SVKM's
Narsee Monjee Mukesh Patel School of Technology Management & Engineering, NMIMS

A REPORT ON

CONSTRUCTION AND MEASUREMENT ETOOL

By

Amrita Ligga

MBA Tech CS

Intern at:

Aptara New Media Pvt Ltd

Contract:

Aptara New Media Pvt Ltd, Pune

A REPORT ON

CONSTRUCTION AND MEASUREMENT ETOOL

By

Amrita Ligga(M519)

MBA Tech CE , MPSTME, NMIMS

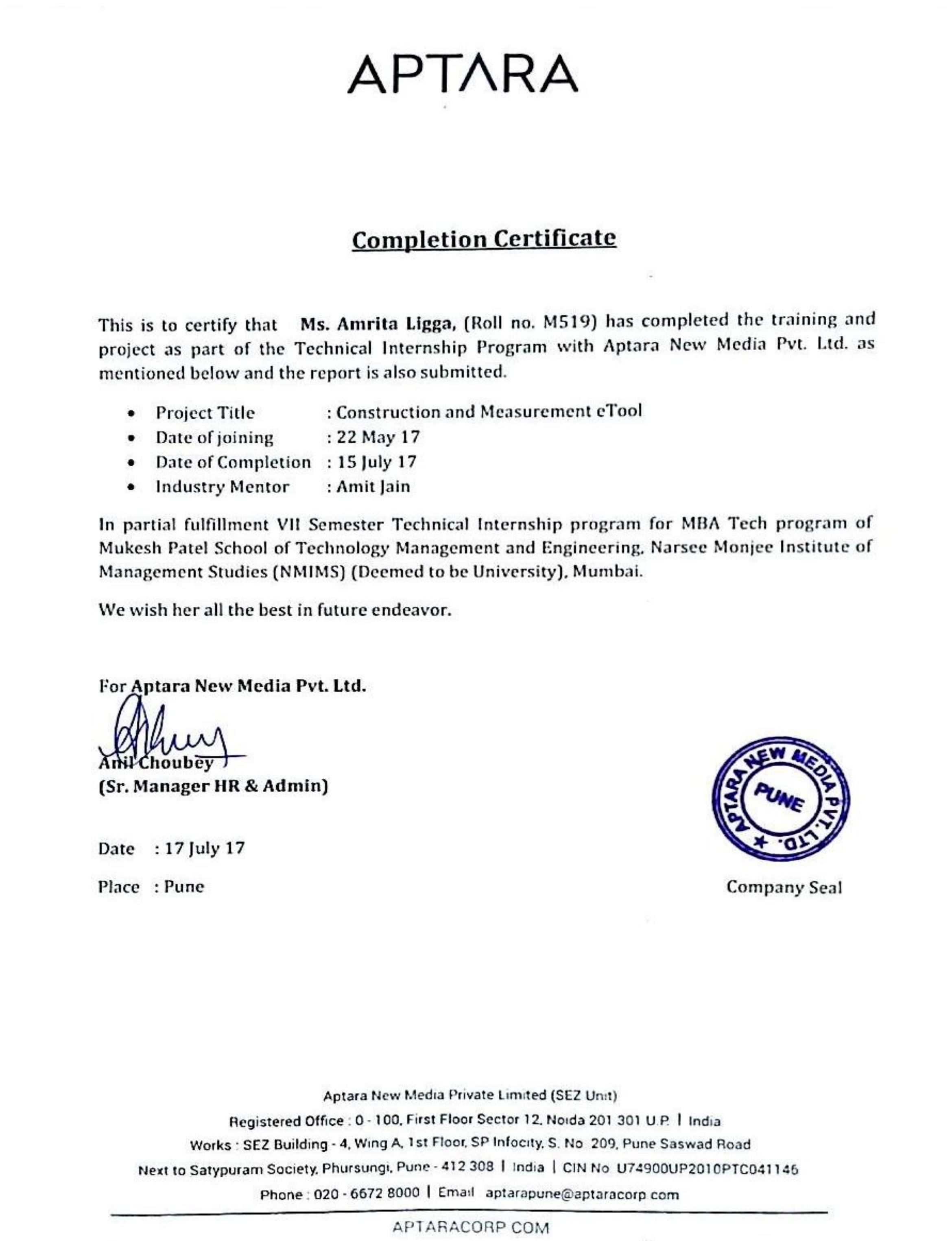
A report submitted in partial fulfillment of the requirements of 5 years Integrated MBA (Tech) Program of Mukesh Patel School of Technology Management & Engineering, NMIMS

Distribution List:

Mr. Sachin Chavan,College Supervisor

Mr.Amit Jain ,Industry Supervisor

Completion Certificate



APTARA

Completion Certificate

This is to certify that **Ms. Amrita Ligga,** (Roll no. M519) has completed the training and project as part of the Technical Internship Program with Aptara New Media Pvt. Ltd. as mentioned below and the report is also submitted.

- Project Title : Construction and Measurement eTool
- Date of joining : 22 May 17
- Date of Completion : 15 July 17
- Industry Mentor : Amit Jain

In partial fulfillment VII Semester Technical Internship program for MBA Tech program of Mukesh Patel School of Technology Management and Engineering, Narsee Monjee Institute of Management Studies (NMIMS) (Deemed to be University), Mumbai.

We wish her all the best in future endeavor.

For **Aptara New Media Pvt. Ltd.**

Anil Choubey
(Sr. Manager HR & Admin)

APTARA NEW MEDIA PVT. LTD. ★ PUNE

Date : 17 July 17

Place : Pune

Company Seal

SVKM's NMIMS

Mukesh Patel School of Technology Management & Engineering

Vile Parle (W), Mumbai - 400056.

TECHNICAL INTERNSHIP REPORT Semester VII – MBA (TECH)

Submitted in Partial Fulfillment of the requirements for Technical Project/Training for VII Semester MBA – (Tech)

Name of the Student: Amrita Ligga

Roll No.: & Batch: M519 / N070

Academic Year: 2017-2018

Name of the Discipline: Computer Science

Name and Address of the Company: Aptara New Media Pvt Ltd, SEZ building 4,Wing A 1st Floor , SP Info City,Pune , Saswad Road Phursungi

Training Period: From 22nd May To 16th July

THIS IS TO CERTIFY THAT

Ms. Amrita Ligga Exam Seat No. has satisfactorily completed her Training/Project Work, submitted the training report and appeared for the Presentation & Viva as required.

External Examiner Internal Examiner Head of Dept. Chairperson/Dean

Date:

Place:

Seal of the University

Acknowledgements

The internship I had at Aptara New Media Pvt. Ltd. was a great chance for learning and professional development. I consider myself very lucky to have been provided with an opportunity to be a part of it .However; this would not have been possible without the kind support and help of many individuals. I would like to extend my sincere thanks to all of them.

I express my deepest gratitude and special thanks to Mr. Samir Kakar, President,Aptara Inc.I am highly indebted to Mr. Amit Jain ,my industry mentor ,for his guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

I would like to express my gratitude towards Mr. Sachin Chavan , my faculty supervisor for his kind co-operation and encouragement which helped me in completion of this project.

I would also like to express my special gratitude and thanks to all the industry persons for giving me such attention and time.

I also take this opportunity to express a deep sense of gratitude to Mrs. Vaishalli Pawar, Placement Head MPSTME, NMIMS for her cordial support and valuable information and guidance that helped me in completing this task through various stages.

Amrita Ligga

TABLE OF CONTENTS

List Of Illustrations

Abstract

The construction and measurement etool is an etool for students to learn construction and measurement using a computer widget. The students use protractor, ruler , line tool , point tool and compass to construct and measure- angles and various other constructions possible through compass and ruler. ‘Construction and Measurement ‘ etool allows students to use a computer widget to learn construction and measurement by using various geometrical instruments that have been created through programming and are easily accessible via a web browser The widget is developed in HTML, CSS , Javascript and Javascript libraries such as Jquery, Createjs/Kineticjs. This project is a live project that has been developed for McGraw Hill Education.

Chapter 1:Introduction

About Aptara

Aptara, Inc. is a US-based media company specializing in digital content development. It is headquartered in Falls Church, Virginia. Aptara's services include content production, digital publishing, data conversion, editorial services, eLearning and technology development, legal content solutions, and business process outsourcing (BPO). Its customers include publishers, information aggregators, professional societies, universities and corporations worldwide. The company has operations in Noida, Sec-60, Pune, Dehradun and Trivandrum, India.Aptara is considered one of the leading knowledge process outsourcing (KPO) companies in India.

Recognition

- The company was one of the world's 50 best-managed BPO vendors according to The Black Book of Outsourcing in 2005.
- In 2014 the company was rated one of the top 100 outsourcing service providers globally according to the International Association of Outsourcing Professionals

Aptara’s digital content, learning and performance, and business support services are in place at market-leading companies worldwide. The industry specialists design and implement strategies that capitalize on new digital and mobile technologies for information providers in IT, law, healthcare, pharmaceuticals, insurance, financial services, and publishing. Aptara's solutions uncover new revenue streams, improve operations, and realize cost savings enterprise-wide.

1.1 Purpose, Scope, and Limitations

Purpose: To develop an etool named construction and measurement that allows students from K-12 to explore construction and measurement with the help of instruments used on paper namely, compass, ruler protractor line and point tool but now to be used on a computer widget.

The tool will help students to explore the following areas:

- **Constructions:**

The student should be able to perform straightedge + compass(ruler and compass) constructions similar to those using physical device, but on computer screen.

- **Measure Angles:**

The student should be able to measure angles.

- **Measure Sides and Segments:**

The student should be able to measure line segments using both customary and metric measure.

Scope: The etool is being developed for MHE . The following has been achieved by the etool

- **Making learning more interactive**

The tool should make learning more interesting, interactive, meaningful and stimulating for the students.

- **Achieving desirable functionality**

To make the following virtual instruments fully functional:

- Ruler
- Protractor

- Line Tool
- Point Tool
- Compass

- **Responsive design:**

The widget must be responsive/reflowable by design and must adapt to any screen resolution / available in given space. For instances where responsive design is not feasible, ratio resize/proportionate scaling must be implemented.

- **User Friendly and Intuitive UI Design**

The instruments viz. ruler , protractor , compass must resemble the real (physical) instruments for maximum intuitiveness. The user interface must be easy to use.

Limitations: The etool uses web browsers to run. Hence if required web browsers are not available the etool won't be able to run efficiently. The etool would not be tested on tier 3 browsers: Edge 12,IE 9, IE 10,Safari 7 on MAC. While viewing the widget if user exits the course or leaves current widget page, the current state of widget will not be saved.

1.2 Sources and methods.

The entire etool application is to be developed using the MVC architectural pattern. The Model-View-Controller (MVC) is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. Each of these components are built to handle specific development aspects of an application. MVC is one of the most frequently used industry-standard web development framework to create scalable and extensible projects.

Model: The Model component corresponds to all the data-related logic that the user works with. This can represent either the data that is being transferred between the View and Controller components or any other business logic-related data.

View: The View component is used for all the UI logic of the application

Controller: Controllers act as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output.

Technologies used

The following technologies with mentioned software and hardware will be used for developing the eTool Application

Component	Components used
Client Side Technology	HTML, CSS, jQuery, JavaScript
Framework	Custom Javascript
Development IDE	Sublime Text
Development environment Hardware	Intel(R) Core TM i3-2100 CPU @3.10GHz, 4.00 GB RAM
Development OS	Windows

Table 1.1:Technologies used

Chapter 2 Study

Before working on the project the following were required to be understood:

2.1. Object oriented programming concepts

OOP refers to using self contained pieces of code to develop applications.These self contained pieces are called objects.This concept can be implemented in different languages like C,C++,Java , Javascript etc.. The first step in OOP is to identify all the objects the programmer wants to manipulate and how they relate to each other, this is often known as data modeling. Once an object has been identified, it is generalized as a class of objects which defines the kind of data it contains and any logic sequences that can manipulate it. All the objects have their own set of properties and methods (logical sequence that manipulates it).

Abstraction

Data abstraction refers to, providing only essential information to the outside world and hiding their background details, i.e., to represent the needed information in program without presenting the details.

Encapsulation

Encapsulation is placing the data and the functions that work on that data in the same place. Data encapsulation led to the important OOP concept of data hiding.

Inheritance

Inheritance is the process of forming a new class from an existing class that is from the existing class called as base class, new class is formed called as derived class. This is a very important concept of object-oriented programming since this feature helps to reduce the code size. Objects can make use of their inherited functionalities and still have their own specialized functionalities.

<u>Polymorphism</u>

The ability to use an operator or function in different ways in other words giving different meaning or functions to the operators or functions is called polymorphism. The concept of overloading is also a branch of polymorphism.

Inheritance and Encapsulation since are the two concepts apply to OOP in JavaScript, particularly because, in JavaScript, objects can encapsulate functionalities and inherit methods and properties from other objects.

2.2 Basics of HTML CSS Javascript

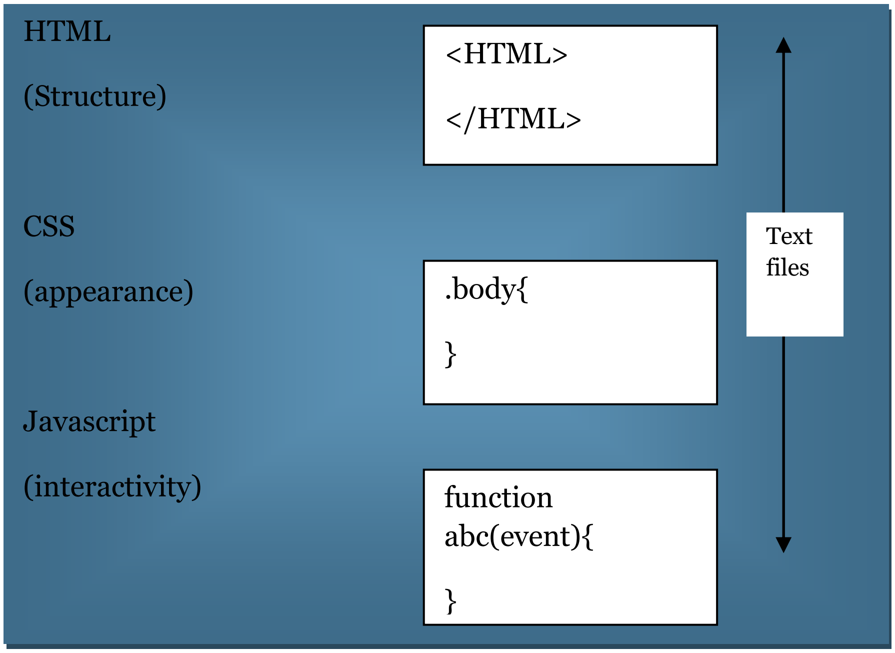


Figure 2.1:HTML CSS and Javascript purpose

In MVC architectural pattern HTML and CSS are used to develop the view component.

HTML stands for Hypertext Markup Language. It consists of some predefined tags that are used to define the structure of the webpage . Before the commencement of actual project following topics under HTML were understood:

- Elements, Attributes, headings
- Paragraphs , links, images, lists
- Building navbar
- HTML canvas

CSS stands for cascading style sheets and is used to control the display of the HTML elements. Following topics were understood under CSS:

- Basic syntax,internal and external style sheet,style attribute(inline)
- Fonts,customizing links,layout control,
- Responsiveness through CSS(media queries)

Javascript is used for interactivity in webpage example interaction through mouseclicks etc. Parts of window are nothing but objects in javascript. The HTML elements in the window can be manipulated by calling functions on objects.Manipulation can also take place with the help of JQuery which is a Javascript library to simplify javascript programming.

2.3 Mouse events in javascript

- onmouseover/onmouseout - When the mouse passes over an element
- onmousedown/onmouseup - When pressing/releasing a mouse button
- onmouseup – When releasing a mouse button
- onmousedown - When mouse is clicked
- onmousemove/onmouseout - When moving the mouse pointer over/out of an image

Out of these mousedown , mouseup,mousemove events are used predominantly in the project.

Chapter 3:Overall Schedule

Activity	From date	To date	Output
Week 1			IIR report
Understanding required technologies necessary for the fulfillment of project requirements. Understanding industrial environment and practices.	22.05.17	29.05.17	
Week 2			Project proposal
Assigning of project/module. Understanding of project	30.05.17	31.05.17	
Identifying functional and system requirements	1.05.17	2.05.17	
Week 3			Mockups, code examples
Study of Javascript libraries	5.06.17	7.06.17	
Deciding the responsive structure of UI	7.07.17	9.06.17	
Preparation of UI	6.07.17	9.06.17	

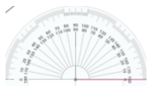
mockups by Graphics Team			
Week 4			Approved Compass images, Line tool study, interim report
Getting the UI mockups approved from client for etool	12.06.17	12.6.17	
Study of how line tool was implemented	12.06.17	16.06.17	
Collection of compass images	16.06.17	16.06.17	
Week 5			Compass module 20%completed
Rendering the compass image on canvas	19.06.17	21.06.17	
Adding draggable functionality to compass	22.06.17	23.06.17	
Week 6			Compass module 50%completed
Change radius functionality	26.06.17	30.06.17	

Week 7			Project Report
Arc drawing functionality along with compass rotation	3.07.17	5.07.17	Compass module 80%completed
Opacity/Thickness change functionality	6.07.17	6.07.17	
Lock Radius functionality	7.07.17	7.07.17	Compass module completed
Week 8			TIP completion
Testing and resolving errors	10.07.17	14.07.17	

CHAPTER 4: Functional and system requirements of the etool.

1.Ruler:

- User can add a 6 inch ruler to the workspace (standard/metric).
- User can drag ruler and rotate.
- User can adjust transparency of ruler.
- User can add a 6 inch (standard) ruler to the workspace.
- User can toggle to a 150cm/mm ruler (metric).



2 . Protractor:

- User can add 180-degree or 360-degree protractor to canvas and rotate freely.
- User can adjust the degree line, set transparency, manually enter a measurement, turn off degree display, reverse angle, and adjust transparency.
- User can create a ray or angles that can be moved around the workspace.

3. Compass:

- User can move the compass around the canvas and modify the width.
- User can lock the width of the compass with the Lock button.
- User can draw full or partial circles.
- User can draw a quick circle using a Quick Circle button.
- User can modify thickness and color of line drawn by pencil.
- User can modify the transparency of the arc drawn by compass.

4. Line Tool:

- User can add line segment, line segment with endpoints, ray, line (with arrows on ends) User can set thickness, transparency, and line color.
- User can activate Add **Point tool** to add point on line, ray, or segment.

- User can rotate resize lines, segments, or rays drawn on canvas.

The module which was given to me as an intern was implementation of Compass tool. The following is a usecase diagram for compass tool:

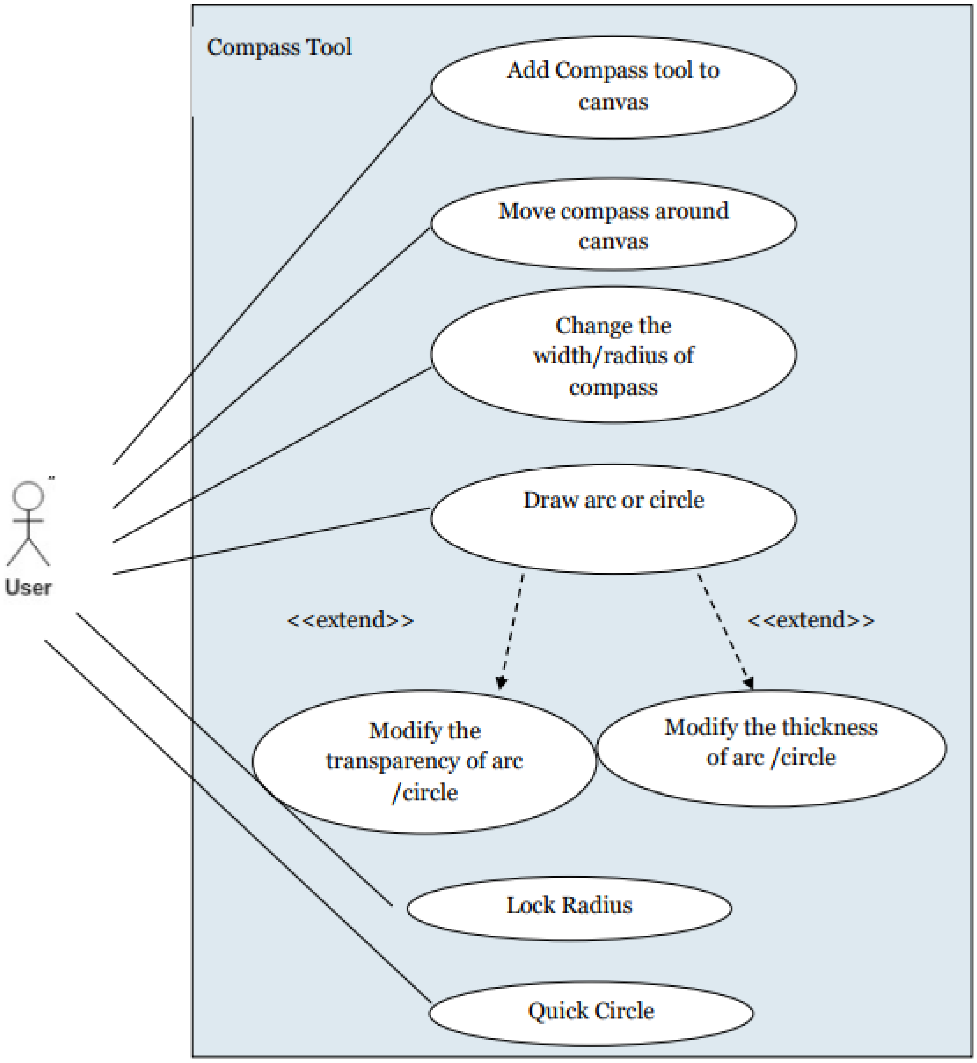

Compass Tool
Add Compass tool to canvas
Move compass around canvas
Change the width/radius of compass
Draw arc or circle
<<extend>>
<<extend>>
Modify the transparency of arc /circle
Modify the thickness of arc /circle
Lock Radius
Quick Circle
User

Figure 4.1: Use case diagram

Following are the requirements on the deployment machine with respect to web browsers

Tier 1 Browsers		
Chrome 30+ IE 11 Safari 8+	IE 11, MS Surface MobileSafari8+ Chrome 50+	IE 11 MobileSafari8+ Chrome 30+
Tier 2: Mitigated Support - Features will be tested on a time available basis for this browser. Any found blocking (or reported) bugs whose severity is considered blocking will be closed.		
Tier 2 Browsers		
Firefox 31+	Chrome 47+ Mobile Safari 7	Mobile Safari 7
Tier 3: Support Deprecated - Features will not be tested on these browsers, however reported/found bugs with severity that is considered blocking will be closed.		
Tier 3 Browsers		
Edge 12 IE 9, IE 10 Safari 7 on MAC		

Table 4.1: Browser Tier Table

CHAPTER 5: Designing

5.1 Responsiveness

Responsive Web design is the approach that suggests that design and development should respond to the user's behavior and environment based on screen size, platform and orientation.

The practice consists of a mix of flexible grids and layouts, images and an intelligent use of CSS media queries.

The tool will have following 3 areas:

- Canvas Area: Area to drag-drop/ add tools (Line, Point, Ruler, Protractor, Compass) from toolbar.
- Controls Area: Controls (Reverse, Create Ray, Create Angle, Level) will be visible in controls area depending on what configuration is set.
- Toolbar: To access Line, Point, Ruler, Protractor, Compass tools and also other tool buttons like undo and clear canvas.

The positioning of these areas varies depending on the device it is viewed on. The HTML will adapt to all browser and device specifications and apply different style sheets, as and when required, based on screen size (all driven by CSS3 guidelines), this provides a framework that helps modify the layout according to the user's screen resolution

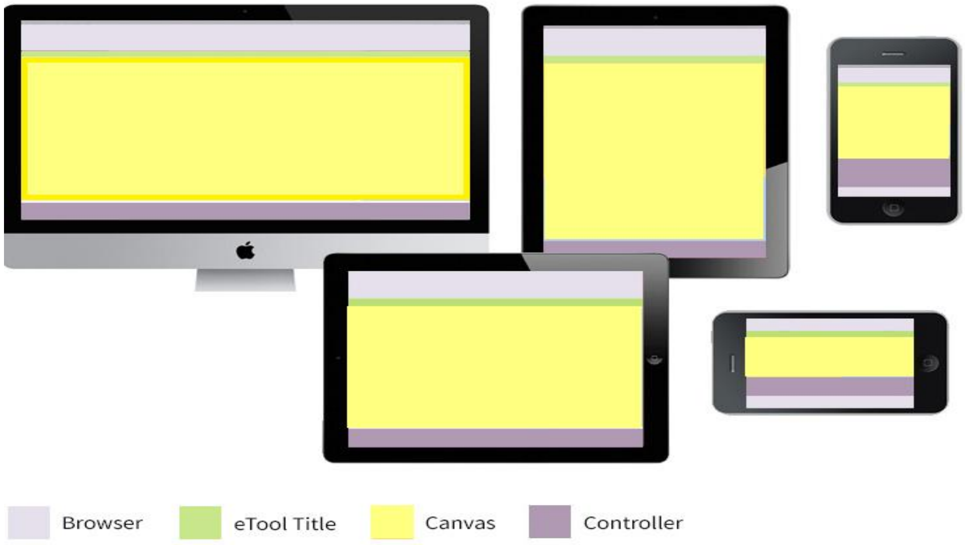


Figure 5.1:Components of UI

```
@media (max-width: 991px) {

 #cmt_tool_container .dd-main-container{

   margin-left: 64px;

 }

}

@media (max-width: 700px) {

 # cmt_tool_container .dd-main-container{

   margin-left: 45px;

}

@media (max-width: 670px) {

 #cmt_tool_container .dd-main-container{

   margin-left: 37px;
```

```
  }

}

@media (max-width: 637px) {

  #cmt_tool_container .dd-main-container{

    margin-left: 57px;

  }

}

@media (max-width: 615px) {

  #cmt_tool_container .dd-main-container{

    margin-left: 77px;

  }

}

@media (max-width: 623px) {

  #mobile_shapes_div{

    margin-left: 63px;

  }

}

@media (max-width: 559px) {

  #cmt_tool_container .dd-main-container{

    margin-left: 65px;
  }

}

@media (max-width: 465px) {

  #mobile_shapes_div {

    margin-left: 98px;

  }

}
```

5.2 Mockups:

A graphic mock-up is simply a non-working image of the design created using graphic design software. The mock-up is often a fully detailed version of the design as it is intended to be viewed when created in HTML and CSS. the mock-up is often created by a graphic designer and sent on to a coder to be developed in HTML and CSS.

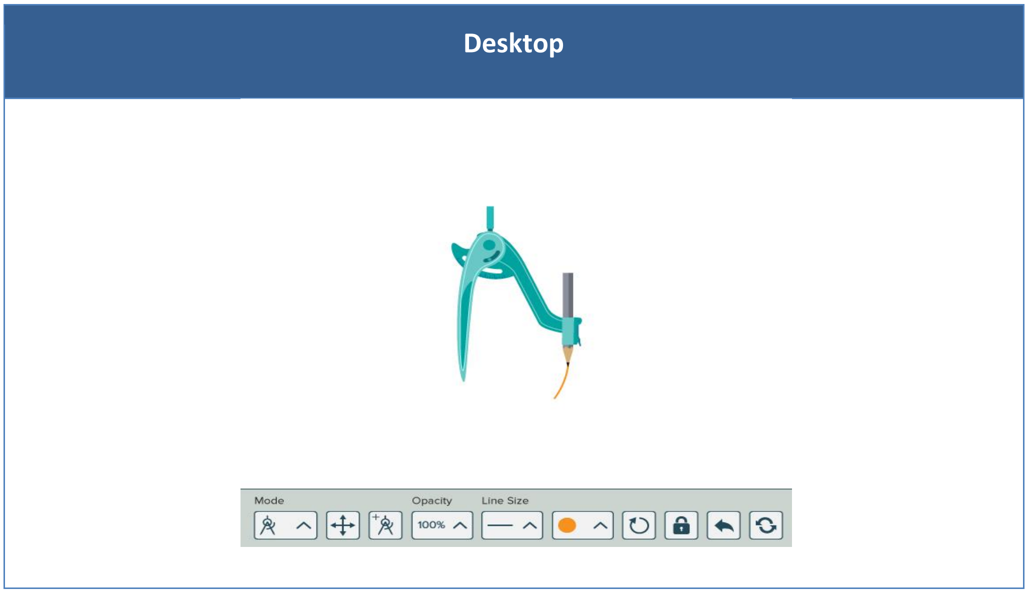


Figure 5.2: Compass Mockup

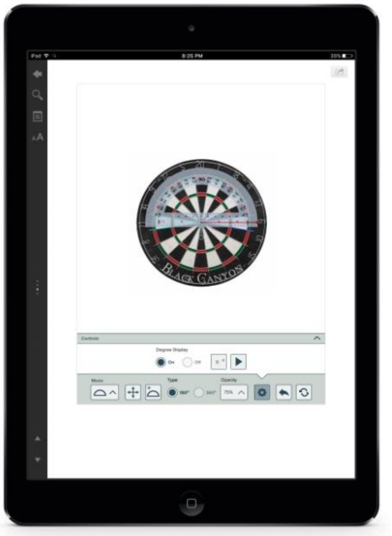
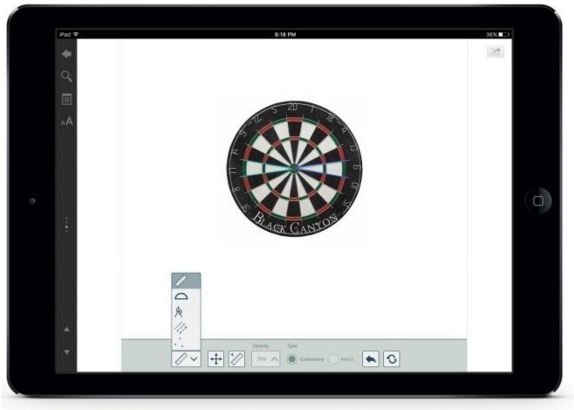
iPad Portrait	iPad Landscape
	

Figure 5.3: Protractor Mockup

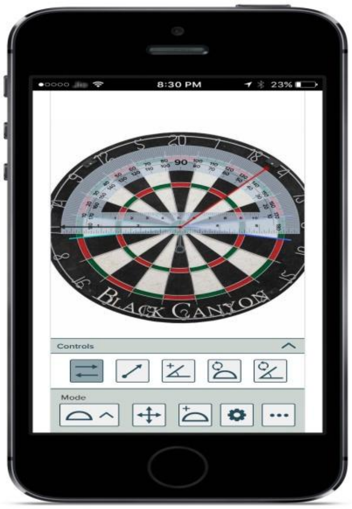
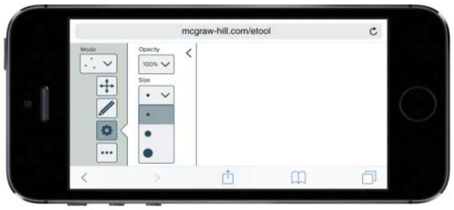
iPhone Portrait	iPhone Landscape
	

Figure 5.4: Protractor and point tool mockups

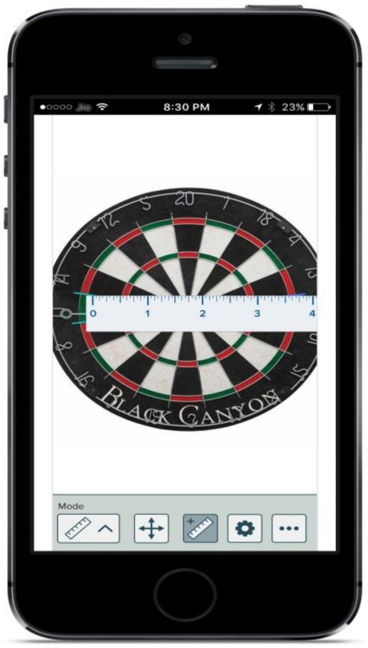
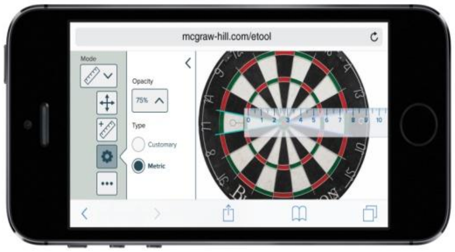
iPhone Retina Portrait	iPhone Retina Landscape
	 mcgraw-hill.com/etool Mode Opacity 75% Type Customary Metric

Figure 5.5:Ruler mockup

CHAPTER 6 Javascript Libraries used

Why do we need libraries?

All the libraries have one aim of abstracting the difficult implementation and simplifying coding in Javascript. Do more with less code –Most of the libraries allow chaining of methods which is used to simplify code in scenarios that involve calling multiple functions on the same object consecutively. Eg:

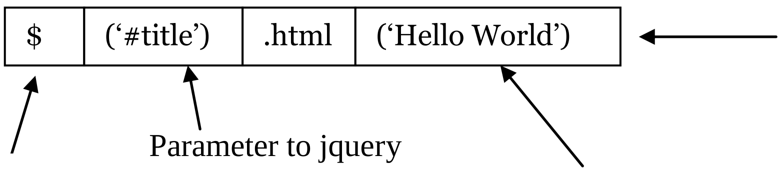
```
$('#my-div').css('background', 'blue').height(100).fadeIn(200); //jquery example
```

Most of the functionalities that we need are already implemented by a good programmer through libraries .So instead of writing our own code , we can use the methods in libraries easily .This reduces our effort. Using libraries saves our time. These libraries also handle cross browser differences. Many of the libraries are regularly maintained by updating to new versions by the library creators.

Jquery

jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers.

Example:



$ | ('#title') | .html | ('Hello World')

Parameter to html() .Sets the html to the specified string.

Shorthand for jquery function

Parameter to jquery function which is nothing but an id of an HTML element.

This $() function returns an HTML object on which a

Html() is a function which is called on HTML element with id title .This function is used to change the HTML

Createjs

Createjs is a suite of javascript libraries created by Grant Skinner which consists of 4components easeljs , preloadjs, soundjs , tweenjs. This suite of library is being used in the development of etool.

Easeljs- A JavaScript library that makes working with the HTML5 Canvas element easy. It is a library for building high-performance interactive 2D content in HTML5. It provides a feature-rich display list to allow us to manipulate graphics. It also provides a robust interactive model for mouse and touch interactions.

Tweenjs: A JavaScript library for tweening and animating HTML5 and JavaScript properties. It supports tweening of both numeric object properties & CSS style properties, and allows you to chain tweens and actions together to create complex sequences.

Soundjs: A JavaScript library that lets you easily and efficiently work with audio on the web. It provides a consistent API for playing audio in different browsers, including using a target plugin model to provide an easy way to provide additional audio plugins like Web Audio, and a Flash fallback. A mechanism has been provided for easily tying in audio preloading to PreloadJS.

Preloadjs: A JavaScript library that lets you manage and co-ordinate the loading of assets. It provides a consistent API for loading different file types, automatic detection of XHR (XMLHttpRequest) availability with a fallback to tag-base loading, composite progress events, and a plugin model to assist with preloading in other libraries such as SoundJS.

In this project easeljs is used in drawing lines(in line tool).

Kineticjs

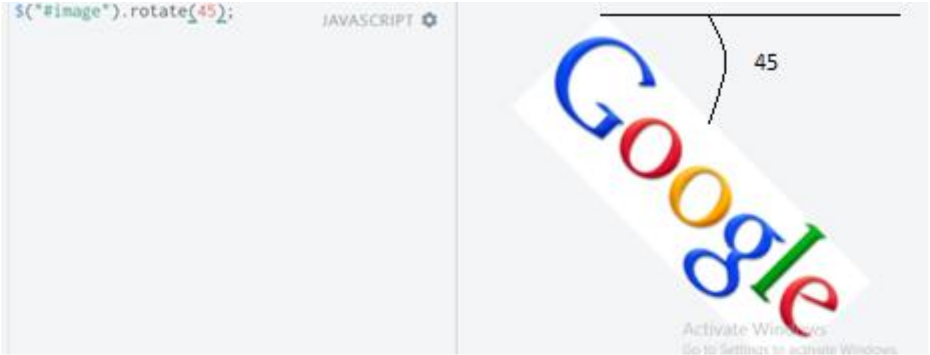
KineticJS is a fast, robust, HTML5 Canvas Library that is no longer maintained. It is used to make arcs in compass tool as mentioned in section 7.2

Jquery-rotate

OpenSource Cross-browser jQuery plugin to rotate image by any-angle.Its use has been explained in section 7.1.

Example:

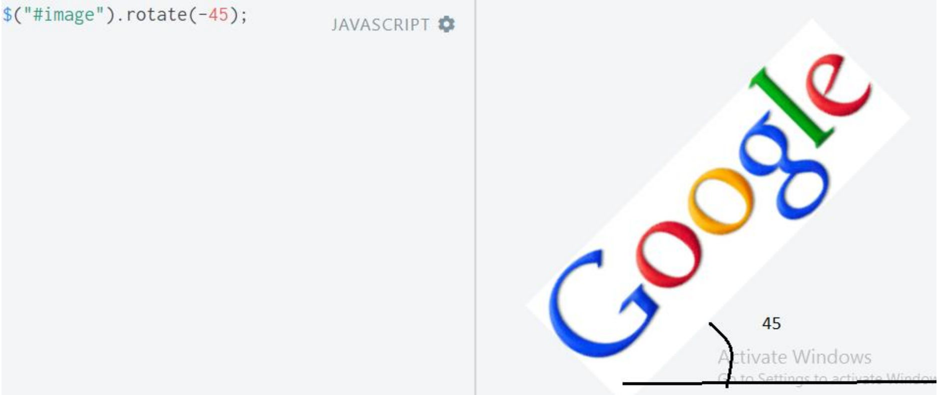
Rotating by 45 degrees:



```
$("#image").rotate(45);
```

Figure 6.1: Rotate by 45degrees

Rotating by -45 degrees



```
$("#image").rotate(-45);
```

Figure 6.2: Rotate by -45 degrees

<u>Rotation illustration by jquery rotate</u>

CHAPTER 7: The Compass Tool

7.1 Rendering a compass image on screen

The compass is not rendered as a single piece instead is divided into following div elements which have got their own positioning:

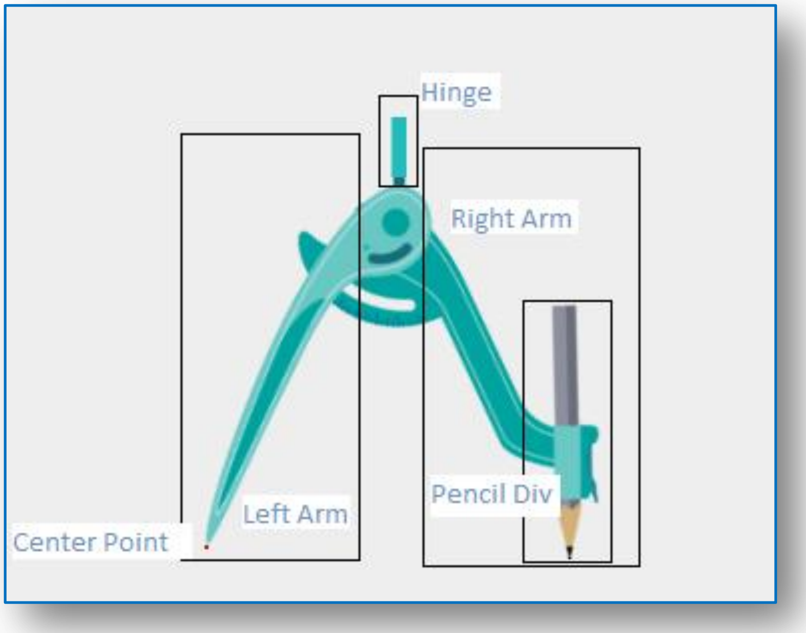


Figure 7.1: Compass broken into its components

This compass was rendered in the following way:

1. Create div elements in HTML for the above parts.

2. Position them with the help of css and apply the corresponding images of different parts.For centerPoint, image is not needed instead a background of red color is required.

3. Since the images are not inclined at an angle we want, we need to use rotate method of jqueryRotate as:

```
$('.rightarm').rotate(-rotationAngle);        //rotationAngle being initialized to 30
```

It is important to note that both the right and left arm must be rotated by same angle but in opposite direction.

CSS properties used:

transform : matrix() - The matrix() method combines all the 2D transform methods into one.The matrix() method take six parameters, containing mathematic functions, which allows you to rotate, scale, move (translate), and skew elements.The parameters are as follows: matrix(scaleX(),skewY(),skewX(),scaleY(),translateX(),translateY())

Transform:origin- The transform-origin property allows you to change the position of transformed elements.

Left , top - For absolutely positioned elements, the left property sets the left edge of an element to a unit to the left/right of the left edge of its nearest positioned ancestor. For absolutely positioned elements, the top property sets the top edge of an element to a unit above/below the top edge of its nearest positioned ancestor

Background - The background CSS property is a shorthand for setting the individual background values in a single place in the style sheet. background can be used to set the values for one or more of: background-clip, background-color, background-image, background-origin, background-position, background-repeat, background-size, and background-attachment.

z-index- The z-index property specifies the stack order of an element.An element with greater stack order is always in front of an element with a lower stack order

7.2 Drawing an arc on canvas:

Kineticjs

Kineticjs has been used in compass tool to create an arc in the following way:

```
arc = new Kinetic.Arc({
        innerRadius: radius,

        outerRadius: radius,

        stroke: strokeColor,

        strokeWidth: 2,

        angle: currentAngle,

        x: radius + arcPadding,

        y: radius + arcPadding,

        opacity: 1});
```

This arc object is added to a layer object which in turn is added to stage object which is a container for display objects.

7.3 Making the compass draggable

With the help of jquery, any div element can be made draggable with draggable function. It enables to move the draggable object by clicking on it with the mouse and dragging it anywhere within the viewport.It can be implemented as follows:

```
<!doctype html>

<html lang="en">

<head>

 <title>jQuery UI Draggable - Default functionality</title>

  <style>

  #draggable { width: 150px; height: 150px; padding: 1em; }

  </style>

  <script src="https://code.jquery.com/jquery-1.12.4.js"></script>
  <script src="https://code.jquery.com/ui/1.12.1/jquery-ui.js"></script>
  <script>
  $( function() {
    $( "#draggable" ).draggable();
     var x = $("#pp").position();
        console.log("Top position: " + x.top + " Left position: " + x.left);
  } ); </script>
</head>
<body>
 <div id="draggable" class="ui-widget-content">
  <p id="pp">Drag me around</p>
</div>
</body>
</html>
```

7.4 Handling the mouseevents

As mentioned earlier there are two mouse events on compass that are listened:mousedown and mouseup.For these events eventhandlers are written in javascript file for compass. The main requirements are:

- Changing the radius of the compass
- Drawing the arc on canvas

For these two requirements different functions are created that handle respective event. The user can click on various div elements and for each element a different handler must be called. For instance if the user clicks on the hinge part of the compass , the compass must be able to get dragged and if the user clicks on right arm (handle) then draggable property must be disabled and on movement of mouse the radius of the compass must change. Similarly on clicking pencil div,another eventhandler must be called to actually draw the arc on canvas.

For attaching an event handler to an event,bind method is used as follows:

$('.handle').bind("mousedown", handleDragStart)

$('.pencil').bind("mousedown", pencilDragStart)

These lines are part of another function addButtonEvents that is called on document ready.These handlers call handleDragging and pencilDragging functions.

On mouseup event, two other handlers are called namely handleDragEnd() and pencilDragEnd().These functions unbind the handlers for mousemove and mouseup events.

In place of bind addEventListener() can also be used but it is not supported in browsers IE8 or lower.

7.5 Use of trigonometry

The compass tool involves using many trigonometric concepts and javascript functions that support trigonometry.Math object of javascript is used extensively. The Math object allows you to perform mathematical tasks. All properties/methods of Math can be called by using Math as an object, without creating it.

- Math.atan: Return the arctangent of a specified number: as a value between -PI/2 and PI/2 radians or NaN if the value is empty.
- Math.sin: The sin() method returns the sine of a number. Return Value: A Number, from -1 to 1, representing the sine of an angle, or NaN if the value is empty

- Math.cos: The cos() method returns the cosine of a number. A Number, from -1 to 1, representing the cosine of an angle, or NaN if the value is empty
- Math.PI: The PI property returns the ratio of a circle's area to the square of its radius, approximately 3.14. Return Value: A Number, representing PI

- Math.sqrt: The sqrt() method returns the square root of a number. Return Value: A Number. If x is a negative number, NaN is returned.
- Math.abs: The abs() method returns the absolute value of a number.

The trigonometric concepts are used to find out the position of hinge element. These are also required to calculate the angle by which both the left and the right arms of the compass need to be rotated if the radius of the compass is changed.The updated radius is also calculated with these methods of Math object.

7.5.1 Calculating angle

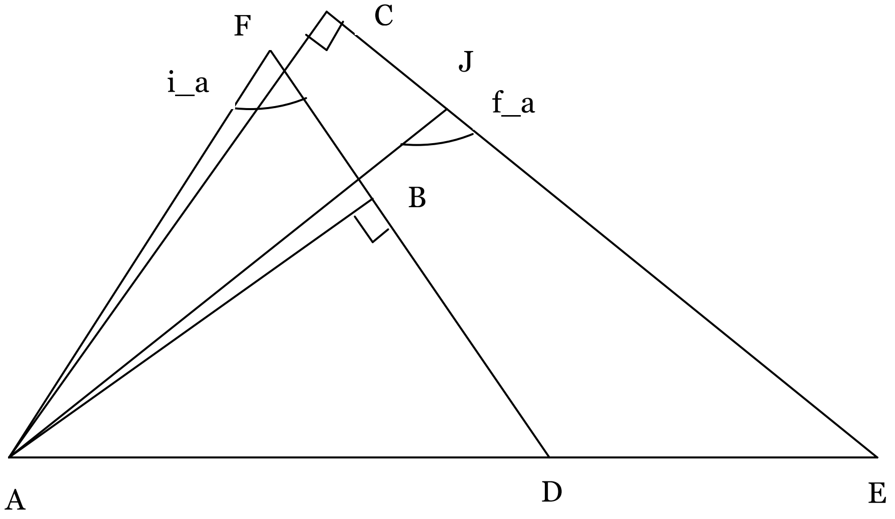


Figure 7.2 :Calculating angle on handleDrag

AFD: previous compass position AJE:new compass position

New angle=previous angle + (AC-AB)/4

There are limits on maximum and minimum angle possible.(70 and 10 degrees).

7.5.2 Calculating position of hinge:

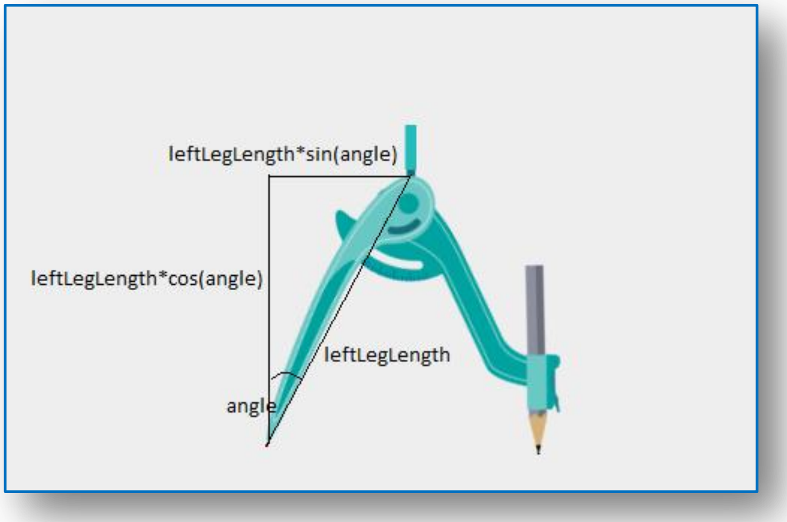
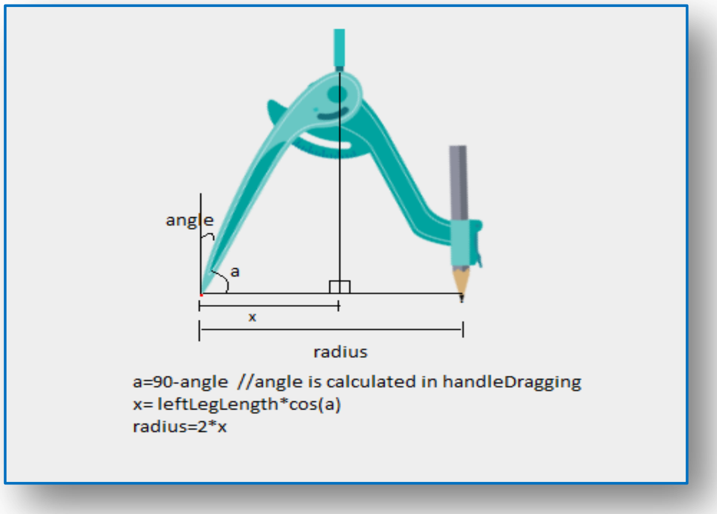


Figure 7.3 : Calculation of hinge position

var a = Math.cos(angle * Math.PI / 180) * leftLegLength

var b = Math.sin(angle * Math.PI / 180) * leftLegLength

7.5.3 Updating radius:



a=90-angle //angle is calculated in handleDragging
x= leftLegLength*cos(a)
radius=2*x

Figure 7.4: Updating radius

7.6 Calculation of angle for compass on Pencil Drag

The angle by which entire compass div must be rotated when pencil div is being dragged is calculated in the following way:

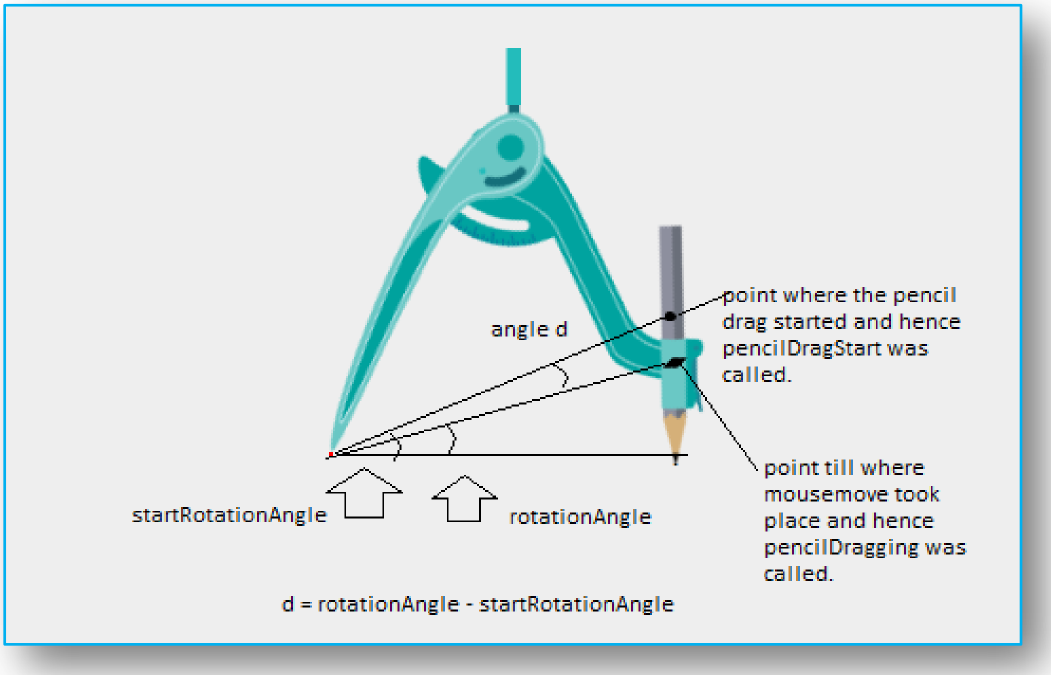

angle d
point where the pencil drag started and hence pencilDragStart was called.
point till where mousemove took place and hence pencilDragging was called.
startRotationAngle
rotationAngle
d = rotationAngle - startRotationAngle

Figure 7.5: rotationAngle

This d is the angle by which the compass div must be rotated. This is done as:

compass.rotate(-d) ;

And, the compass gets rotated by this difference as shown in the next figure:

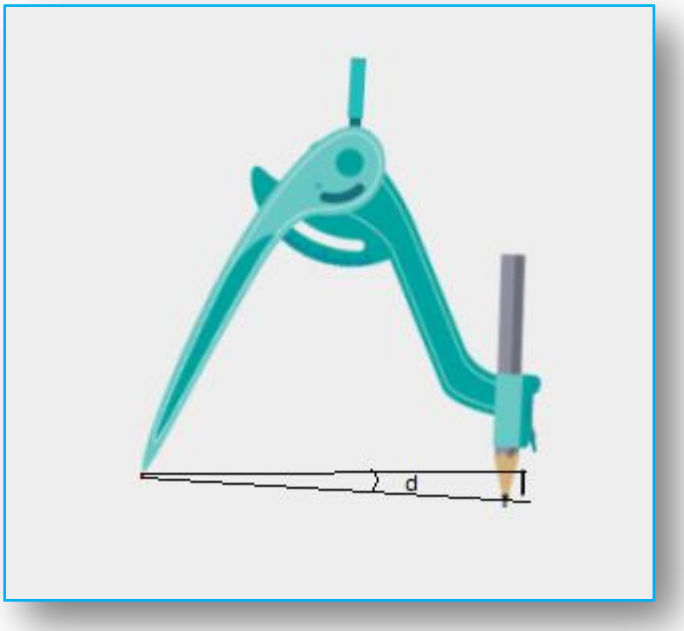


Figure 7.6: Compass rotation

7.7 Lock Radius

The compass radius can be locked with the help of lock button .This is implemented in the following way:

```
var oLockButton= $("<div>", {id: "cp_lockButton", title:'Lock Radius',
'class':'control_div'}).bind('click keyup',handleLockEvents);
```

The above line creates a div element with the help of jquery.The element is given its id, class and title.Along with that an event handler is bound to the element on a click event called handleLockEvents.

```
toolBoxDiv.append(oLockButton);
```

This line adds the div element into toolBoxDiv.

```
function handleLockEvents(e){

                    $("#cp_lockButton").toggleClass("selected");

                    if($("#cp_lockButton").hasClass("selected")){

                              locked = true; //lock radius

                    }else{

                              locked = false;   //unlock radius.

                    }

}
```

The handleLockEvents function which is called when the div is clicked ,is defined as above. Here, a class is added to div element when the div is clicked for the first time and then removed when div is clicked for the second time .This toggling functionality is added with the help of jquery's toggleClass() method. Selected is a class defined in CSS that changes the background color of the div element to show that it has already been selected and lock functionality is active.If the div element has this class then lock functionality must be active therefore locked variable is set to true otherwise it is set to false.

This value of locked variable is checked before updating radius of compass.(in function handleDragStart() explained in next chapter).

7.8 Quick Circle

The user can use this button to create a circle at the current position of compass

```
var oQuickCircleButton= $("<div>", {id: "cp_quickCircleButton", title:'Quick Circle',
'class':'control_div'}).bind('click keyup',handleQuickCircleEvents);
```

The above line creates a div element with the help of jquery.The element is given its id, class and title.Along with that an event handler is bound to the element on a click event called handleQuickCircleEvents

```
toolBoxDiv.append(oQuickCircleButton);
```

This line adds the div element into toolBoxDiv

```
function handleQuickCircleEvents(e){

//creation of arc container

//creation of stage

//creation of kinetic js layer

var d = new Kinetic.Circle({                    //for creation of a dot at the center
of circle

                    radius: centerDotRadius,            //constant

                    x: radius + arcPadding,

                    y: radius + arcPadding

               });

var f = new Kinetic.Circle({

                    radius: radius,

                    stroke: selectedColor,

                    strokeWidth: selectedStrokeWidth,

                    x: radius + arcPadding,

                    y: radius + arcPadding,

                    opacity: selectedOpacity

}

  //Add both circles to kinetic js layer.Then add this layer to stage.
```

7.9 Opacity

The users can set the transparency of the arcs/circles drawn.The Kinetic.arc object has an attribute called opacity which sets the transparency of the arc/circle to be drawn . The value of this attribute can range from 0 and 1.At 0% transparency opacity must be 1 .At 25% transparency opacity must be 0.7 and so on.For this a switch case is used and accordingly selectedOpacity is given a value.

```
{
        case '0':  selectedOpacity = 1;
                         break;
        case '25': selectedOpacity = 0.7;
                         break;
        case '50': selectedOpacity = 0.4;
                         break;
        case '75': selectedOpacity = 0.2;
                         break;

}
```

In kinetic.arc creation, opacity attribute is assigned this selectedOpacity variable.

7.10 Thickness of arc

The users can set the thickness of the arcs/circles drawn.The Kinetic.arc object has an attribute called strokeWidth which sets the thickness of the arc/circle to be drawn.The user is given 3 choices of line size :Thin , Thick and medium.

If thin is selected, strokeWidth is set to 1.For medium it is set to 3 and for thick it is set to 5.Here also switch case is used

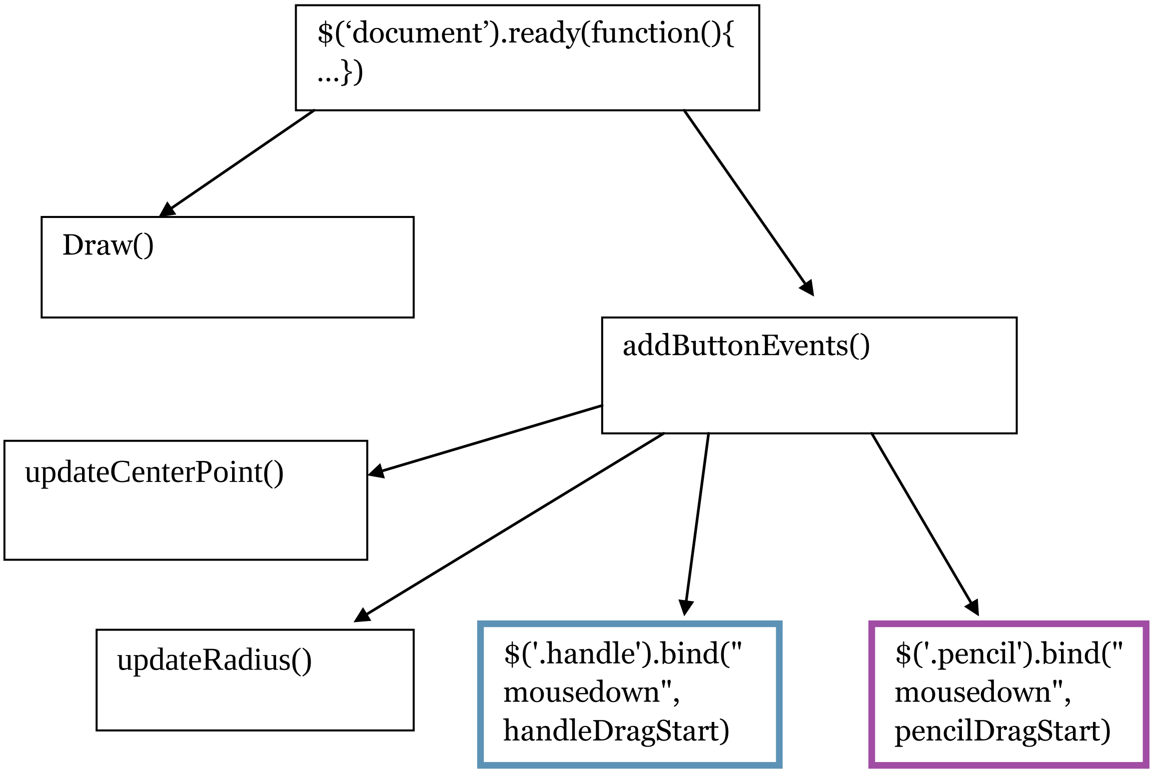
```
switch(lineSize){

                        case "Thin":

                                selectedStrokeWidth = 1;

                        break;

                        case "Medium":

                                selectedStrokeWidth = 3;

                        break;

                        case "Thick":

                                selectedStrokeWidth = 5;

                        break;

                }
```

In kinetic.arc creation, strokeWidth attribute is assigned this selectedStrokeWidth variable.

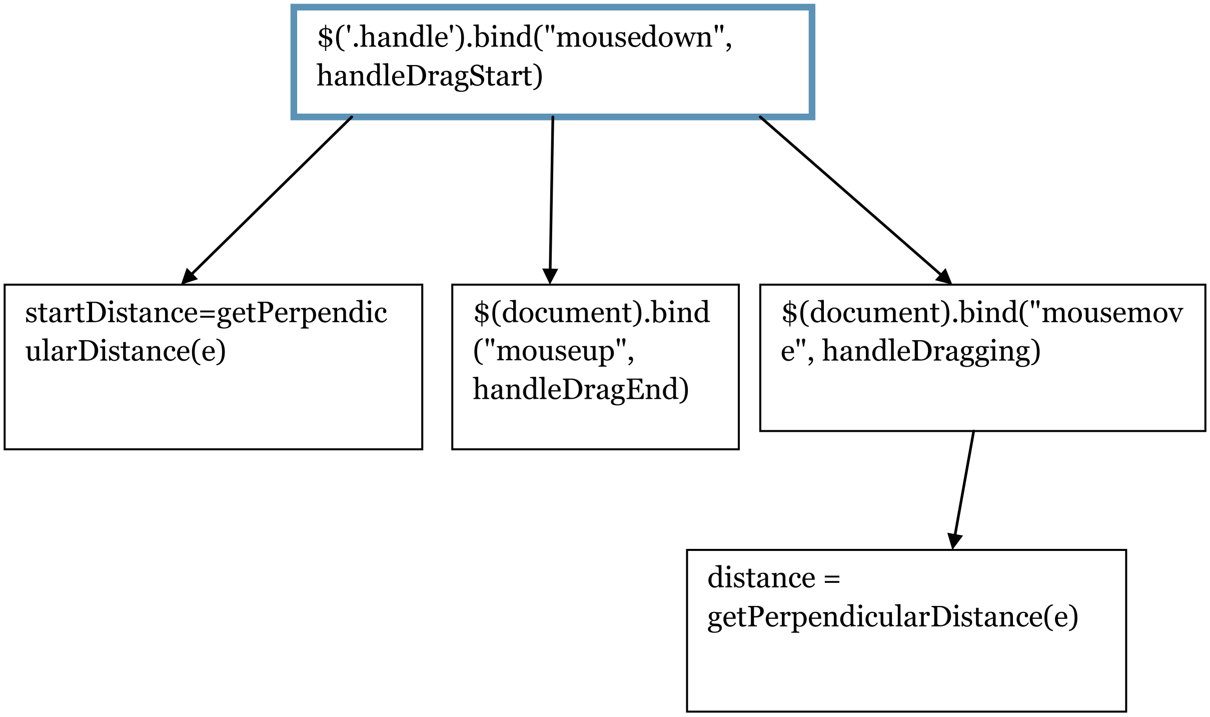
Chapter 8 Control / Program Flow

This chapter explores the logical sequence of function calls in the javascript file starting from ready function called on document ready to pencilDragEnd .The flow branches off at a point where user has option to either to click on handle or to click on pencil (*highlighted with different color boxes*) hence two kinds of functionalities have been implemented. The arrows mean a function call from within another function .At arrow head the function that is called is specified and at tail the function that calls is specified. Sometimes in the boxes the entire statement is written that calls a function on mouse event.

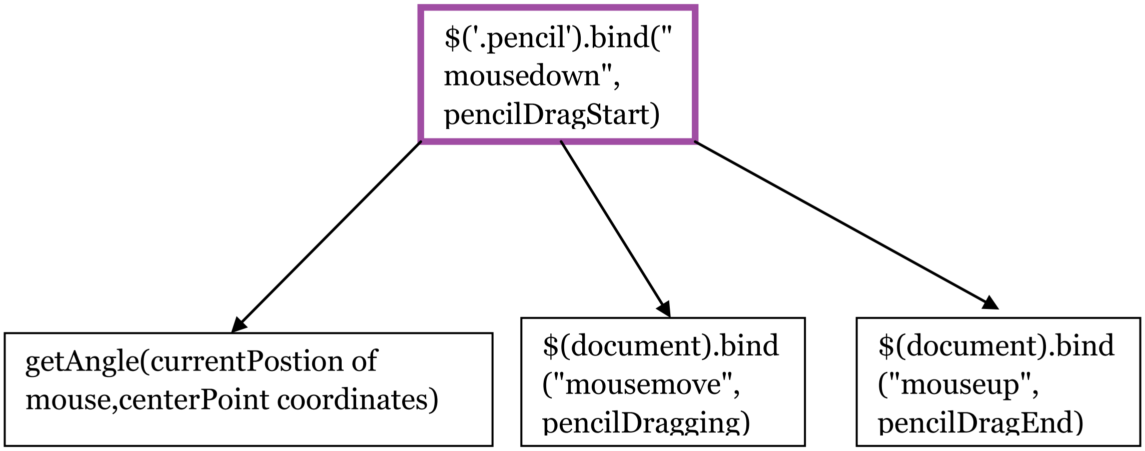
After the flow diagrams, each and every function is explained i.e what that function does and which functions it calls.


$('document').ready(function(){ ...})
Draw()
addButtonEvents()
updateCenterPoint()
updateRadius()
$('.handle').bind(" mousedown", handleDragStart)
$('.pencil').bind(" mousedown", pencilDragStart)

Handle Dragging functionality:


$('.handle').bind("mousedown", handleDragStart)
startDistance=getPerpendicularDistance(e)
$(document).bind ("mouseup", handleDragEnd)
$(document).bind("mousemove", handleDragging)
distance = getPerpendicularDistance(e)

Pencil dragging functionality


$('.pencil').bind(" mousedown", pencilDragStart)
getAngle(currentPostion of mouse,centerPoint coordinates)
$(document).bind ("mousemove", pencilDragging)
$(document).bind ("mouseup", pencilDragEnd)

Explanation:

Ready function: This is a jquery function which is called when DOM has been loaded.If we have some JavaScript code that needs to access the DOM structure to change the properties of some HTML object, you need to wait till the DOM has been loaded completely.It is a good practice to place all the function calls in ready method. Here with the help of jquery,variables are assigned their corresponding elements

addButtonEvents(): In order to handle mouse events on handle and pencil this function is used.It also calls updateRadius(),updateCenterPoint(),updatePencilPoint().

updateCenterPoint():Uses jquery’s getBoundingClientRect method to calculate the left and top position of center point.

updateRadius(): This function calculates radius of compass as mentioned in section 7.5.3

handleDragStart():to get initial perpendicular distance and assign handleDragging as event handler on mousemove event.

handleDragging():to get current perpendicular distance and subsequently calculating angle as mentioned in section 7.5.1. It also calls draw method.

Draw() method: It calculates the hinge position as mentioned in section 7.5.2.It also uses jquery’s rotate method to rotate the left and right arm of the compass.

getPerpendicularDistance():It calculates the perpendicular distance between the centerpoint and the right arm for current position of compass.

handleDragEnd():It gets called if there is a mouseup event after mousedown/mousemove event on handle.It unbinds the handleDragging and handleDragEnd handlers with mousemove and mouseup events.

pencilDragStart(): It includes creation of arc container and arc with the help of kinetic js,calculation of startRotationAngle with the help of getAngle method and binding mousemove event with event handler pencilDragging and mouseup event with pencilDragEnd.

getAngle():This method is called to find out startAngle and rotationAngle.

pencilDragging(): It calculates rotationAngle with the help of getAngle method .Then it calculates difference between startRotationAngle and rotationAngle to calculate the offset by which the compass must be rotated. For this rotate method of jquery is used on entire compass div.

pencilDragEnd():It gets called if there is a mouseup event after mousedown/mousemove event on pencil.It unbinds the pencilDragging and pencilDragEnd handlers with mousemove and mouseup events.

CHAPTER 9 Line tool study

Line creation with createjs

Createjs is a suite of libraries that consist of libraries: Preloadjs,easeljs,soundjs and tween js.EaselJS is a library for working with the HTML5 canvas. It consists of a full, hierarchical display list, a core interaction model to make working with Canvas much easier. We need to define a canvas element in HTML and reference it in JavaScript. EaselJS uses the notion of a Stage which is the top level Container for the display list. EaselJS comes with a Graphics class that exposes an easy to use API for generating vector drawing instructions and drawing them to a specified context. The commands are very similar to the normal HTML5 Canvas, while EaselJs adds some of its own new commands as well. We normally do not use the Graphics class by itself, but rather access it by the Shape class. The following html enables drawing lines through mouseup mouse down events using createjs.

```
<Html>
<head>
 <script src="https://code.createjs.com/createjs-2013.02.12.min.js"></script>
<script>
 var canvas;
 var stage;
 var line;
 function init(){
   var x1=0;
   var y1=0;
   var x2=0;                                        1
   var y2=0;
 canvas=document.getElementById("myCanvas");                  2
```

```
            stage= new createjs.Stage(canvas);                3
            line = new createjs.Shape();                      4
            stage.addChild(line);                             5
            line.graphics.setStrokeStyle(1).beginStroke("red");      6
            canvas.addEventListener("mousedown",function(e){  7
               x1=e.pageX;
               y1=e.pageY;
               line.graphics.moveTo(x1,y1);
              console.log("down");});
              canvas.addEventListener("mouseup",function(e){  8
               x2=e.pageX;
               y2=e.pageY;
              line.graphics.lineTo(x2,y2);
              console.log("up");
              stage.update(); });
               }
            function clr(){                                   9
            stage.removeChild(line);
             stage.update();
             init();
            }
</script>
         </head>
          <body onload="init()" style="background-color:#D3D3D3;">
            <canvas id="myCanvas" height="250" width="1500" style="background-
            color:white; border:1px solid black;">            10
            </canvas>
```

```
<button id="clear" onclick="clr()">Clear</button>

</body> </html>
```

Output:

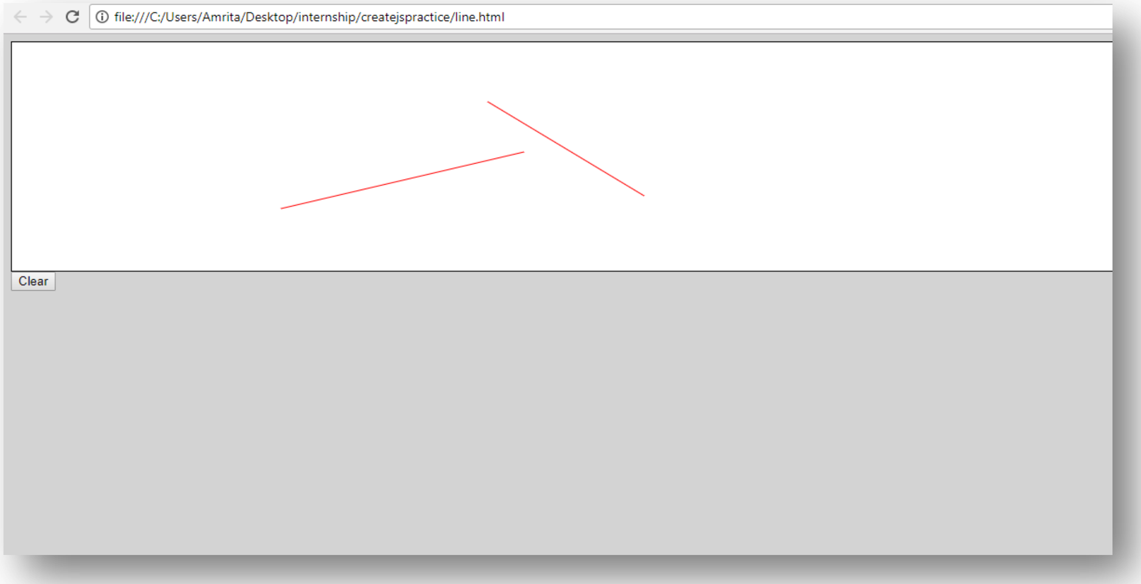

file:///C:/Users/Amrita/Desktop/internship/createjspractice/line.html
Clear

Figure 9.1: Drawing line output

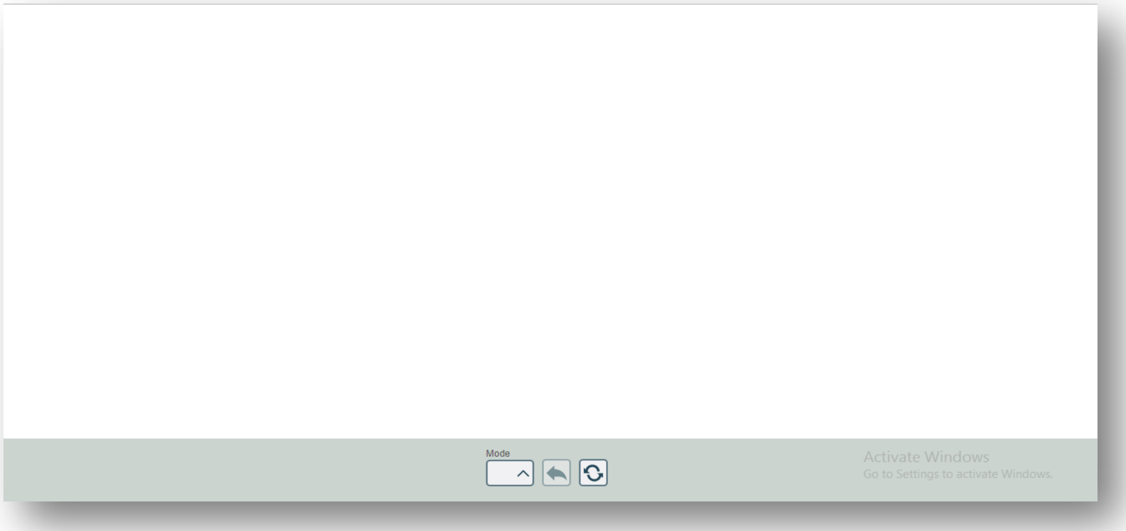
Explanation:

1	To draw a line we need 2 points : (x1,y1) and (x2,y2).Hence this line initializes 4 variables to zero.
2	The canvas element in HTML with id “myCanvas” is returned and assigned to canvas variable which becomes a reference to canvas element
3	This line creates a new stage and points it at our canvas variable in 2.’createjs’ used in the statement is namespace in which Stage class is defined.
4	This creates a new shape instance and assigns it to variable line .Shapes are display objects .

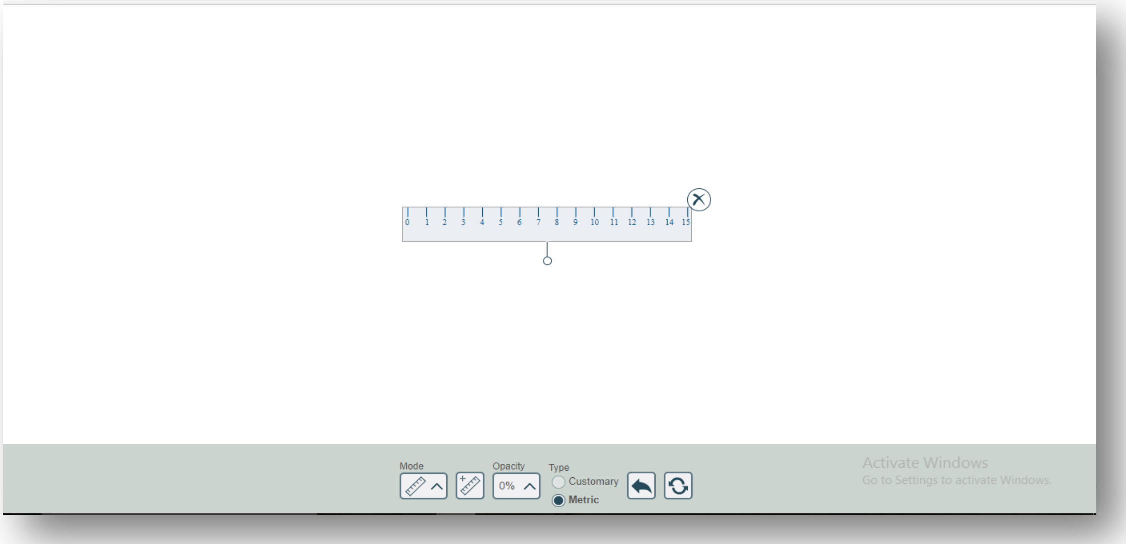
5	Every display object must be added to Stage since it is a container for all display objects. Here addChild method is called upon the stage object and line created in 4 is passed as the parameter.
6	Graphics is shape object's property which points to graphics instance.This instance has all the methods to define vector shapes.Two such methods are beginStroke() for setting the color of object and setStrokeStyle for thickness of the object.
7	An event Listener is defined for mousedown event.At mousedown event, x1 and y1 are recorded using pageX and pageY properties of event. These are then passed as parameters to moveTo method which is used along with lineTo method to draw a line.
8	An event Listener is defined for mouseup event .At this event,x2 and y2 are recorded and then are passed to lineTo method.After this stage.update() is called to reflect the above changes on stage.
9	This function is used to clear the canvas area.Corresponding to this function there is a button named clear which when clicked triggers this function.Here stage's method removeChild is used to remove the line object from stage.
10	This is part of body tag where a canvas element with specified height and width is created and n id is assigned to it.

Chapter 10: Final Results

When no tool is selected :



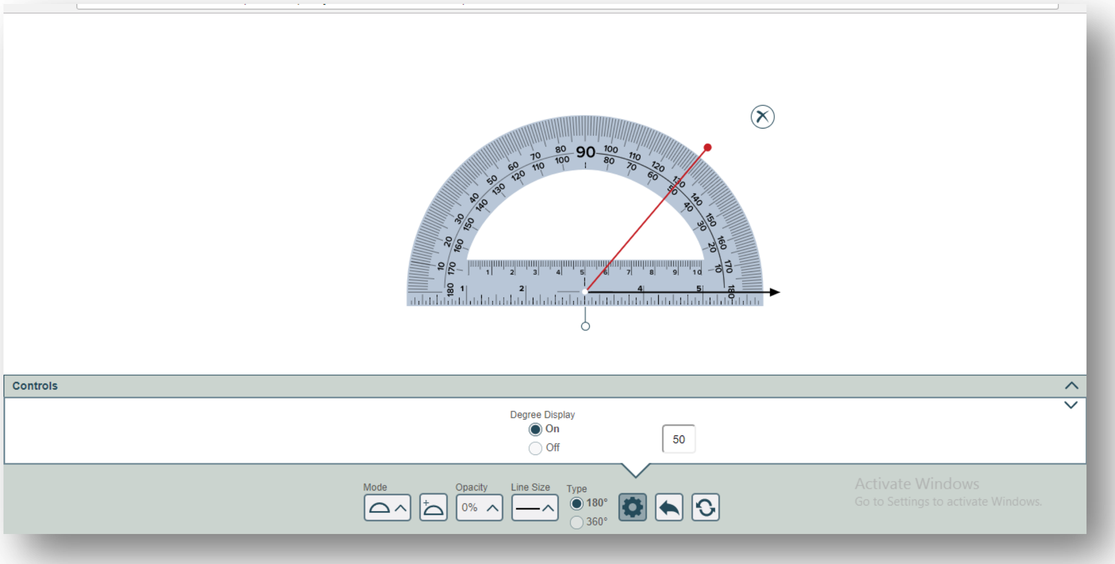
Ruler:



Opacity: 0%

Type: ○ Customary ● Metric

Protractor:



Controls

Degree Display
● On
○ Off

50

Mode

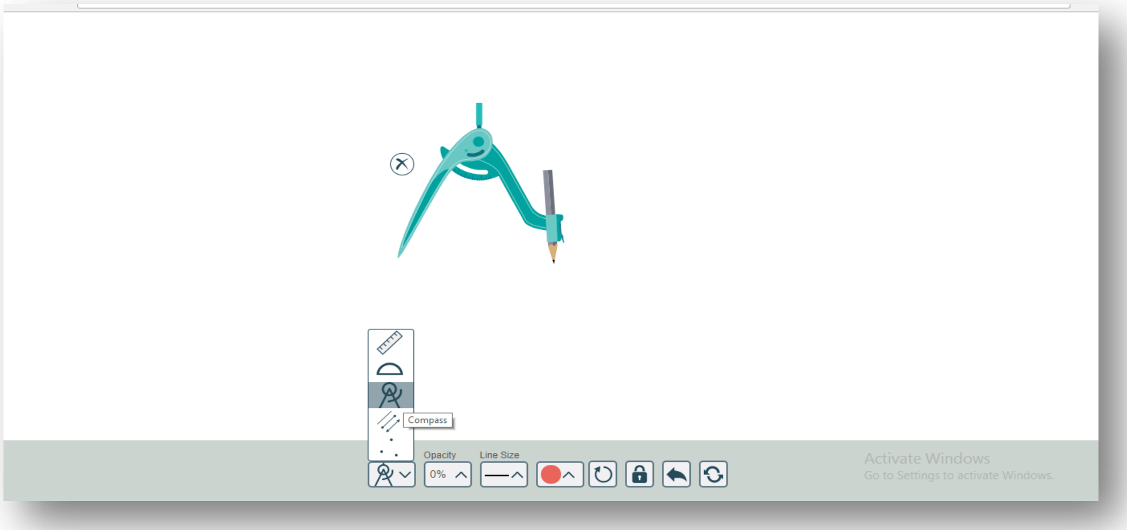
Opacity 0%

Line Size

Type
● 180°
○ 360°

Activate Windows
Go to Settings to activate Windows.

Compass:

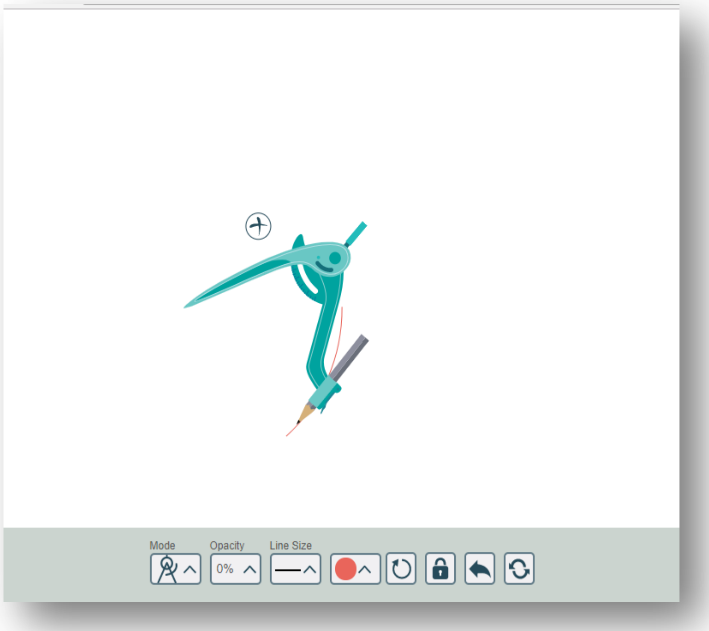


Opacity 0%

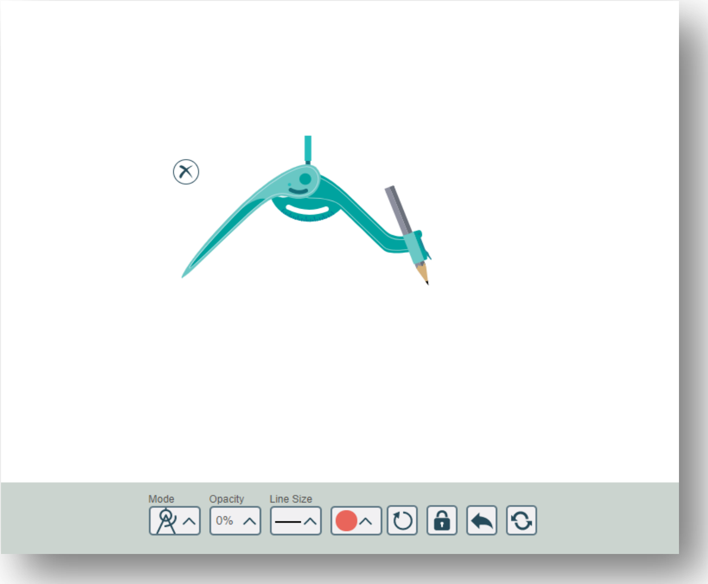
Line Size

Activate Windows
Go to Settings to activate Windows.

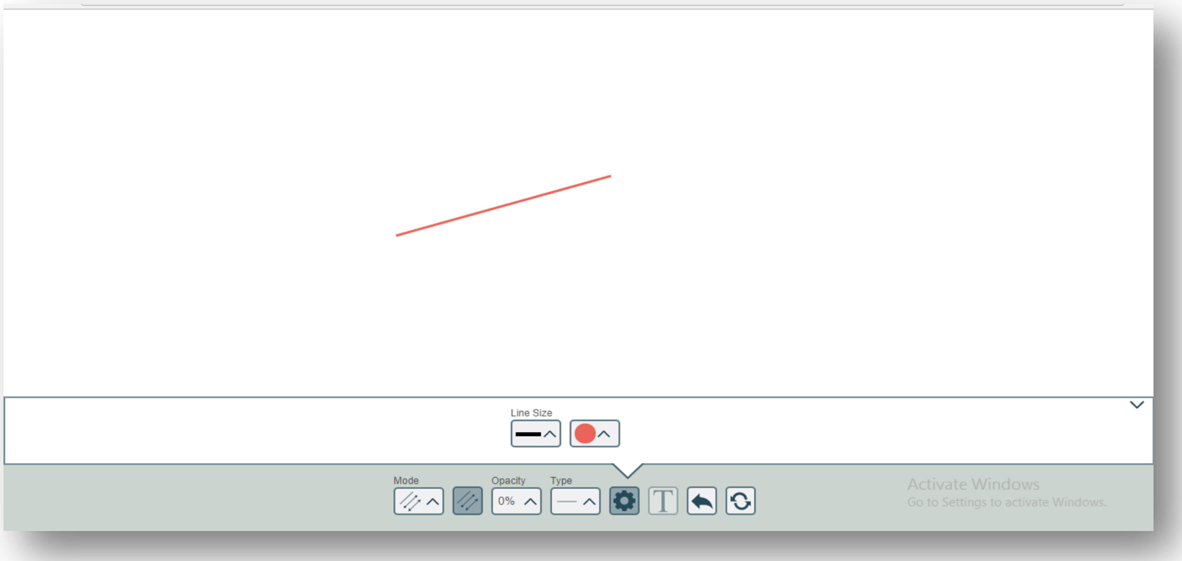
Pencil Dragging :



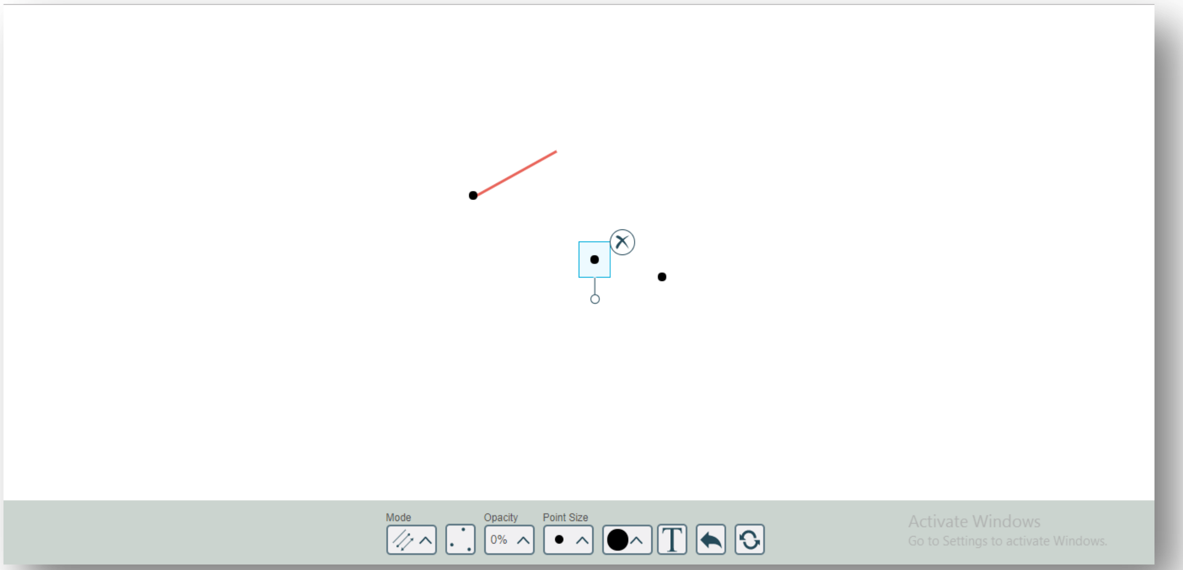
Handle Dragging (changing radius)



Line tool:



Point tool:



Chapter 11: Future Scope

Although this application fulfills the major requirements of the client yet there can be certain features that can be added in future according to the needs of user. With the current code, the user can add protractor, ruler, compass to the canvas. Advancement in features could be seen if users are able to upload their own images or provide URL of the images so that they can be rendered on the canvas and then use the ruler protractor and compass on them.

Another major feature can be added which can save the state of current canvas i.e. what all display objects have been drawn on the canvas and the tools that have been added to the canvas. Upon accessing the widget, once the user starts adding tools, modifying it, current state of tool will be recorded at each point post re-arrangement of tools. If the user accidently closes the window, then the previous state of the canvas can be restored and user can continue working.

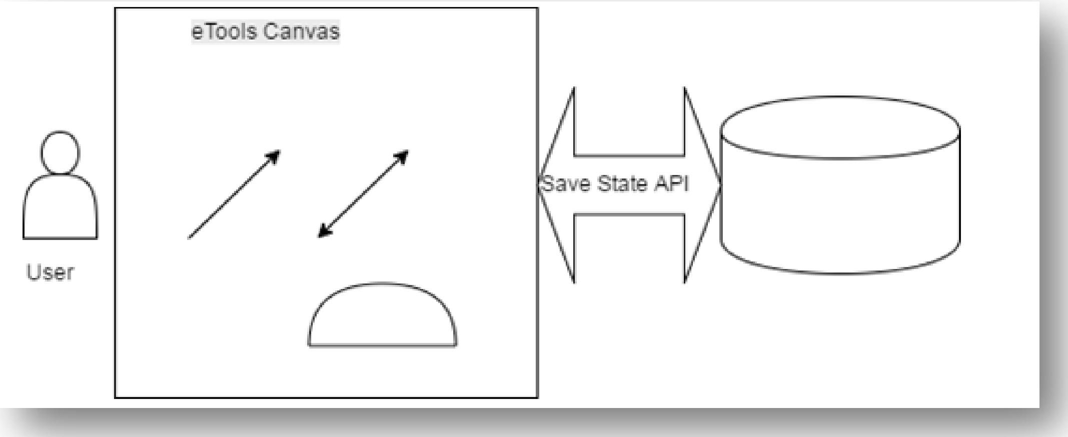


Figure 11.1: Save State

Chapter 12 : Conclusion

This internship program has enhanced my practical knowledge .Here I have learnt how to visualize things in the first place and how all of these things can be implemented with the help of programming. As an intern, the entire project was not assigned to me but one of the most intriguing and challenging module of the etool namely compass was. The compass tool was challenging because the compass movement along with arc creation had to be dealt with mouse events. In this report maximum of the implementation details have been specified pertaining to compass tool .This report also contains a basic study of line tool.

The etool is a simple tool to help students explore and simulate interest in the world of construction and measurement. It just needs a web browser on a computer widget to run and is very easy to use since it requires the use of mouse with which the students are well acquainted.

During the development of the tool I have learnt various new concepts and have got my first exposure to the industrial world. This project/training also required to put all the concepts already known through academic education into action and hence has bridged the gap between the academic institution and corporate world.

Chapter 13: References

https://en.wikipedia.org/wiki/Aptara

http://www.aptaracorp.com/about-aptara/overview

http://createjs.com/

http://www.jsdelivr.com/projects/soundjs

https://code.tutsplus.com/tutorials/using-createjs-easeljs--net-34840

https://jquery.com/

https://jqueryui.com/draggable/

https://www.w3schools.com/jsref/jsref_obj_math.asp

https://www.w3schools.com/css/css3_2dtransforms.asp

https://stackoverflow.com/questions/23031412/kinetic-js-draw-a-quarter-arc

https://developer.mozilla.org/en/docs/Web/CSS/background

http://jsfiddle.net/RwEme/8596/

http://agavestorm.com/kineticjs