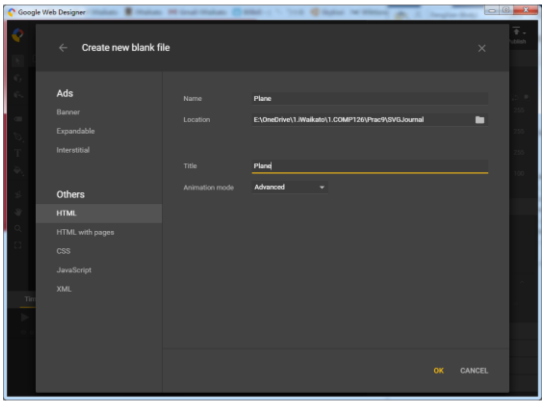
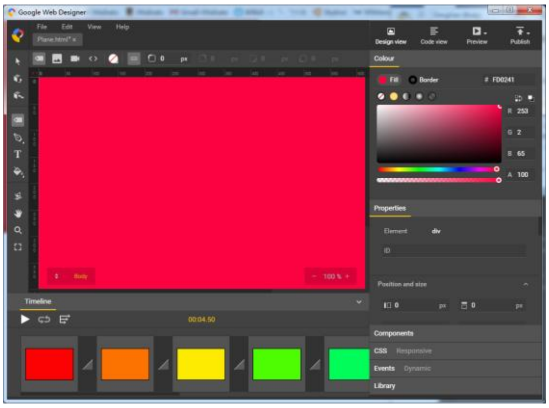
COMP126-16B
SVG Journal with Google Web Design
Yunhao Fu
1255469

Firstly, I constructed a folder to store all the materials that I will use and the website pages generated.
My topic of animation is the Journey. In the first page, I have a human chasing an airplane, kind of interesting maybe he or she missed the flight. Then it comes to my second page, where a Google map appears. In this page, the black circle is targeting where this trip to go, and the green circle is rotating to scan where we could visit. Thanks to the Google map controller, I can use Google street view to have this electronic trip.

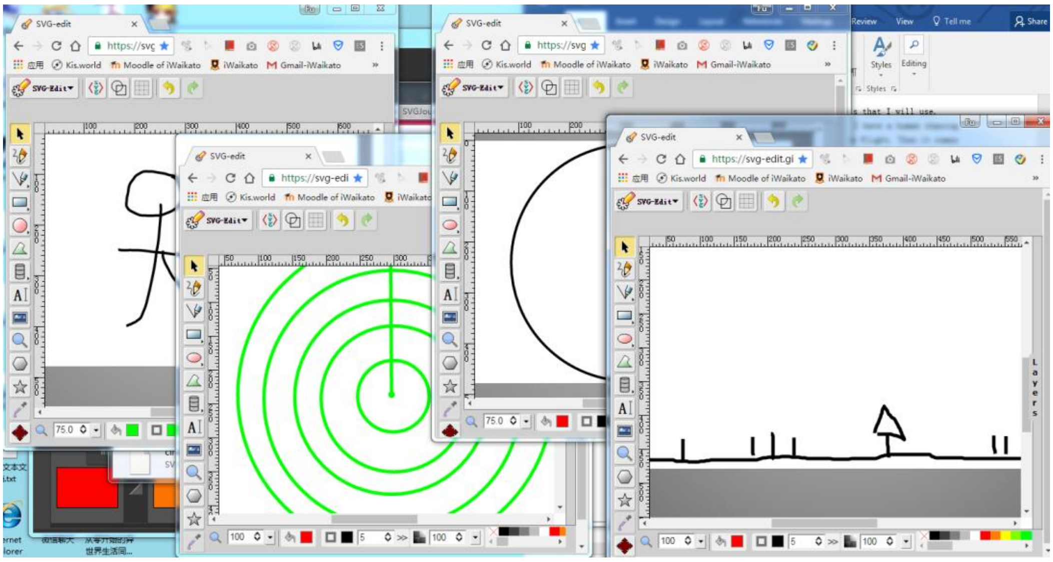
All the files made by Google Web Designer, at beginning, I create a blank file to go.



Name: Plane
Location: E:\OneDrive\1.iWaikato\1.COMP126\Prac9\SVGJournal
Title: Plane
Animation mode: Advanced



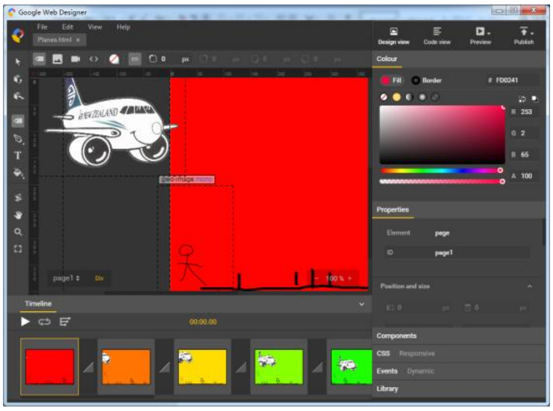
Then, I create serval frames of different color background, because the animation is linear, it is quite good to transit next keyframe.

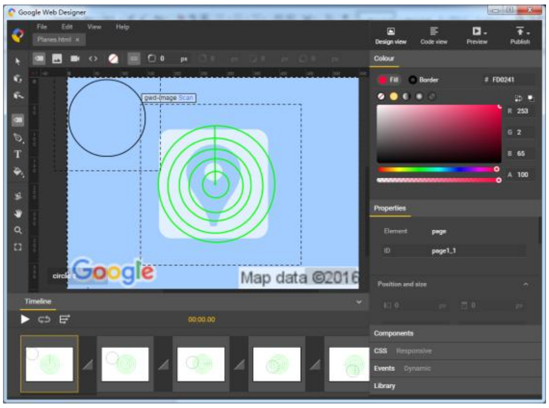


The SVG picture I draw at the SVG-edit online. By collecting the code from the website, I create new txt file and paste code into it and change the name to

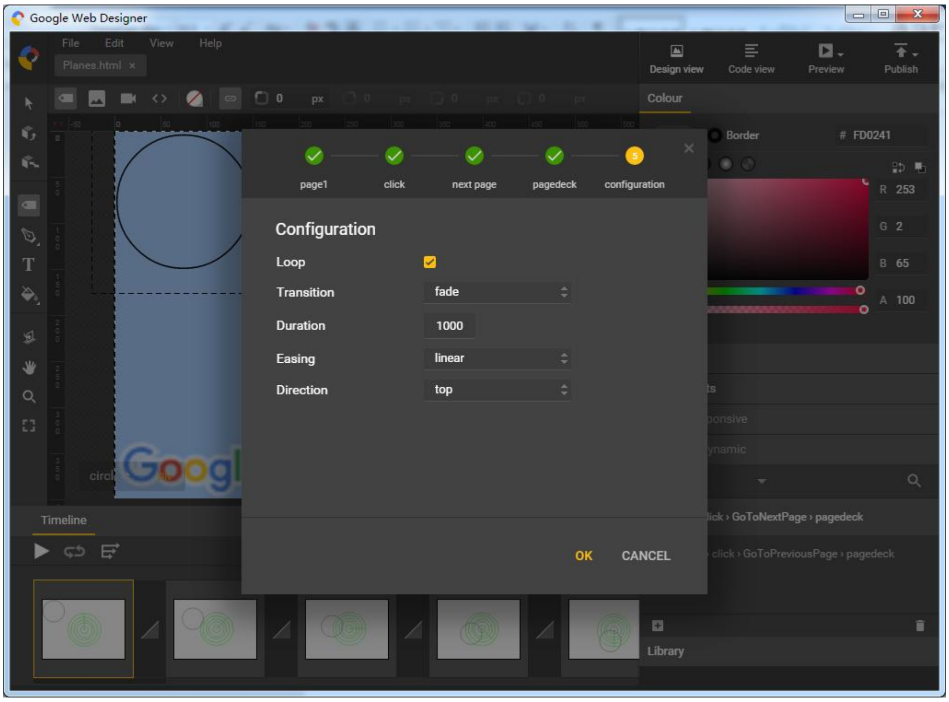
*.svg. However, the airplane picture is retrieved from the Internet. The URL is http://www.gayepardy.co.nz/.cm4all/iproc.php/Airnz%20twin.JPG/downsize_1280_0/Airnz%20twin.JPG

Put everything I need into different div framework. Adjust they to different position to form the animation idea and input their parameters to make it work.





The trigger event to another page is the mouse click, where I set it in the Events box at the right side.



page1 › click › next page › pagedeck › configuration

Configuration

Loop ☑

Transition fade

Duration 1000

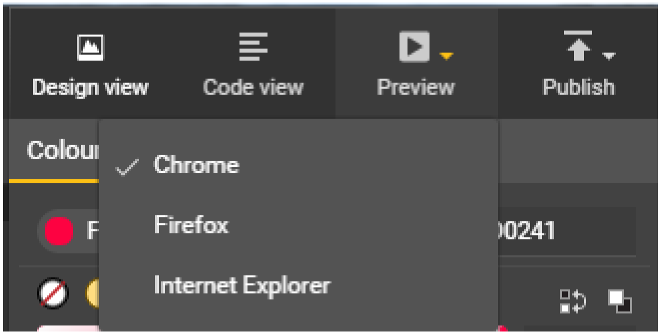
Easing linear

Direction top

click › GoToNextPage › pagedeck

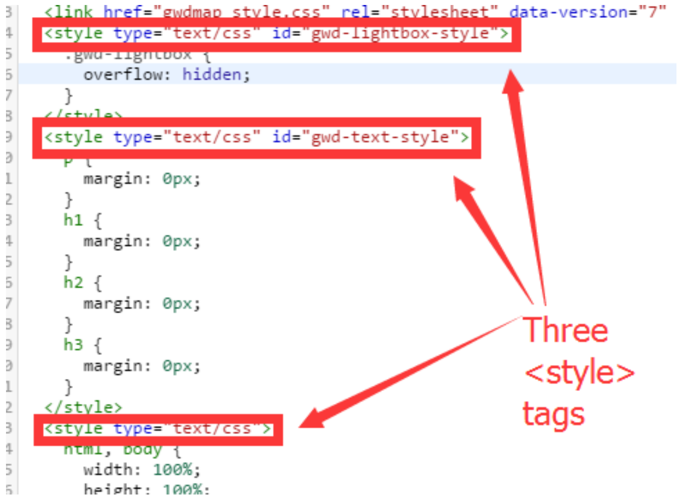
click › GoToPreviousPage › pagedeck

Thanks to the functionality of preview, I could easily run my web page on different browser to see if they are in good shape.



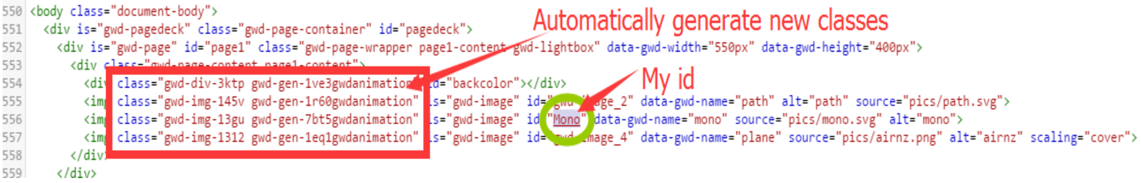
Analysis

To be honest, the code inside the page somewhere is really strange. But somehow I could learn new things from it. For example, my opinion and habit before is that there should only have one <style></style> or one <script></script> tag in the file, however, I am wrong, there can have as many these kind of tags as I wish.



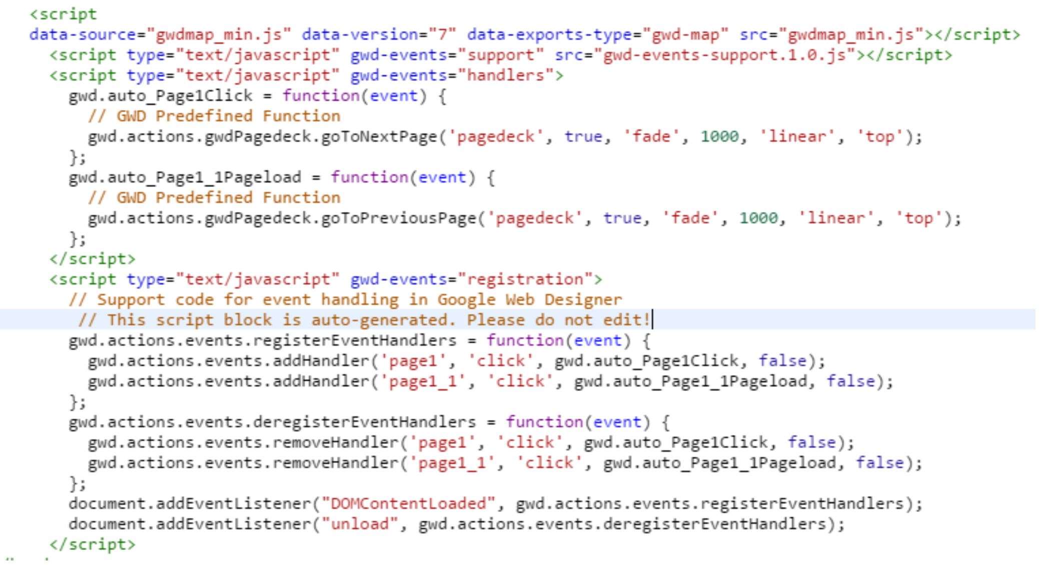
```
<link href="gwdmap_style.css" rel="stylesheet" data-version="7"
<style type="text/css" id="gwd-lightbox-style">
  .gwd-lightbox {
    overflow: hidden;
  }
</style>
<style type="text/css" id="gwd-text-style">
  p {
    margin: 0px;
  }
  h1 {
    margin: 0px;
  }
  h2 {
    margin: 0px;
  }
  h3 {
    margin: 0px;
  }
</style>
<style type="text/css">
  html, body {
    width: 100%;
    height: 100%;
```

I fully understand how the animation works in this webpage because of practical 7. It divides into different stages, the percentage of total animation. I think GWD will name different animation by its id I give to the controller, however, GWD is like to generate random letters as a class and do it.



```
<body class="document-body">
  <div is="gwd-pagedeck" class="gwd-page-container" id="pagedeck">
    <div is="gwd-page" id="page1" class="gwd-page-wrapper page1-content gwd-lightbox" data-gwd-width="550px" data-gwd-height="400px">
      <div class="gwd-page-content page1-content">
        <di  class="gwd-div-3ktp gwd-gen-1ve3gwdanimation  d="backcolor"></div>
        <im  class="gwd-img-145v gwd-gen-1r60gwdanimation"  s="gwd-image" id=   ge_2" data-gwd-name="path" alt="path" source="pics/path.svg">
        <im  class="gwd-img-13gu gwd-gen-7bt5gwdanimation"  s="gwd-image" id "Mono" data-gwd-name="mono" source="pics/mono.svg" alt="mono">
        <im  class="gwd-img-1312 gwd-gen-1eq1gwdanimation"  s="gwd-image" id=   mage_4" data-gwd-name="plane" source="pics/airnz.png" alt="airnz" scaling="cover">
      </div
    </div>
```

The below is the JavaScript stuff, besides it having external js files, the in-file js is kind of hard to understand. After reading these auto-generated code, it is like to establish a new event handler and then use this handler to do the switching page job.



```
<script
data-source="gwdmap_min.js" data-version="7" data-exports-type="gwd-map" src="gwdmap_min.js"></script>
  <script type="text/javascript" gwd-events="support" src="gwd-events-support.1.0.js"></script>
  <script type="text/javascript" gwd-events="handlers">
    gwd.auto_Page1Click = function(event) {
      // GWD Predefined Function
      gwd.actions.gwdPagedeck.goToNextPage('pagedeck', true, 'fade', 1000, 'linear', 'top');
    };
    gwd.auto_Page1_1Pageload = function(event) {
      // GWD Predefined Function
      gwd.actions.gwdPagedeck.goToPreviousPage('pagedeck', true, 'fade', 1000, 'linear', 'top');
    };
  </script>
  <script type="text/javascript" gwd-events="registration">
    // Support code for event handling in Google Web Designer
     // This script block is auto-generated. Please do not edit!
    gwd.actions.events.registerEventHandlers = function(event) {
      gwd.actions.events.addHandler('page1', 'click', gwd.auto_Page1Click, false);
      gwd.actions.events.addHandler('page1_1', 'click', gwd.auto_Page1_1Pageload, false);
    };
    gwd.actions.events.deregisterEventHandlers = function(event) {
      gwd.actions.events.removeHandler('page1', 'click', gwd.auto_Page1Click, false);
      gwd.actions.events.removeHandler('page1_1', 'click', gwd.auto_Page1_1Pageload, false);
    };
    document.addEventListener("DOMContentLoaded", gwd.actions.events.registerEventHandlers);
    document.addEventListener("unload", gwd.actions.events.deregisterEventHandlers);
  </script>
```