

CSCI 4131 – Fall 2021
Internet Programming Assignment 3
Due Date: *Friday Oct 8th at 11:59pm*
Late Submission Deadline: Sunday, Oct 10th at 5:59am

1 Description

For this assignment, you will add Google Maps to the functionality you have developed on your Schedule Web pages through homework 2, enhance the styling of your Widgets Page, (CSS / HTML), and eliminate errors. There are 14 pages in this assignment description.

The objective of this assignment is to continue to develop your JavaScript skills, introduce you to the Google Maps JavaScript application programming interface (API), the Google Places JavaScript library, the Google Maps Directions Service, and geolocation. While the Google Maps API will be introduced in subsequent lectures, developing and/or bolstering the ability to read library documentation and then use it to develop functionality is an essential skill for today's web developer (or any software developer). New libraries with new APIs are introduced, and existing libraries and their APIs are updated, on a regular basis. Teaching a specific API is counterproductive in such an environment. Instead, one of the objectives of this assignment is to motivate you to develop and/or bolster your skills to learn and use new libraries and APIs.

You will update your Schedule page as follows:

1. You will write code to dynamically mark the addresses of the places specified in your schedule web page on an embedded Google map.
2. You will develop additional functionality to search for specific places near your current location and display them on the map.
3. You will also add functionality to calculate directions and display a route between your current location and destination on your Google map via various modes of transportation.
4. You will add a timer to the table that displays the current time, and update it every second.
5. Upon opening the Schedule webpage, mark those physical events which occur on the current day of the week with some special color.
6. And, finally, you will replace the large picture displayed on your Schedule page with a widget that “fades-in” or “fades-out” the large figure associated with the current event on the Schedule

You will update your form page as follows:

1. You will add a Google map on which you can select a location on the Google map and use the click on point of interest capability to fill in the address field on your form.

Finally, if you have not done so already, you should restyle the widgets on your widgets page, so they are better positioned and error-free (so you will have to fix any errors due to the embedded widgets you obtained from YouTube, Twitter, and Instagram).

2 Preparation/Reference

Sign up for the Google Maps JavaScript API – see:

<https://developers.google.com/maps/documentation/javascript/get-api-key>

Note, you need a credit or debit card to sign up for the Google Maps Application Programmer Interface. The uses required by this course will never necessitate a charge for use of Google Maps – it is only required should you deploy your Maps to a production website. Contact the instructor if you have questions or concerns about this.

2.1 Google API Setup

Some setup is required to use the Maps API – you need to get your API key (see above). Google provides an excellent tutorial for obtaining your API key and setting up your map, and it is recommended to complete the tutorial. The tutorial can be found at the following link:

<https://developers.google.com/maps/documentation/javascript/tutorial>

When signing up for your API key, use your UMN x500 account

2.2 Google Places JavaScript Library Reference

You can refer the following link to obtain more information about places library:

<https://developers.google.com/maps/documentation/javascript/places>

2.3 Google Maps Directions Service Reference

You can refer the following link to obtain more information about direction service:

<https://developers.google.com/maps/documentation/javascript/directions>

2.4 Geolocation Reference

You can refer the following link to obtain more information about geolocation:

<https://developers.google.com/maps/documentation/javascript/geolocation>

2.6 Google Click on Points of Interest (used to fill address field on Form when points of interest are selected/clicked on the map next to it):

<https://developers.google.com/maps/documentation/javascript/examples/event-poi>

3 Functionality

Update functionality on your Schedule page

1. Update the functionality that displays a large picture of the last icon that a user hovers over with their selection device to a fading show “widget” that you develop using JavaScript (**DO NOT USE jQuery to implement this functionality**). The fading show “widget” you develop should be placed next to the Schedule table as illustrated in figures 4.1, 4.2 and 4.3 below. This widget should also have a special button, that displays either: “Go away!” or “Come back!” as follows. When the button displays “Go away” and is clicked, the large picture “fades out” over a period of 2 seconds or more. When the button displays: “Come back!” and is clicked, the large picture “fades-in” over a period of 2 seconds or more.
2. Modify the address column of the table from homework 2 to include the complete address information in table entries for those events, e.g., for Shepherd Labs - 100 Union St. SE, Minneapolis. MN 55455. If you have less than 5 physical events (with an actual address), add some physical events to the table.
3. Embed a Google map under the table.
4. The map should be centered on University coordinates (44.9727, -93.23540000000003) with zoom level of 14 (or any zoom level that you find appropriate). University coordinates can also be obtained using the following link: <https://www.gps-coordinates.net/>
5. You should write JavaScript code to **dynamically** extract the name, time and location of the physical events from the DOM objects that are in your Schedule table. Your code should then place a custom marker on the map for each location extracted. The markers should display the name and time of the event, and also display the complete address of this event (e.g. Room number 2-209 in Keller or Shepherd 383) upon being clicked. **Note, do not use hard-coded latitude and longitude positions to do this.** More specifically, write JavaScript that obtains a list of addresses from your physical events and use the geocoding or places library to obtain a latitude and longitude for each of them. Then create a marker for each unique latitude and longitude. This can be achieved using an information window (see figures 4.4 and 4.5 below).
6. The next step is to insert an input area on the right of the map. The elements in this input area will require you to use ‘Google Places JavaScript library’ and ‘Google Maps Directions Service’. See figure 4.6 below.
7. The first row of input area (created as per step 6 above) should enable a user to for search places, e.g., restaurant near the user’s current geolocation. It should consist of:
 - A drop down field that lists the category of places to search for. The default categories for this homework are restaurant, hospital, parking, supermarket. You can replace these categories with additional categories present at:

https://developers.google.com/places/supported_types. You should ensure that at least a few places exist in the search results for each of the categories you include.

- A text box that specifies the numeric radius around your current location in which the search results will preferably be shown.
 - A textbox that allows users to enter the keywords of other places that they may search for. This textbox is by default disabled. The drop down field should include an extra option named: “Other”. When “Other” is selected, the textbox is enabled. Try one of the following keywords (or another keyword of your choice) when you are testing this - burger, bus, library, laundromat etc.
 - A button labelled ‘Search’: Upon clicking this button, your code finds all the nearby places within the specified radius using ‘Google Places JavaScript library’ and displays the results by placing a marker on the map for each of the found places. The old markers (e.g. the markers for your events location, and markers for any other category) should be cleared before placing new markers on the map. When clicking on a marker, the name and address of the location should be displayed.
 - Refer to figures 4.6 and 4.7 for an illustration of this functionality.
8. The second search component that should be added is the functionality to get directions from a user’s current location (as determined by HTML Geolocation – see https://www.w3schools.com/html/html5_geolocation.asp) to the destination they provide. The group of elements you should use to implement this functionality are as follows:
- A textbox that allows the user to input the destination location. The source location will be the current location of the user and it can be obtained using geolocation.
 - A radio button group: Three modes of travel should be specified: DRIVING, WALKING, and TRANSIT. One of these must be selected at all times.
 - A button labelled ‘Go’: Upon clicking this button, the route between the current location and the destination should be displayed on the map. The displayed route will be based on the mode of travel selected by the user. The directions associated with this route should be displayed on a scrollable side panel floating to the left of the map. Directions involving higher number of instructions should be wrapped into this fixed dimension panel with the scrollable feature. The source for the directions is the user’s current location extracted using Google geolocation services. The source coordinates should not be hard-coded, they should be extracted dynamically using code (JavaScript). Make use of ‘Google Maps Directions Service’ for this functionality.
 - Refer to figures 4.8 and 4.9 for an illustration of this functionality

Update Functionality on your Forms page

- a.) Add a Google Map to the right of your form
- b.) ***To earn up to 10 bonus points***, use Google Map’s autocomplete functionality to fill in the address field on the form from locations selected from the Google map to the right of your form AND/ OR autofill the address field with a click on a point of interest on the map. See Figures 4.10 and 4.11 below for an example of click on POI functionality. See the code snippet on Page 9 to help you incorporate autofill functionality

Update the Style on your widgets page and get rid of the errors caused by the embedded widgets obtained from YouTube, Instagram, and Twitter.

See figure 4.12 below for an improvement on styling your widgets page (if you have not already done so).

4 Screenshots

4.1 Add slide show “widget” to the right of the Schedule table, and a clock above the table, change the color of the PHYSICAL events that happens today (Wednesday in this screenshot).

12:23:38

My Weekly Schedule

Day	Event	Time	Event Location (Virtual or Physical)	Phone Number	Extra Info (URL)
Monday	Csci 4131 Lecture	9:45am - 11:00am	5 Blegen Hall	See Class information on Canvas	CSci 4131 Info
Monday	Dr C's In-Person Office Hours	3:00pm - 4:00pm	383 Shepherd Labs	See Meeting info on Class Google calendar	Calendar Link
Tuesday	Csci 4011 Lecture	2:30pm - 3:45pm	3-210 Keller Hall	See Class information on Canvas	CSci 4011 Canvas
Wednesday	Csci 4131 Lecture	9:45am - 11:00am	5 Blegen Hall	See Class information on Canvas	CSci 4131 Info
Wednesday	Dr C's Virtual Office Hours	7:15pm - 8:15pm	Virtual (Zoom)	See Zoom Meeting info on Class Google Calendar	Calendar Link
Thursday	Walk Dog	7:30 - 9:00am	901 15th Ave SE Minneapolis, MN 55414	(612) 370-4926	Dog Walk Info
Thursday	Csci 4011 Lecture	2:30pm - 3:45pm	3-210 Keller Hall	See Class information on Canvas	CSci 4011 Canvas
Saturday	Breakfast	10:00am - 11:00am	413 14th Ave SE Minneapolis, MN 55414	(612) 331-9991	Breakfast Spot
Sunday	Jog	10:00am - 11:00am	1801 6th St SE, Minneapolis MN 55455	None	Info on where I'm Jogging

Go away!

Map Satellite

Find Restaurant

Enter Other Places

in 500 meters

Search

4.2 The fading show widget should display the **unique** images drawn from the thumbnail pictures of your Schedule table. The default button should be “Go away!”. When the “Go Away!” button (feel free to give it another name) is clicked, the current picture should start to fade out, and completely fade out over a period of 2 or more seconds - , and the button should be changed to “Come back!”. Once the “Come back!” button is clicked, the picture associated with the current event should re-appear over a period of 2 or more seconds, and the button should change to “Go away!” again.


Clicked “Go away!”

12:28:25

My Weekly Schedule

Day	Event	Time	Event Location (Virtual or Physical)	Phone Number	Extra Info (URL)
Monday	Csci 4131 Lecture	9:45am - 11:00am	5 Blegen Hall	See Class information on Canvas	CSci 4131 Info
Monday	Dr C's In-Person Office Hours	3:00pm - 4:00pm	383 Shepherd Labs	See Meeting info on Class Google calendar	Calendar Link
Tuesday	Csci 4011 Lecture	2:30pm - 3:45pm	3-210 Keller Hall	See Class information on Canvas	CSci 4011 Canvas
Wednesday	Csci 4131 Lecture	9:45am - 11:00am	5 Blegen Hall	See Class information on Canvas	CSci 4131 Info
Wednesday	Dr C's Virtual Office Hours	7:15pm - 8:15pm	Virtual (Zoom)	See Zoom Meeting info on Class Google Calendar	Calendar Link
Thursday	Walk Dog	7:30 - 9:00am	901 15th Ave SE Minneapolis, MN 55414	(612) 370-4926	Dog Walk Info
Thursday	Csci 4011 Lecture	2:30pm - 3:45pm	3-210 Keller Hall	See Class information on Canvas	CSci 4011 Canvas
Saturday	Breakfast	10:00am - 11:00am	413 14th Ave SE Minneapolis, MN 55414	(612) 331-9991	Breakfast Spot
Sunday	Jog	10:00am - 11:00am	1801 6th St SE, Minneapolis MN 55455	None	Info on where I'm Jogging

Come Back!



The figure gradually fades out (over a period of 2 seconds or more).

12:26:48

My Weekly Schedule

Day	Event	Time	Event Location (Virtual or Physical)	Phone Number	Extra Info (URL)
Monday	Csci 4131 Lecture	9:45am - 11:00am	5 Blegen Hall	See Class information on Canvas	CSci 4131 Info
Monday	Dr C's In-Person Office Hours	3:00pm - 4:00pm	383 Shepherd Labs	See Meeting info on Class Google calendar	Calendar Link
Tuesday	Csci 4011 Lecture	2:30pm - 3:45pm	3-210 Keller Hall	See Class information on Canvas	CSci 4011 Canvas
Wednesday	Csci 4131 Lecture	9:45am - 11:00am	5 Blegen Hall	See Class information on Canvas	CSci 4131 Info
Wednesday	Dr C's Virtual Office Hours	7:15pm - 8:15pm	Virtual (Zoom)	See Zoom Meeting info on Class Google Calendar	Calendar Link
Thursday	Walk Dog	7:30 - 9:00am	901 15th Ave SE Minneapolis, MN 55414	(612) 370-4926	Dog Walk Info
Thursday	Csci 4011 Lecture	2:30pm - 3:45pm	3-210 Keller Hall	See Class information on Canvas	CSci 4011 Canvas
Saturday	Breakfast	10:00am - 11:00am	413 14th Ave SE Minneapolis, MN 55414	(612) 331-9991	Breakfast Spot
Sunday	Jog	10:00am - 11:00am	1801 6th St SE, Minneapolis MN 55455	None	Info on where I'm Jogging

Come Back!


Clicked “Come back!” The picture of the current event should fade back in over a period of 2 seconds or more.

12:27:2

My Weekly Schedule

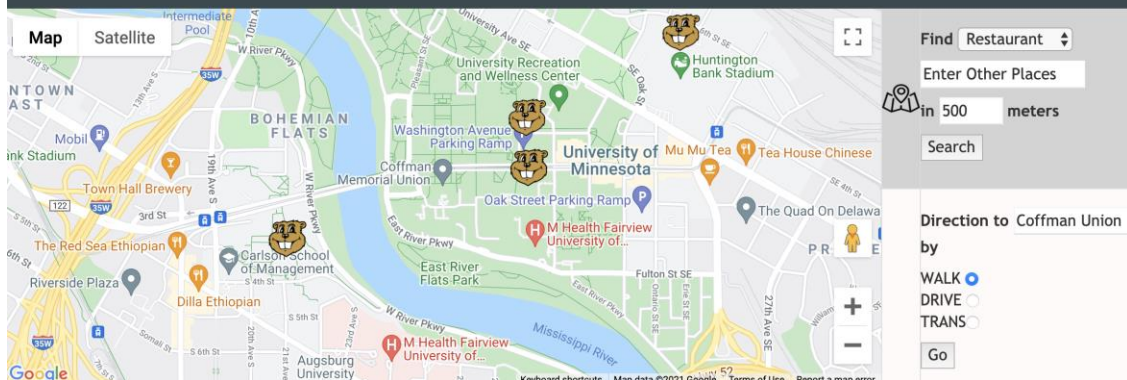
Day	Event	Time	Event Location (Virtual or Physical)	Phone Number	Extra Info (URL)
Monday	Csci 4131 Lecture	9:45am - 11:00am	5 Blegen Hall	See Class information on Canvas	CSci 4131 Info
Monday	Dr C's In-Person Office Hours	3:00pm - 4:00pm	383 Shepherd Labs	See Meeting info on Class Google calendar	Calendar Link
Tuesday	Csci 4011 Lecture	2:30pm - 3:45pm	3-210 Keller Hall	See Class information on Canvas	CSci 4011 Canvas
Wednesday	Csci 4131 Lecture	9:45am - 11:00am	5 Blegen Hall	See Class information on Canvas	CSci 4131 Info
Wednesday	Dr C's Virtual Office Hours	7:15pm - 8:15pm	Virtual (Zoom)	See Zoom Meeting info on Class Google Calendar	Calendar Link
Thursday	Walk Dog	7:30 - 9:00am	901 15th Ave SE Minneapolis, MN 55414	(612) 370-4926	Dog Walk Info
Thursday	Csci 4011 Lecture	2:30pm - 3:45pm	3-210 Keller Hall	See Class information on Canvas	CSci 4011 Canvas
Saturday	Breakfast	10:00am - 11:00am	413 14th Ave SE Minneapolis, MN 55414	(612) 331-9991	Breakfast Spot
Sunday	Jog	10:00am - 11:00am	1801 6th St SE, Minneapolis MN 55455	None	Info on where I'm Jogging

Go Away!

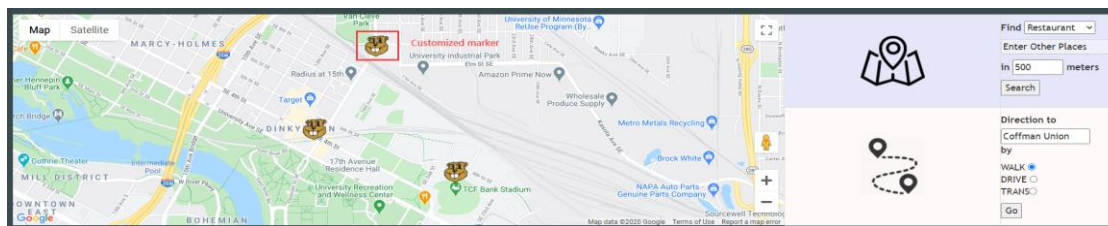


4.3 Table with a list of events, google map and input area floating to the right of the map

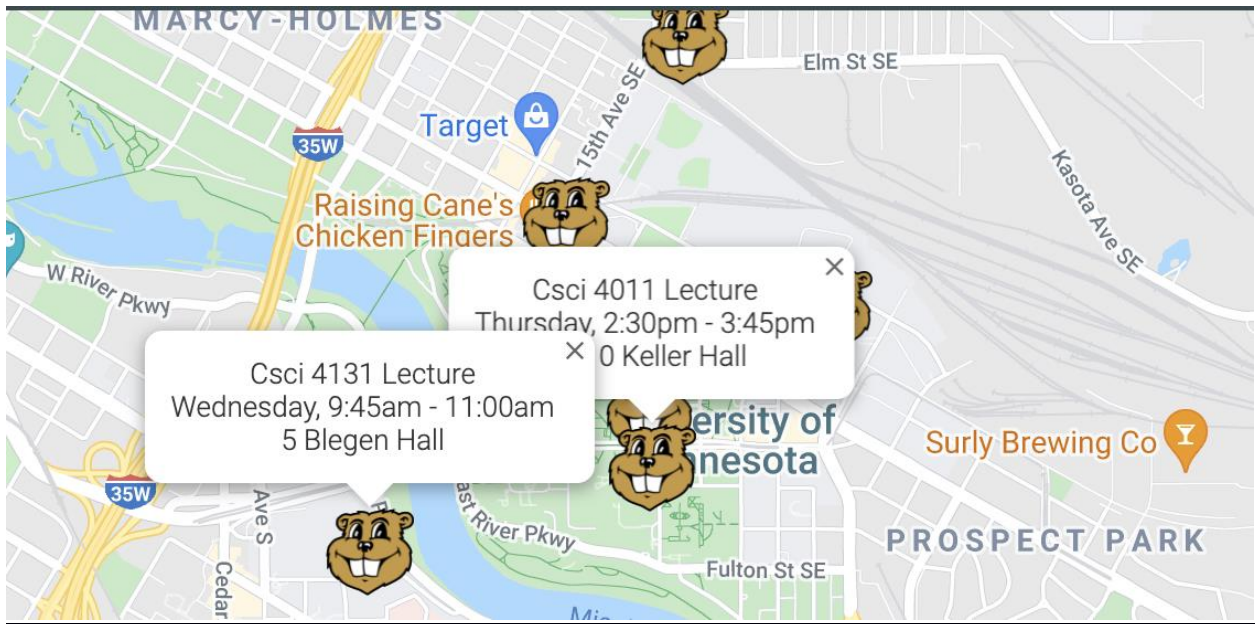
Monday	Dr C's In-Person Office Hours	3:00pm - 4:00pm	383 Shepherd Labs	See Meeting info on Class Google calendar	Calendar Link
Tuesday	Csci 4011 Lecture	2:30pm - 3:45pm	3-210 Keller Hall	See Class information on Canvas	CSci 4011 Canvas
Wednesday	Csci 4131 Lecture	9:45am - 11:00am	5 Blegen Hall	See Class information on Canvas	CSci 4131 Info
Wednesday	Dr C's Virtual Office Hours	7:15pm - 8:15pm	Virtual (Zoom)	See Zoom Meeting info on Class Google Calendar	Calendar Link
Thursday	Walk Dog	7:30 - 9:00am	901 15th Ave SE Minneapolis, MN 55414	(612) 370-4926	Dog Walk Info
Thursday	Csci 4011 Lecture	2:30pm - 3:45pm	3-210 Keller Hall	See Class information on Canvas	CSci 4011 Canvas
Saturday	Breakfast	10:00am - 11:00am	413 14th Ave SE Minneapolis, MN 55414	(612) 331-9991	Breakfast Spot
Sunday	Jog	10:00am - 11:00am	1801 6th St SE, Minneapolis MN 55455	None	Info on where I'm Jogging



4.4 Markers of the physical events in the schedule on the google map.



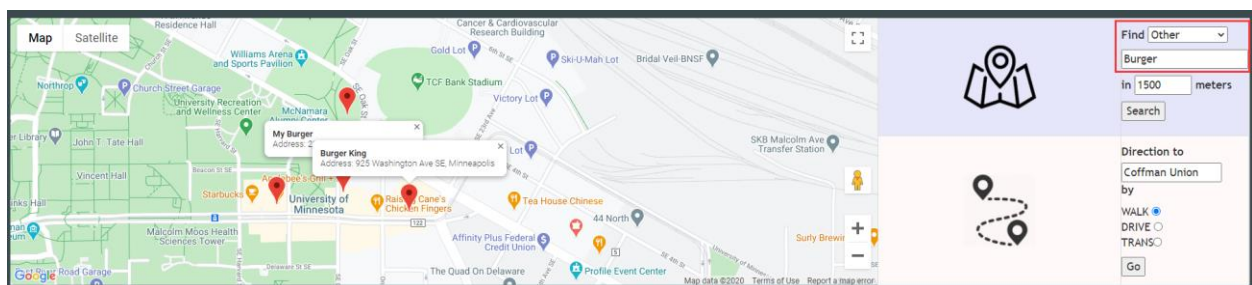
4.5 Marker shows the name, time and address of a physical event on click (you should use JavaScript to extract that information from your schedule table, in the screenshot, two markers have been clicked)



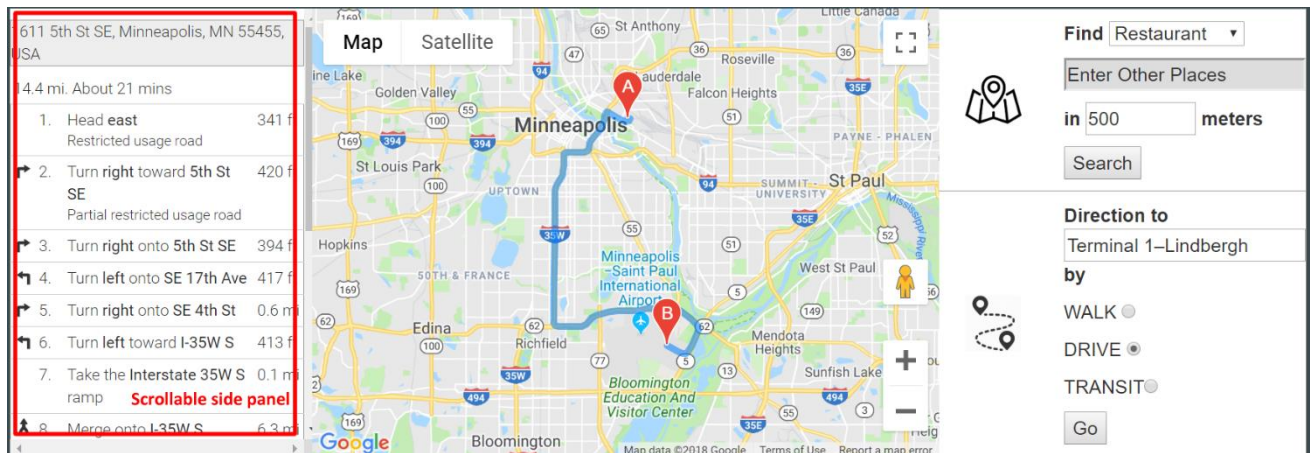
4.6 Map showing the result of nearby restaurants in a radius of 1500 meters



4.7 When “Other” is selected in the dropdown list, user can enter the keyword to search for items related to places they are looking for, e.g., “burger”. The nearby places with the keyword in their names will show up when “search” button is clicked (See the results for “burger” below)

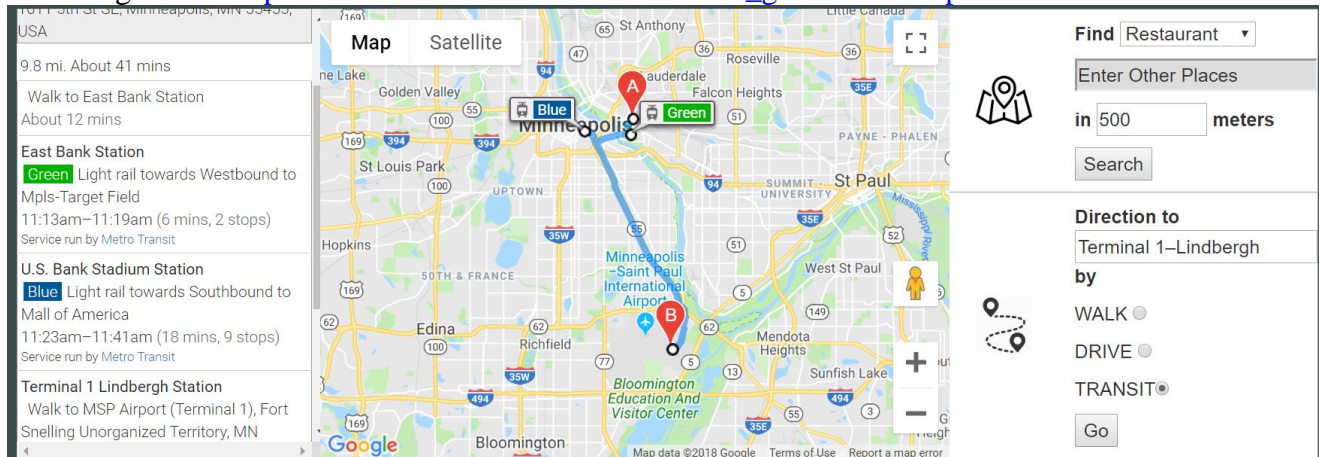


4.8 Map showing the DRIVING route between the current location and Terminal 1–Lindbergh



4.9 Map showing the TRANSIT route between the current location and Terminal 1–Lindbergh

Note: you can use HTML5 Geolocation to set your current location – see the information at the following link: https://www.w3schools.com/html/html5_geolocation.asp



4.10 Add a Google Map to your Form Page

My Contacts Form Input My Widgets

Name	<input type="text"/>
Category	Personal
Location	<input type="text"/>
Contact Information	<input type="text"/>
Email	<input type="text"/>
Website	<input type="text"/>
<input type="button" value="Submit"/>	

地図 航空写真

Washington Avenue Parking Ramp
501 SE Washington Ave
Minneapolis, MN 55455
Googleマップで見る

4.11 When a place is selected on the Map, the “Location” field on your form should be populated automatically with the address of the place you selected using Google’s click on POI (Points of Interest) functionality.

(501 SE Washington Ave Minneapolis, MN, 55455 for example shown in the screenshot)

My Contacts Form Input My Widgets

Name	<input type="text"/>
Category	Personal
Location	501 SE Washington A
Contact Information	<input type="text"/>
Email	<input type="text"/>
Website	<input type="text"/>
<input type="button" value="Submit"/>	

地図 航空写真

Washington Avenue Parking Ramp
501 SE Washington Ave
Minneapolis, MN 55455
Googleマップで見る

The “snippet” of HTML and JavaScript included at the top of the next page can be used to help incorporate Google Maps Auto-complete functionality into the address text field on your form.

```

<input id="addressTextField" type="text" size="50">
  <script>
    function init() {
      var input =      I
      document.getElementById('addressTextField');
      var autocomplete =
        new google.maps.places.Autocomplete(input);
    }
  </script>

```

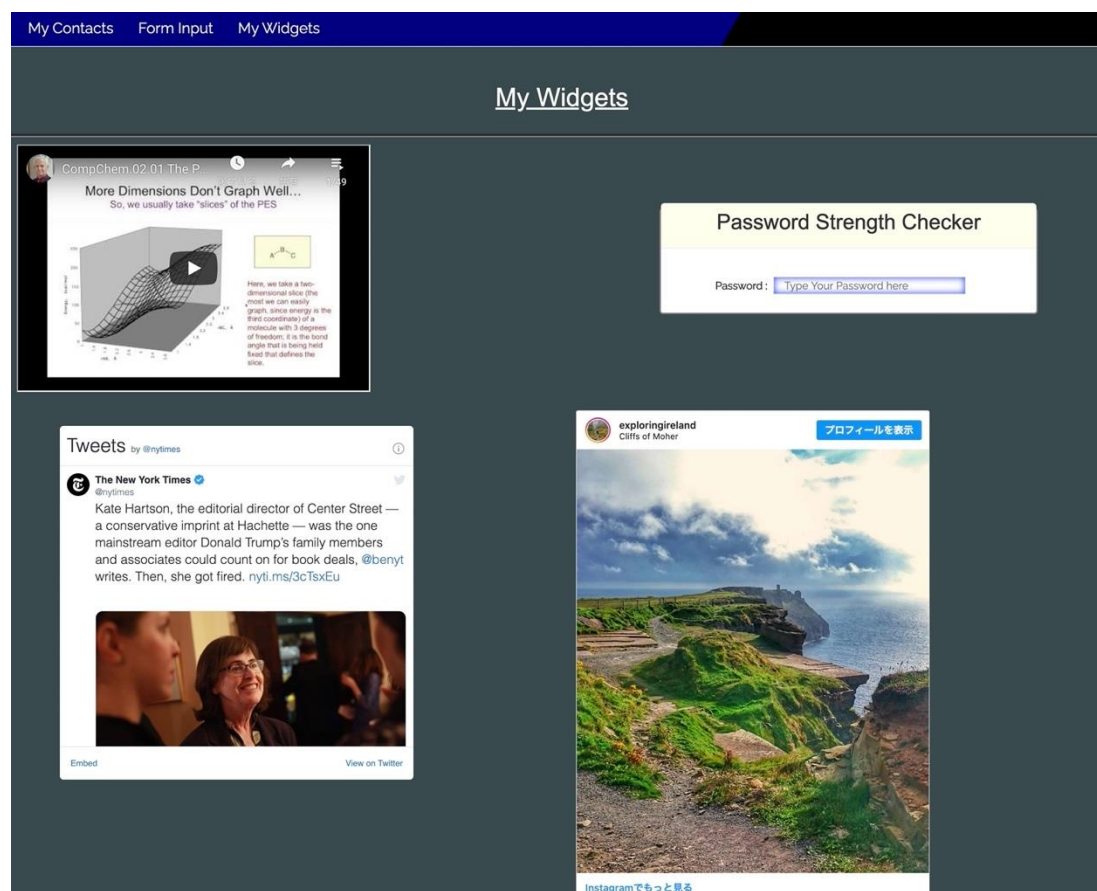
For “click on POI functionality”, you will need to add the JavaScript API (with your key) to your forms page to include another Google map on the page. See the example at the link specified in section 2.6 on page two of this document:

```

<script type="text/javascript"
src="https://maps.googleapis.com/maps/api/js?key=YOUR-GOOGLE-
MAPS-KEY=places&callback=initMap"></script>

```

4.12 Improve the styling on your widgets page, and get rid of HTML and CSS errors caused by the Instagram, YouTube, and Twitter widgets (check to ensure you have done that with the HTML and CSS validators).



5 Submission Instructions

1. For this assignment, you are expected to separate the HTML, CSS, and JavaScript files.
2. Your submission should include a **minimum** of 5 files (3 HTML, and 1 each for JavaScript and CSS). You can include more than one JavaScript and/or CSS file. However, it is required that the CSS is separated from HTML (that is, in an external file), and you should include the majority of your JavaScript in an external file or files as well.
3. Zip these files and the name of the zipped folder should be **yourx500id_hwk03**.
PLEASE ENSURE TO TEST YOUR CODE Using Google Chrome and Firefox.

6 Evaluation

Your submission will be graded out of 110 points (100 points and 10 bonus points) on the following items:

1. ALL HTML files pass the w3schools validator (<http://validator.w3.org>) without errors. **Warnings are accepted (5 points).**
2. All CSS files pass w3schools validator (<http://jigsaw.w3.org/css-validator/>) without errors. **Warnings are accepted (5 points).**
3. A digital clock is shown on the schedule page and updates each second. The physical events that happen on the same day as the clock time should be marked with a different color. **(5 points)**
4. “Fading picture” widget is placed on the Schedule page next to the Schedule table with proper alignment as shown in the pictures, and functions as specified in this write-up.
 - a. All functionality from Assignment 2 works correctly. **(2 points)**
 - b. “Go away” button fades the current large picture out over a period of 2 or more seconds when it is clicked. And the button changes into “Come back”. **(4 points)**
 - c. “Come back” button fades the currently hidden large picture back in over a period of 2 or more seconds when it is clicked, and the button changes into “Go away”. **(4 points)**
5. Google Map is placed on webpage below the Schedule table with proper alignment as shown in pictures **(10 points).**
6. Custom markers are dynamically placed on the physical locations specified in your Schedule table. You must dynamically obtain the event names, addresses, and time information from your Schedule table and use the Google Maps Geocoding or Places functionality to create markers and place them on the map. **(15 points).**
7. When the markers created in step 5 above a selected (via a click), the markers display the event name, address, and time information of the event. **(5 points).**
8. Nearby places can be searched within a specific radius and corresponding markers are created on map **(5 points).**
9. When clicking your mouse on each of the markers placed on map in response to a place search, each marker displays the name of the nearby place **(5 points)**.
10. All previously displayed markers are removed before the results of a new search are displayed **(5 points).**
11. Functionality for finding directions to a custom destination from the user’s current location functions as specified in this write-up.
 - i. Accept destination through input text box and correctly display directions on side

- panel (**10 points**).
 - ii. Functioning capability to switch between multiple travel modes (**2 points**).
 - iii. Style and alignment of the side panel with scroll feature as shown in picture (**3 points**).
12. The form page is modified to include a Google Map. (**10 points**)
- 13. On the forms page, when a location on the Google Map is selected, it will automatically populate the address field of the form. (5 bonus points)**
- 14. Enable autocomplete functionality on the address field of the form (5 bonus points)**
15. The Webpage should be responsive. (5 points)
16. All the files required for your solution should be packaged in a tar or zip file, and submitted via the link on the Moodle class site. The name of the zipped file should be **yourx500id_hwk03.zip**. Failure to do this correctly may result in a deduction ranging from 10 points up to full credit for the assignment.