5.3 Collaborative Apps 2

Drawing Machine

5.3.1 Introduction / Recap

Drawing Machine

5.3.1 Recap

- Meteor allows us to make creative, collaborative apps in real time

- Both the Client and the Server can insert information into a mongo ‘collection’.

- Ordinary web users can do this interactively, without having to program the database manually

5.3.1 This week

- Last session we looked at how to create a collaborative music application
- This session we are going to look at how to make a collaborative drawing application

5.3.1 This week

- You will learn how to
  - Create a canvas using d3.js
  - Change attributes of the canvas
  - Create client code for inserting data into the database
  - Render database entries as a drawing

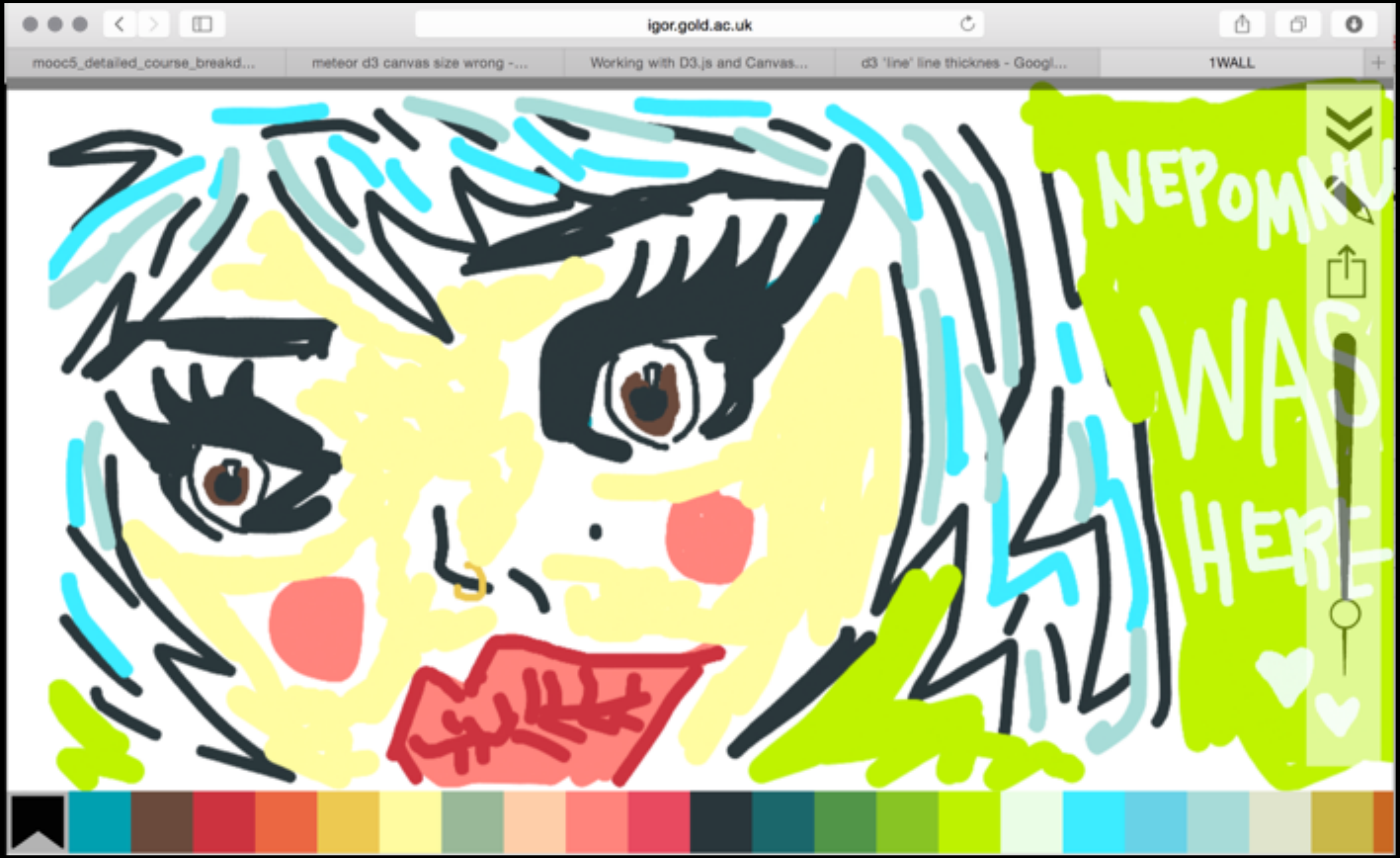
5.3.1

- We will start by looking at a worked example of the a Collaborative drawing application
- 1-WALL was Created by a Goldsmiths Creative Computing student, Jakub Fiala

5.3.2 Worked Example

1-WALL by Jakub Fiala

Jakub Fiala - 1Wall


igor.gold.ac.uk
mooc5_detailed_course_breakd...
meteor d3 canvas size wrong -...
Working with D3.js and Canvas...
d3 'line' line thicknes - Googl...
1WALL
NEPOMNU
WAS
HERE

5.3.3 Creating the Application

Music Machine 2015

5.3.3 Program Structure

- drawingApp.html, drawingApp.js
  - Basic app logic
- drawing.js
  - drawing functions, based on data

5.3.3 SVG canvas

- Scalable Vector Graphics (SVG)
- XML-based vector image format
- Supports interactivity / animation.

```
function Canvas() {
  var self = this;
  var svg;
```

5.3.3 Set up the Canvas

```
var createSvg = function() {
  svg = d3.select('#canvas').append('svg')
    .attr('width', 800) //Fixed width
    .attr(‘height’,600);//Fixed Height
};
createSvg();//call the function
```

5.3.3

- self.clear //clears the screen
- self.draw //grabs points from the database and passes them to an svg object to render
- Points are stored in a single array in a mongoDB collection

5.3.4 Creating and Serving Data

Drawing Machine

5.3.4

- //mongo database.
- points = new Meteor.Collection('pointsCollection');

5.3.4 Creating Points

```
var markPoint = function() {

  var offset = $('#canvas').offset();

      points.insert({

      x: (event.pageX - offset.left),

      y: (event.pageY - offset.top)});

}
```

5.3.4 Canvas Events

```
Template.canvas.events({
  'click': function (event) {
    markPoint();
  },
  'mousedown': function (event) {
    Session.set('draw', true);
  },
  }
```

5.3.4 Canvas Events

```
'mouseup': function (event) {
  Session.set('draw', false);
},
'mousemove': function (event) {
  if (Session.get('draw')) {
    markPoint();// Calls Mark Points Function
  }
```

5.3.5 Review

Drawing Machine