Code Artifacts

(a) **Microsoft Cognitive Services Face API (REST Service)**

-> To create People Face Database

API: **https://api.projectoxford.ai/face/v1.0/facelists/{faceListId}**

```
$(function() {
    var params = {
        // Request parameters
    };

    $.ajax({
        url: "https://api.projectoxford.ai/face/v1.0/facelists/{faceListId}?" + $.param(params),
        beforeSend: function(xhrObj){
            // Request headers
            xhrObj.setRequestHeader("Content-Type","application/json");
            xhrObj.setRequestHeader("Ocp-Apim-Subscription-Key","{subscription key}");
        },
        type: "PUT",
        // Request body
        data: "{body}",
    })
    .done(function(data) {
        alert("success");
    })
    .fail(function() {
        alert("error");
    });
});
```

-> API for Face Detection and find Similar Face from People Group

API: https://api.projectoxford.ai/face/v1.0/findsimilars

```
$(function() {
    var params = {
        // Request parameters
        "returnFaceId": "true",
        "returnFaceLandmarks": "false",
        "returnFaceAttributes": "{string}",
    };

    $.ajax({
        url: "https://api.projectoxford.ai/face/v1.0/detect?" + $.param(params),
        beforeSend: function(xhrObj){
            // Request headers
```

```
            xhrObj.setRequestHeader("Content-Type","application/json");
            xhrObj.setRequestHeader("Ocp-Apim-Subscription-Key","{subscription key}");
        },
        type: "POST",
        // Request body
        data: "{body}",
    })
    .done(function(data) {
        alert("success");
    })
    .fail(function() {
        alert("error");
    });
});
```

(b) **Microsoft Cognitive Services Emotion API (REST Service)**

API: **https://api.projectoxford.ai/emotion/v1.0/recognize**

```
$(function() {
    var params = {
        // Request parameters
    };

    $.ajax({
        url: "https://api.projectoxford.ai/emotion/v1.0/recognize?" + $.param(params),
        beforeSend: function(xhrObj){
            // Request headers
            xhrObj.setRequestHeader("Content-Type","application/json");
            xhrObj.setRequestHeader("Ocp-Apim-Subscription-Key","{subscription key}");
        },
        type: "POST",
        // Request body
        data: "{body}",
    })
    .done(function(data) {
        alert("success");
    })
    .fail(function() {
        alert("error");
    });
```

```
  });
```

(c) **Tessel Camera Module (JavaScript Library)**

```
var tessel = require('tessel');
var camera = require('camera-vc0706').use(tessel.port['A']);

var notificationLED = tessel.led[3]; // Set up an LED to notify when we're taking a picture

// Wait for the camera module to say it's ready
camera.on('ready', function() {
  notificationLED.high();
  // Take the picture
  camera.takePicture(function(err, image) {
    if (err) {
      console.log('error taking image', err);
    } else {
      notificationLED.low();
      // Name the image
      var name = 'picture-' + Math.floor(Date.now()*1000) + '.jpg';
      // Save the image
      console.log('Picture saving as', name, '...');
      process.sendfile(name, image);
      console.log('done.');
      // Turn the camera off to end the script
      camera.disable();
    }
  });
});

camera.on('error', function(err) {
  console.error(err);
});
```

(d) **Web App (Dashboard View) – Python/JavaScript/HTML**

-> Create HTML page using JavaScript and then Host the page on the web server.

```
var express = require('express')
  , logger = require('morgan')
  , app = express()
  , template = require('jade').compileFile(__dirname + '/source/templates/homepage.jade')

app.use(logger('dev'))
app.use(express.static(__dirname + '/static'))

app.get('/', function (req, res, next) {
```

```
  try {
    var html = template({ title: 'Home' })
    res.send(html)
  } catch (e) {
    next(e)
  }
})

app.listen(process.env.PORT || 3000, function () {
  console.log('Listening on http://localhost:' + (process.env.PORT || 3000))
})
```

(e) **Sending SMS (Amazon SNS)**

```
var sns = new AWS.SNS({apiVersion: '2010-03-31'});

var sns = new AWS.SNS();
sns.addPermission(params, function (err, data) {
  if (err) console.log(err, err.stack); // an error occurred
  else     console.log(data);           // successful response
});
```

(f) **Web Server (Django or Node.js)**

Simply, install the web framework (most probably NodeJS) on Amazon EC2.

(g) **DBMS (MySQL Server)**

Create SQL Tables and connect the same to the web server.

(h) **AWS Analytics or Google Analytics API**

As discussed in class**,** accordingly use the best API to predict the user's needs at time t (based on historical data and current emotional state).