1st – 10th requests, ‘getDistanceMatrix’ is invoked instantly, from 10th – 20th, execution delays one second for each ‘getDistanceMatrix’ request.
* A printout of the commits from your visualstudio.com repository
* See the following pages.









* The url of your visualstudio.com repository
* https://minwin.visualstudio.com/
* https://minwin.visualstudio.com/OutdoorActivityApp