ISIT 343 Major Assignment

By Jingwang Teh

Last Name :	Jingwang
Family Name :	Teh
Lab Code :	35
Student Number :	5238699

Task 1A

Debugging Features Used

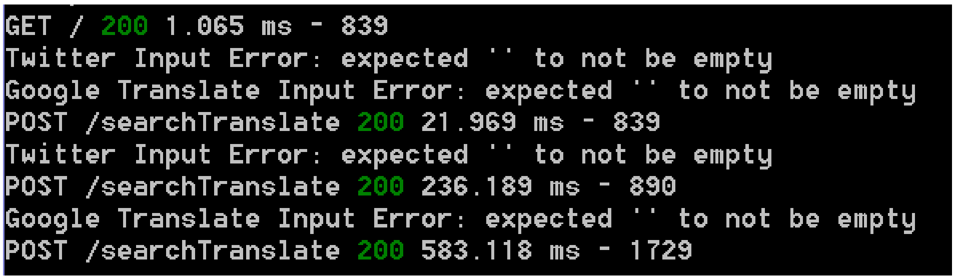
- (#1) Functions from Expect.js
- (#2) Mocha Framework

Source Code #1 (/Controller/searchTranslate.js)

```
var twitterSearch = require('./twitterSearch')
        , googleTranslate = require('./googleTranslate')
        , expect = require('expect.js')
        , async = require('../node_modules/async');

async.parallel([
        // twitter search
        function(callback) {
                try {
                        expect(req.body).to.have.property('search_term');

                        var query = req.body.search_term.trim();
                        expect(query).to.not.be.empty();

                        twitterSearch(query, function (err, tweets){
                                if (err) { console.log(err.message); }
                                else { callback(null, tweets.statuses); }
                        });
                } catch (err) {
                        console.log('Twitter Input Error: '+err.message);
                        callback(null, '');
                }
        },
        // google translate
        function(callback) {
                try {
                        expect(req.body).to.have.property('source_text');
                        expect(req.body).to.have.property('target_lang');

                        var query = req.body.source_text.trim()
                        ,       targetLang = req.body.target_lang;
                        expect(query).to.not.be.empty();
                        expect(targetLang).to.not.be.empty();

                        googleTranslate(query, targetLang, function (err, translatedText){
                                if (err) { console.log(err.message); }
                                else { callback(null, translatedText); }
                        });
                } catch (err) {
                        console.log('Google Translate Input Error: '+err.message);
                        callback(null, '');
                }
        }
], function(err, results) {
        if (err) console.error(err);
        res.render('index', { twitterResult : results[0], translateResult : results[1] });
});
```

Brief Outputs #1

Expect.js functions to “expect” input is not empty

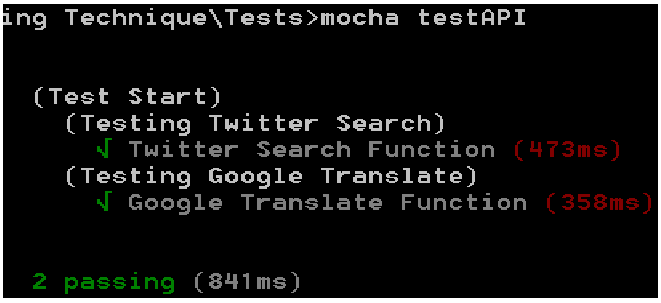


```
GET / 200 1.065 ms - 839
Twitter Input Error: expected '' to not be empty
Google Translate Input Error: expected '' to not be empty
POST /searchTranslate 200 21.969 ms - 839
Twitter Input Error: expected '' to not be empty
POST /searchTranslate 200 236.189 ms - 890
Google Translate Input Error: expected '' to not be empty
POST /searchTranslate 200 583.118 ms - 1729
```

Source Code #2 (/Tests/testAPI.js)

```
var expect = require('expect.js')
        , search = require('../Controller/twitterSearch')
        , translate = require('../Controller/googleTranslate');

describe('(Test Start) ', function () {

describe('(Testing Twitter Search) ', function () {
it('Twitter Search Function', function (done) {
        var query = 'University of Wollongong';
        search(query, function (err, tweets){
        try {
                expect(err).to.not.be.ok();   // should be null
                expect(tweets).to.be.ok();   // should exists
                expect(tweets).to.be.an('object'); // tweets is an object
                expect(tweets.statuses).to.be.an('array'); // tweets.statuses is an array inside tweets object
        } catch (error) {     console.log('\033[31m\n'+error.message+'\033[39m');  }
                done();
        });
}); });

describe('(Testing Google Translate) ', function () {
it('Google Translate Function', done => {
        var query = '苹果'
        ,       targetLang = 'en';
        translate(query, targetLang, function (err, translatedText){
        try {
                expect(err).to.not.be.ok(); // should be null
                expect(translatedText).to.be.ok(); // should exists
        } catch (error){ console.log('\033[31m\n'+error.message+'\033[39m'); }
                done();
        });
}); });

});
```

Brief Output #2



```
ing Technique\Tests>mocha testAPI


  (Test Start)
    (Testing Twitter Search)
      √ Twitter Search Function (473ms)
    (Testing Google Translate)
      √ Google Translate Function (358ms)


  2 passing (841ms)
```

Errors encountered

I have encountered errors from a response from Google Translate Client due to unhandled JavaScript promises, and after learning more about promises, I managed to handle the error accordingly. When I applied Mocha Testing Framework, there was also an error regarding the method and place done() should be applied.

Understanding of Debugging Techniques

My understanding is that several debugging techniques allow a conditional comparison to verify a variable's condition, which will trigger an Exception if condition was not met, and will be required to be manually handled through try-catch block. There are also several Test Frameworks that can be used to check coverage and variable values to ensure the program does not crash unexpectedly. In addition, there are also ways to handle execution flow and set breakpoints, similar to most compiler programs.

Task 1B

Source Code #1 Angular.JS (/View/index.ejs)

```
<div ng-mouseup="getSelectionText()">
<% if(locals.twitterResult) { %>
        <% if (Array.isArray(locals.twitterResult)) { %>
                <h3>Twitter Search Result</h3>
                <% twitterResult.forEach(function (tweet) { %>
                        <p><%= tweet.text %></p>
                        <b><%= tweet.user.name %></b>
                <% }); %>
        <% } else { %>
                <%= twitterResult %>
        <% } %>
<% } %>
</div>
<div>
<% if(locals.translateResult) { %>
        <h3>Google Translate Result</h3>
        <p><%= translateResult %></p>
<% } %>
</div>

<script>
var app = angular.module('myApp', []);
app.controller('myCtrl', function ($scope) {
        $scope.selectedText = '';
        $scope.getSelectionText = function () {
                var text = "";
                if (window.getSelection) {
                        text = window.getSelection().toString();
                } else if (document.selection && document.selection.type != "Control") {
                        text = document.selection.createRange().text;
                }
                $scope.selectedText = text;
        };
});
</script>
```

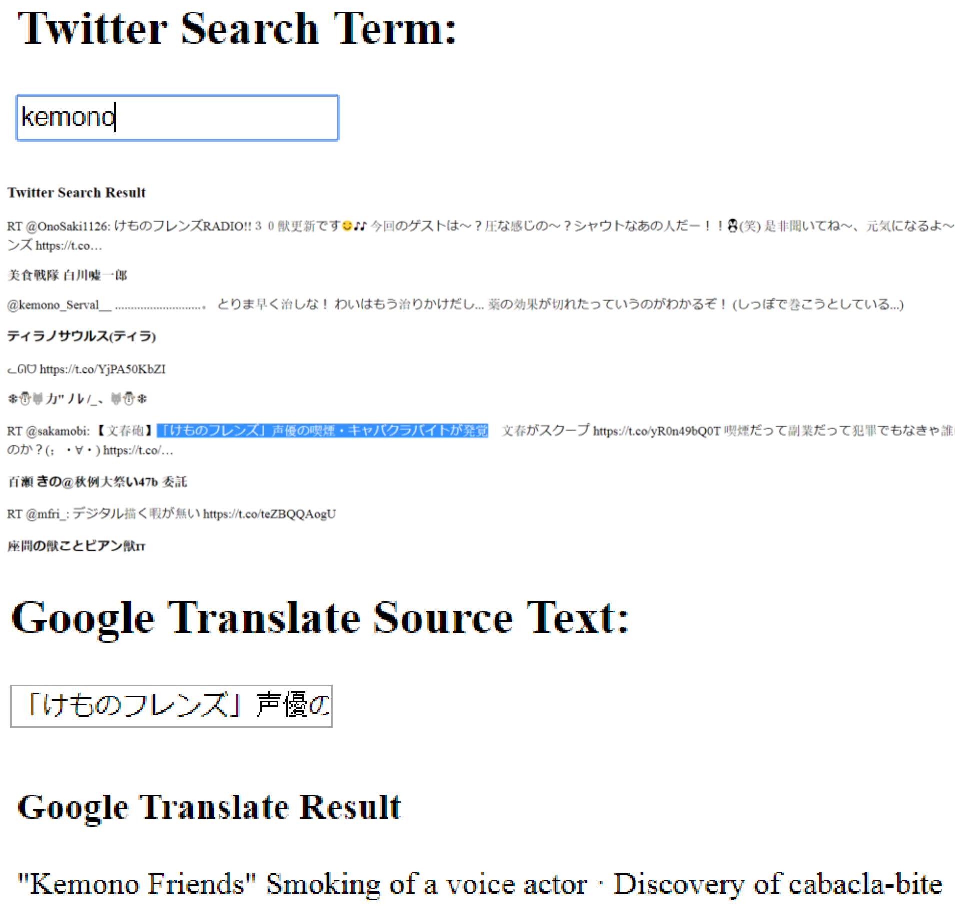
Source Code #2 Twitter Search (/Controller/twitterSearch.js)

```
var request = require('superagent')
        , TwitterClient = require('../Model/twitterClient');

module.exports = function twitterSearch (query, fn) {
TwitterClient.get(
        'search/tweets',
        {q: query, count:5},
        function (err, tweets, response) {
                if (tweets && Array.isArray(tweets.statuses)) {
                        return fn(null, tweets);
                } else { fn(new Error('Bad Twitter Response')); }
}       );
};
```

Source Code #3 Google Translate (/Controller/twitterSearch.js)

```
var request = require('superagent')
		, translate = require('../node_modules/google-translate-api');

module.exports = function googleTranslate (query, targetLang, fn) {
		translate(query, {to: targetLang}).then(res => {
				return fn(null, res.text);
		}).catch(err => { fn(new Error('Bad Google Translate Response')); });
};
```

Brief Output



Twitter Search Term:

kemono

Twitter Search Result

RT @OnoSaki1126: けものフレンズRADIO!! 3 0 獣更新です😊♪♪ 今回のゲストは～？圧な感じの～？シャウトなあの人だー！！👻(笑) 是非聞いてね～、元気になるよ～😂！！ https://t.co/Ftvgn1m9GB #けものフレンズ https://t.co…

美食戦隊 白川嘘一郎

@kemono_Serval__ ....。 とりま早く治しな！ わいはもう治りかけだし... 薬の効果が切れたっていうのがわかるぞ！ (しっぽで巻こうとしている...)

ティラノサウルス(ティラ)

ت_ᘏᗢ https://t.co/YjPA50KbZI

❄☃🐺カ" ノレ/_、🐺☃❄

RT @sakamobi: 【文春砲】「けものフレンズ」声優の喫煙・キャバクラバイトが発覚　文春がスクープ https://t.co/yR0n49bQ0T 喫煙だって副業だって犯罪でもなきゃ誰にも迷惑かけてない。なんで問題視されるのか？(；　・∀・) https://t.co/…

百瀬 きの@秋例大祭い47b 委託

RT @mfri_: デジタル描く暇が無い https://t.co/teZBQQAogU

座間の獣ことピアン獣rr

Google Translate Source Text:

「けものフレンズ」声優の

Google Translate Result

"Kemono Friends" Smoking of a voice actor · Discovery of cabacla-bite

Errors Encountered

An error that I encountered was when dealing with mouse events that is appropriate for mouse selection, where I end up using an AngularJS event, which is easier to use. I also had issue with setting a function to be accessed by said mouse event, in which I solved it by assigning the function to a scope variable, which is then accessed by the event.

Task 2A

2 Twitter Features: Count and Language

Source Code #1 (/Controller/search.js)

```
var twitterSearch = require('./twitterSearch');

exports.home = function (req, res, next) {
        if (!req.body || req.body.search_term == undefined || req.body.search_term.trim() == '') {
                return res.redirect('/');
        }

        var query = req.body.search_term.trim();
        var count = 5;
        var lang = "en";
        if (req.body.search_count != undefined) {
                count = req.body.search_count;
        }
        if (req.body.search_lang != undefined) {
                lang = req.body.search_lang;
        }

        twitterSearch(query, count, lang, function (err, tweets) {
                if (err) { res.render('index', { result : null}); }
                else { res.render('index', { result : tweets.statuses }); }
        });
}
```

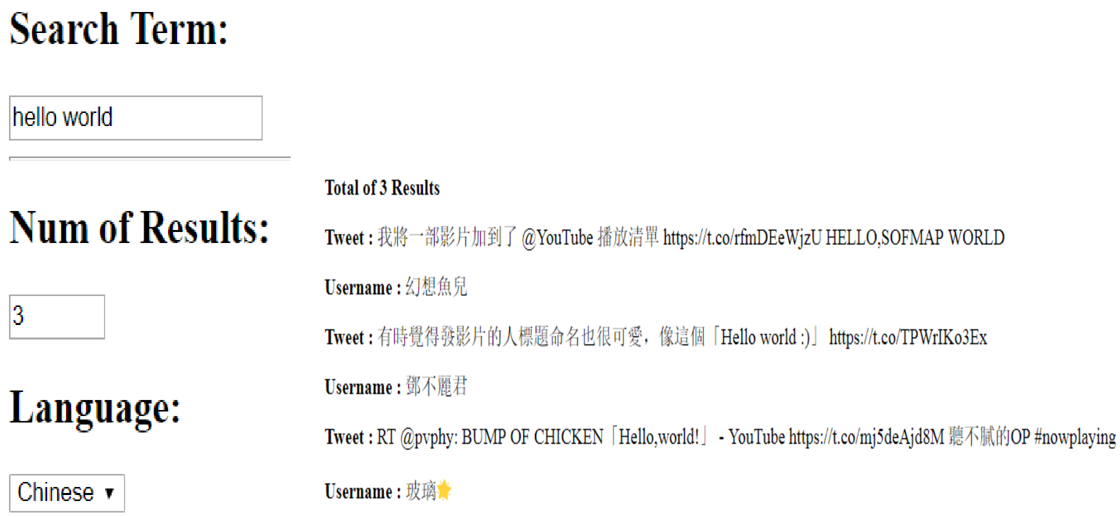
Source Code #2 (/Controller/twitterSearch.js)

```
var request = require('superagent')
        , TwitterClient = require('../Model/twitterClient');

module.exports = function twitterSearch (query, search_count, search_lang, fn) {
TwitterClient.get(
        'search/tweets',
        {q: query, count: search_count, lang: search_lang},
        function (err, tweets, response) {
                if (tweets && Array.isArray(tweets.statuses)) {
                        return fn(null, tweets);
                } else { fn(new Error('Bad Twitter Response')); }
        }
);      };
```

Source Code #3 (/Model/twitterClient.js)

```
var Twitter = require('../node_modules/twitter');

module.exports = new Twitter({
        consumer_key: ‘abcdef’, // false keys used for documentation/report
        consumer_secret: 'abcdef',
        access_token_key: 'abcdef',
        access_token_secret: 'abcdef'
});
```

Brief Output



Search Term:

hello world

Num of Results:

3

Language:

Chinese ▾

Total of 3 Results

Tweet : 我將一部影片加到了 @YouTube 播放清單 https://t.co/rfmDEeWjzU HELLO,SOFMAP WORLD

Username : 幻想魚兒

Tweet : 有時覺得發影片的人標題命名也很可愛，像這個「Hello world :)」 https://t.co/TPWrIKo3Ex

Username : 鄧不麗君

Tweet : RT @pvphy: BUMP OF CHICKEN「Hello,world!」 - YouTube https://t.co/mj5deAjd8M 聽不膩的OP #nowplaying

Username : 玻璃🌟

Errors Encountered

There was an error on textbook API access as the API was no longer active, which I solved by using npm Twitter package and checking the updated API documentation. Aside from that, there were sometimes fewer to no results as either the search term was empty (before adding trim()), and the search term simply was too random for there to be much result.

Task 2B

Features Added: Damage/Heal Score by clicking on mouse images, Broadcast announcement

Source Code #1 (/public/index.html)

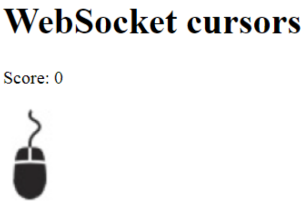
```
document.onclick = function (ev) {
	if (!lost) { ws.send(JSON.stringify({ type: 'clickDmg', x: ev.clientX, y: ev.clientY })); }
	else { ws.send(JSON.stringify({ type: 'clickHeal', x: ev.clientX, y: ev.clientY })); }
}

var initialized;
ws.onmessage = function (ev) {
	var obj = JSON.parse(ev.data);
	// the first message is the position of all existing cursors
	if (!initialized) {
		initialized = true;
		for (var id in obj) {
			move(id, obj[id]);
		}
	} else {
		if ('disconnect' == obj.type) {
			remove(obj.id);
		} else if ('position' == obj.type) {
			move(obj.id, obj.pos);
		} else if ('clickDmg' == obj.type) {
			if (!lost) {
				updateScore(obj.pos);
			}
		} else if ('clickHeal' == obj.type) {
			if (!lost) {
				healScore(obj.pos);
			}
		} else if ('deathAnnounce' == obj.type) {
			setAnnouncement(obj.pos);
		}
	}
}

function updateScore (pos) {
if ( (pos.x < (curX + 48)) && (pos.x > (curX - 5)) && (pos.y < (curY + 94)) && (pos.y > (curY - 5)) ) {
	--score;

	if (score <= -20) {
		lost = true;
		scoreElement.innerHTML = score + ' (you lost)';
		ws.send(JSON.stringify({ type: 'lost', x: curX, y: curY }));
	} else if (!lost) {
		scoreElement.innerHTML = score;
	}
}	}

function healScore (pos) {
if ( (pos.x < (curX + 48)) && (pos.x > (curX - 5)) && (pos.y < (curY + 94)) && (pos.y > (curY - 5)) ) {
	++score;
	scoreElement.innerHTML = score;
}	}

function setAnnouncement (pos) {
if ( (pos.x < (curX + 148)) && (pos.x > (curX - 105)) && (pos.y < (curY + 194)) && (pos.y > (curY - 105)) ) {
	messageElement.innerHTML = 'someone was pwned in area';
	setTimeout(function(){ messageElement.innerHTML = '' }, 2000);
}	}
```

Source Code #2 (/public/index.html)

```
socket.on('message', function (msg) {
        try { var obj = JSON.parse(msg); }
        catch (e) { return; }

        if (obj.type == 'position') {
                pos = { x: obj.x, y: obj.y };
                positions[socket.id] = pos;
                broadcast(JSON.stringify({ type: 'position', pos: pos, id: socket.id }));
        } else if (obj.type == 'clickDmg'){
                pos = { x: obj.x, y: obj.y };
                broadcast(JSON.stringify({ type: 'clickDmg', pos: pos, id: socket.id }));
        } else if (obj.type == 'clickHeal') {
                pos = { x: obj.x, y: obj.y };
                broadcast(JSON.stringify({ type: 'clickHeal', pos: pos, id: socket.id }));
        } else if (obj.type == 'lost') {
                delete positions[socket.id];
                broadcast(JSON.stringify({ type: 'disconnect', id: socket.id }));
                pos = { x: obj.x, y: obj.y };
                broadcast(JSON.stringify({ type: 'deathAnnounce', pos: pos }));
        }
});
```

Brief Output

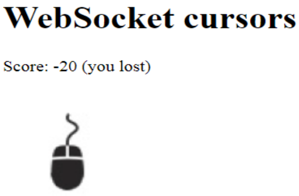
Default Score:



WebSocket cursors

Score: 0

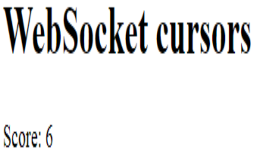
Damage Score:



WebSocket cursors

Score: -20 (you lost)

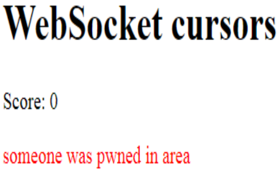
Heal Score:



WebSocket cursors

Score: 6

Announcement:



WebSocket cursors

Score: 0

someone was pwned in area

Errors Encountered

An error encountered was when trying to send 2 parameters with ws.send(), wherein the 2nd parameter was not received, therefore was null. It was solved by passing both parameters as a single JSON object.

Task 3A

Source Code #1 (server.js)

```
socket.on('song', function (song) {
        if (socket.dj) {
                currentSong = song;
                socket.broadcast.emit('song', song);
        }
});

socket.on('search', function (q, fn) {
        if (socket.dj) {
                var searchQuery = q.trim();
                spotify.search({ type: 'track', query: searchQuery }, function(err, data) {
                        if (err) { return console.log('Error occurred: ' + err); }
                        currentSongList = data.tracks.items;
                        fn(data.tracks.items);
                        socket.broadcast.emit('displaySearch', data.tracks.items);
                });
        } else {
                if (currentSongList != undefined) {
                        fn(currentSongList);
                }
        }
});
```

Source Code #2 (/public/chat.js)

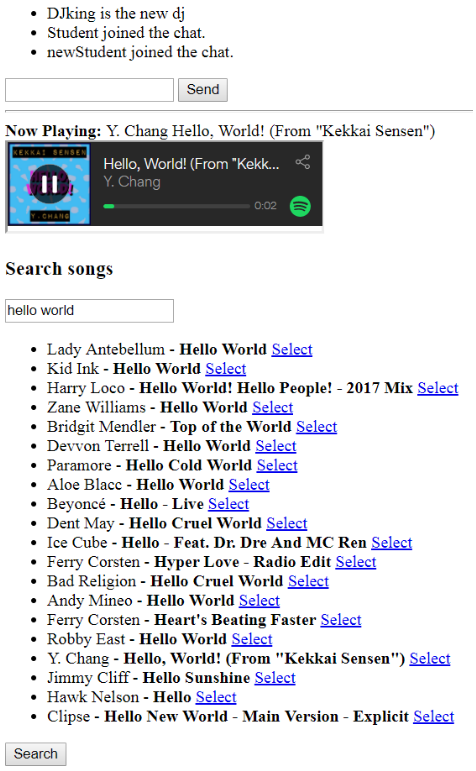
```
socket.emit('search', document.getElementById('s').value, function (songs) { displaySongs(songs); });

function displaySongs (songs) {
while (results.hasChildNodes()) {
        results.removeChild(results.lastChild);
}
for (var i = 0, l = songs.length; i < l; i++) {
        (function (song) {
                var result = document.createElement('li');
                result.innerHTML = song.artists[0].name + ' - <b>' + song.name + '</b> ';
                var a = document.createElement('a');
                a.href = '#';
                a.innerHTML = 'Select';
                // play current song
                a.onclick = function () {
                        socket.emit('song', song);
                        play(song);
                        return false;
                }
                result.appendChild(a);
                results.appendChild(result);
        })(songs[i]);
}       }

// plays a song
var playing = document.getElementById('playing');
function play (song) {
        if (!song) return;
        //song
        playing.innerHTML = '<hr><b>Now Playing: </b>' + song.artists[0].name + ' ' + song.name + '<br/>';
        var iframe = document.createElement('iframe');
        iframe.frameborder = 0;
        iframe.src = 'https://open.spotify.com/embed?uri='+song.uri;
        playing.appendChild(iframe);
};
```

Brief Output

DJ (Left), Normal User (Right)

DJ can search, and broadcast song to everyone.

User sees DJ's search result as well, and can select the songs from the list



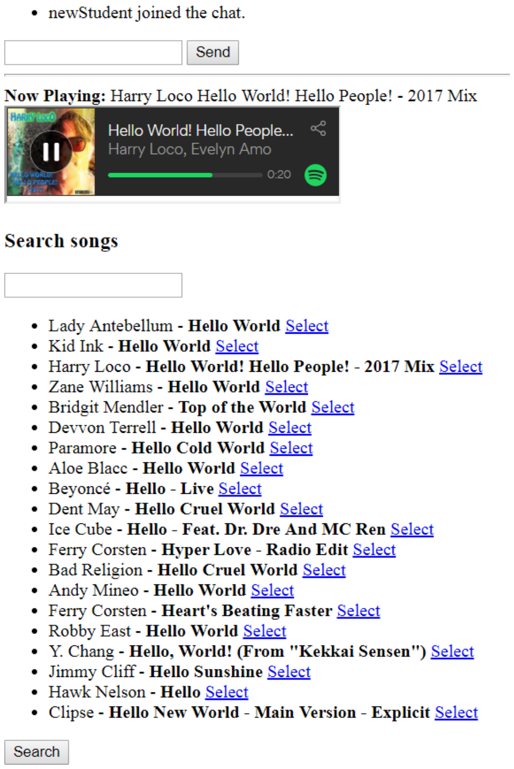
- DJking is the new dj
- Student joined the chat.
- newStudent joined the chat.

Now Playing: Y. Chang Hello, World! (From "Kekkai Sensen")

Search songs

hello world

- Lady Antebellum - **Hello World** Select
- Kid Ink - **Hello World** Select
- Harry Loco - **Hello World! Hello People!** - **2017 Mix** Select
- Zane Williams - **Hello World** Select
- Bridgit Mendler - **Top of the World** Select
- Devvon Terrell - **Hello World** Select
- Paramore - **Hello Cold World** Select
- Aloe Blacc - **Hello World** Select
- Beyoncé - **Hello** - **Live** Select
- Dent May - **Hello Cruel World** Select
- Ice Cube - **Hello** - **Feat. Dr. Dre And MC Ren** Select
- Ferry Corsten - **Hyper Love** - **Radio Edit** Select
- Bad Religion - **Hello Cruel World** Select
- Andy Mineo - **Hello World** Select
- Ferry Corsten - **Heart's Beating Faster** Select
- Robby East - **Hello World** Select
- Y. Chang - **Hello, World! (From "Kekkai Sensen")** Select
- Jimmy Cliff - **Hello Sunshine** Select
- Hawk Nelson - **Hello** Select
- Clipse - **Hello New World** - **Main Version** - **Explicit** Select



- newStudent joined the chat.

Now Playing: Harry Loco Hello World! Hello People! - 2017 Mix

Search songs

- Lady Antebellum - **Hello World** Select
- Kid Ink - **Hello World** Select
- Harry Loco - **Hello World! Hello People!** - **2017 Mix** Select
- Zane Williams - **Hello World** Select
- Bridgit Mendler - **Top of the World** Select
- Devvon Terrell - **Hello World** Select
- Paramore - **Hello Cold World** Select
- Aloe Blacc - **Hello World** Select
- Beyoncé - **Hello** - **Live** Select
- Dent May - **Hello Cruel World** Select
- Ice Cube - **Hello** - **Feat. Dr. Dre And MC Ren** Select
- Ferry Corsten - **Hyper Love** - **Radio Edit** Select
- Bad Religion - **Hello Cruel World** Select
- Andy Mineo - **Hello World** Select
- Ferry Corsten - **Heart's Beating Faster** Select
- Robby East - **Hello World** Select
- Y. Chang - **Hello, World! (From "Kekkai Sensen")** Select
- Jimmy Cliff - **Hello Sunshine** Select
- Hawk Nelson - **Hello** Select
- Clipse - **Hello New World** - **Main Version** - **Explicit** Select

Errors Encountered

An error that I encountered was when appending search results to results container, in that it does not overwrite existing results but kept extending more, which was solved by removing all child nodes first, then appending the respective search results.

Task 3B

Source Code #1 (/Controller/search.js)

```
var googleMapsSearch = require('./googleMapsSearch');

exports.home = function (req, res, next) {
        if (!req.body || req.body.search_term.trim() == '') {
                return res.redirect('/');
        }

        var query = req.body.search_term.trim();
        googleMapsSearch(query, function (err, response){
                if (err) { res.render('index', { result : null}); }
                else {
                        var response = response[0];
                        var latitude = response.geometry.location.lat;
                        var longitude = response.geometry.location.lng;
                        res.render('index', { result : [latitude, longitude] });
                }
        });
}
```

Source Code #2 (/Controller/googleMapsSearch.js)

```
var request = require('superagent')
        , GoogleMapsClient = require('../Model/googleMapsClient');

module.exports = function googleMapsSearch (query, fn) {
GoogleMapsClient.geocode(
        { address: query },
        function (err, response)
        {
                if (!err) { fn(null, response.json.results); }
                else { fn(new Error('Bad GoogleMaps Response : ' + err.json.error_message)); }
        }
);      };
```

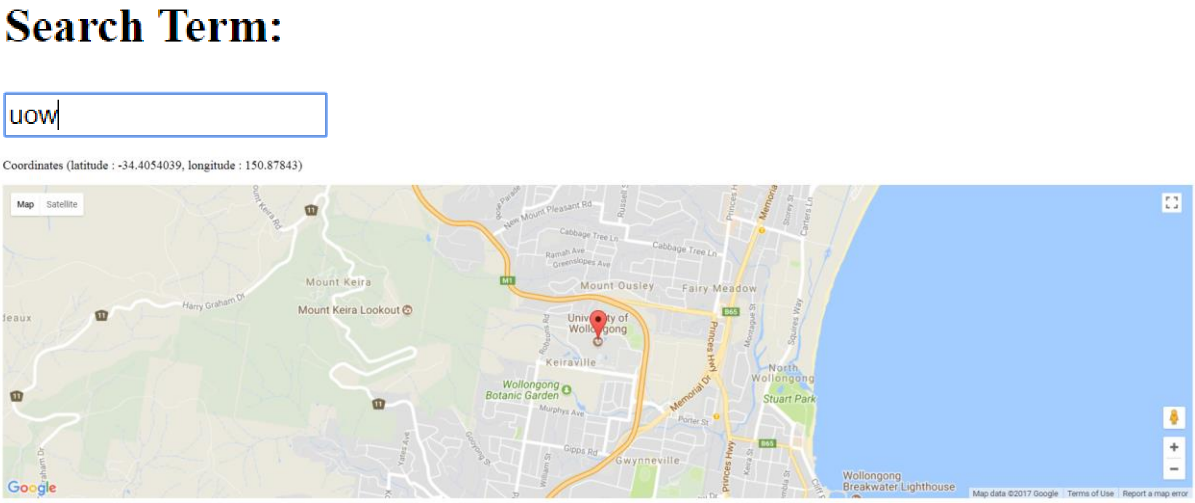
Source Code #3 (/Model/googleMapsClient.js)

```
var GoogleMaps = require('../node_modules/@google/maps');
module.exports = GoogleMaps.createClient({
        key: 'abcdef' // false keys used for documentation/report
});
```

Source Code #4 (/Model/googleMapsClient.js)

```
<div id="map"></div>
 <script>
<% if(result) { %>
        function initMap() {
                var targetLoc = {lat: <%= result[0] %>, lng: <%= result[1] %>};
                var map = new google.maps.Map(document.getElementById('map'), {
                        zoom: 14,
                        center: targetLoc
                });
                var marker = new google.maps.Marker({
                        position: targetLoc,
                        map: map
                });
        }
<% } %>
</script>
<script async defer src="https://maps.googleapis.com/maps/api/js?key=abcdef&callback=initMap"></script>
```

Brief Output



Search Term:

uow

Coordinates (latitude : -34.4054039, longitude : 150.87843)

Errors Encountered

An error that I encountered was not seeing the map output even though the result was provided, which I solved by setting the appropriate dimensions for the map, as it does not set a default dimension.

Description of Functionality

The application allows user to provide an address, in which the application will find the corresponding geocode (latitude, longitude), and use the geocode to retrieve an appropriate map from Google Maps API, to be shown in the webpage.

Task 4A

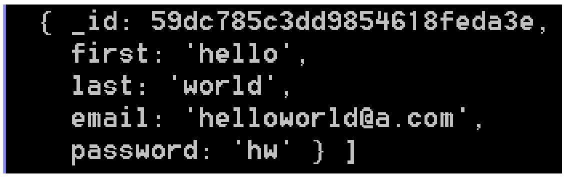
2 Additional Features: Change Password and Find User by first/last name

Source Code (server.js)

```
server.post('/changepw', function (req, res, next) {
        if (req.body == undefined || req.body['user_password'] == undefined) {
                res.redirect('/');
        } else if (req.body['user_password'].trim() == '') {
                res.redirect('/');
        } else {
                // update db with new password
                var newPassword = req.body['user_password'].trim();
                server.users.update({ _id: mongodb.ObjectId(req.session.loggedIn) },
                                        { $set: { password: newPassword } });
                // log user out after changing password
                req.session.loggedIn = null;
                res.redirect('/');
        }
});

server.post('/search', function (req, res, next) {
        if (req.body == undefined || req.body['search_term'] == undefined) {
                res.redirect('/');
        } else if (req.body['search_term'].trim() == '') {
                res.redirect('/');
        } else {
                // search user by first/last name
                var searchVal = req.body['search_term'].trim();
                server.users.find({ $or: [ { first: { $regex: searchVal } },
                                            { last: { $regex: searchVal  } } ]}).toArray(function(err, result) {
                        if (err) return next(err);
                        res.locals.userlist = result;
                        res.render('index');
                });
        }
});
```

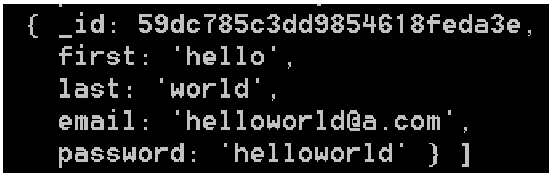
Brief Output (Change Password)

Before Changing Password:



```
{ _id: 59dc785c3dd9854618feda3e,
  first: 'hello',
  last: 'world',
  email: 'helloworld@a.com',
  password: 'hw' } ]
```

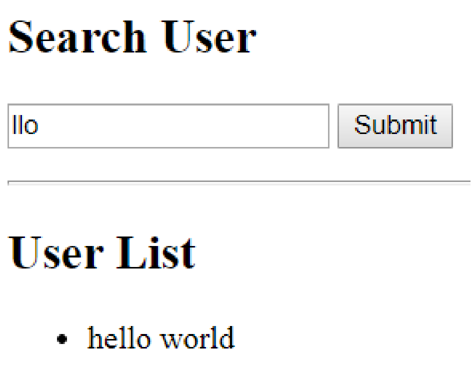
After Changing Password:



```
{ _id: 59dc785c3dd9854618feda3e,
  first: 'hello',
  last: 'world',
  email: 'helloworld@a.com',
  password: 'helloworld' } ]
```

Brief Output (Find User by first/last name)

Search by first name:



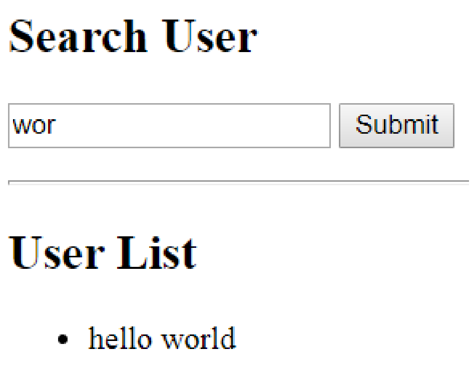
Search User

llo

User List

- hello world

Search by last name:



Search User

wor

User List

- hello world

Errors Encountered

An error was encountered when finding documents by id with a id string, in which I solve it by passing the id string to mongodb.Object() function, then searching with an id object instead of an id string. Another error was when updating a document, which ended up replacing all fields in record with the new property. It was solved by using $set so that only the selected property was replaced.

Task 4B

Application for users to add and delete their own comments

Source Code #1 (/Model/comments.js)

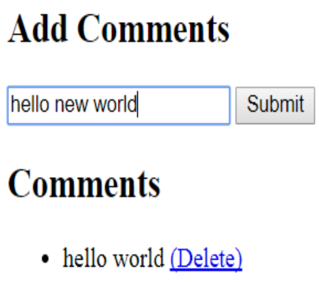
```
var mongoose = require('../node_modules/mongoose')
        , Schema = mongoose.Schema
        , ObjectId = Schema.ObjectId;

var commentSchema = new Schema({
        comment : String,
        userId : { type: Schema.Types.ObjectId, index: true }
});
module.exports = mongoose.model('userComments', commentSchema);
```

Source Code #2 (server.js)

```
server.post('/comments', function (req, res, next) {
if (req.body == undefined || req.body['user_comment'] == undefined) {
        res.redirect('/');
} else if (req.body['user_comment'].trim() == '') {
        res.redirect('/');
} else {
        var comment = {
                'comment' : req.body['user_comment'].trim(),
                'userId' : mongodb.ObjectId(req.session.loggedIn)
        };
        // add comment to database
        new comments(comment).save(function (err, doc) {
                if (err) return next(err);
                comments.find({ userId: mongodb.ObjectId(req.session.loggedIn) }, function (err, result) {
                        if (err) return next(err);
                        res.locals.userCommentlist = result;
                        res.render('index');
                });
        });
}       });

server.get('/delete/:commentId', function (req, res, next) {
// remove comment
try {
        comments.find({ _id: mongodb.ObjectId(req.params.commentId),
                       userId: mongodb.ObjectId(req.session.loggedIn) })
        .remove(function (err, result) {
                if (err) return next(err);
                comments.find({ userId: mongodb.ObjectId(req.session.loggedIn) }, function (err, result) {
                        if (err) return next(err);
                        res.locals.userCommentlist = result;
                        res.render('index');
                });
        });
} catch (err) { res.render('index'); }
});
```

Brief Output

Add New Comment:



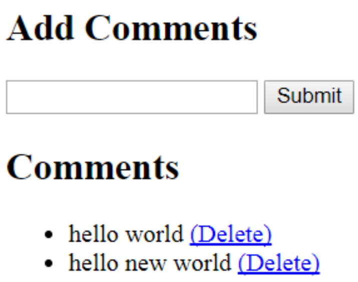
Add Comments

hello new world

Comments

- hello world (Delete)

Comment Added:

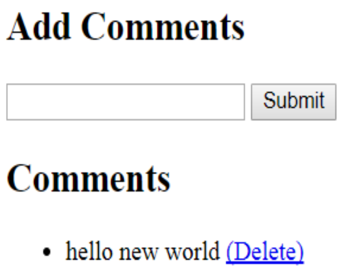


Add Comments

Comments

- hello world (Delete)
- hello new world (Delete)

Delete First Comment:



Add Comments

Comments

- hello new world (Delete)

Errors Encountered

An error was encountered when attempting to add ObjectId as a data type in schema, but was solved by referring it to Schema.Types.ObjectId instead of ObjectId.

Advantage Provided by Mongoose Module

Using Mongoose allowed me to easily perform operations on a collection simply by referring to the Model. It also allow easy indexing, and checking a document's validity through the Schema, which makes the process very convenient.

Task 5A

Source Code #1 (/Model/messages.js)

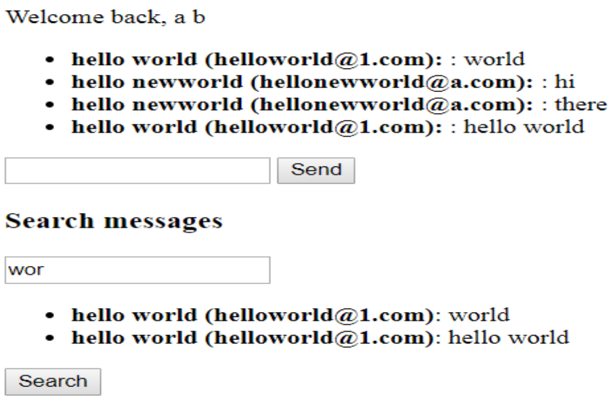
```
var mongoose = require('../node_modules/mongoose')
        , Schema = mongoose.Schema
        , ObjectId = Schema.ObjectId;

var messageSchema = new Schema({
        userMessage : { type: String, index: true },
        messageDate : { type: Number, index: true },
        userID : { type: Schema.Types.ObjectId },
        author : String,
        email : String
});

module.exports = mongoose.model('messages', messageSchema);
```

Source Code #2 (/Model/messages.js)

```
// ensure user exists
users.findById(userID, function (err, userDoc) {
        if (err) { return next(err); }
        // user exists
        else {
                socket.userID = userID;
                socket.username = userDoc.first + " " + userDoc.last;
                socket.email = userDoc.email;
                socket.broadcast.emit('announcement', socket.username + ' joined the chat.');
                // get last 4 messages
                messages.find().sort('-messageDate').limit(4).exec(function (err, message) {
                        message.reverse().forEach(function (messageDoc) {
                                socket.emit('text', '<b>'+messageDoc.author+' ('+messageDoc.email+')</b>:
', messageDoc.userMessage);
                        });
                });
        }
});

// search for message
messages.find({ "userMessage" : { $regex: q, $options: 'i' } }).sort('-messageDate').exec(function (err, message) {
        fn(message.reverse());
});
```

Brief Output

Last 4 Messages (with Author and Email), and Searched messages in database that include a particular world (e.g. wor, world, and etc)



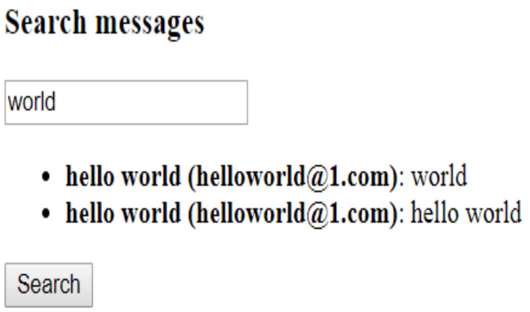
Welcome back, a b

- **hello world (helloworld@1.com):** : world
- **hello newworld (hellonewworld@a.com):** : hi
- **hello newworld (hellonewworld@a.com):** : there
- **hello world (helloworld@1.com):** : hello world

Search messages

wor

- **hello world (helloworld@1.com)**: world
- **hello world (helloworld@1.com)**: hello world



Search messages

world

- **hello world (helloworld@1.com)**: world
- **hello world (helloworld@1.com)**: hello world

Errors Encountered

An error encountered was of the ordering of the results when finding the last 4 messages, in which it was solved by using a .sort('-messageDate') and reverse() on the result.

Task 5B

Improved message display functionality using ng-repeat

Source Code #1 (/Public/js/chat.js)

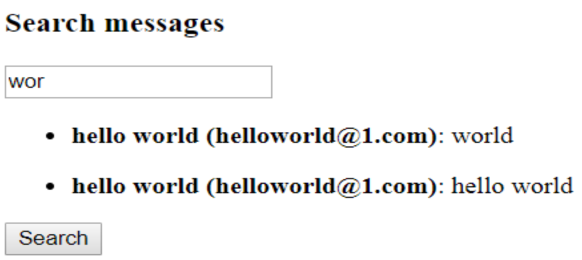
```
socket.emit('search', document.getElementById('s').value, function (messages) {
        var appElement = angular.element(document.querySelector('[ng-app=myApp]'));
        var $scope = angular.element(appElement).scope();
        $scope.$apply(function(){
                $scope.messageResults = messages;
        });
});
```

Source Code #2 (/View/index.ejs)

```
<ul id="results" ng-repeat="message in messageResults">
        <li><b>{{ message.author }} ({{message.email}})</b>: {{message.userMessage}}</li>
</ul>

<script>
var app = angular.module('myApp', []);
app.controller('myCtrl', ['$scope', '$rootScope', function ($scope, $rootScope) {
        $scope.messageResults = [];
}]);
</script>
```

Brief Output

Displaying Messages using AngularJS



Search messages

wor

- **hello world (helloworld@1.com)**: world
- **hello world (helloworld@1.com)**: hello world

Errors Encountered

An error was encountered when referring to an angular element to access its scope, which was fixed by referring to the correct element instead, which is the angular app.

Reason for Combination of Node.js, MongoDb, and AngularJS being Effective

All 3 technologies use JSON to pass values around, which can be stored in MongoDb, retrived and utilized without much restriction in Node.js, then assigned to an AngularJS variable and displayed using directives and other functionalities that are available. This makes the process simple, convenient, and very much straightforward.

Task 6A

Source Code #1 (/Model/friends.js)

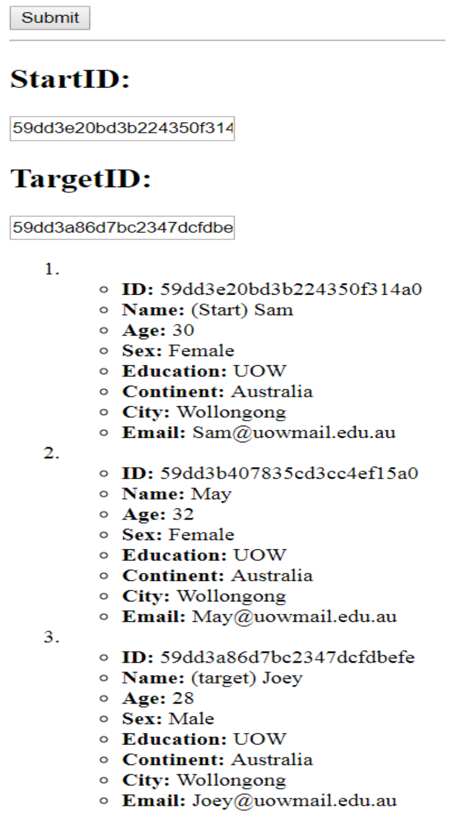
```
var mongoose = require('../node_modules/mongoose')
    , Schema = mongoose.Schema
    , ObjectId = Schema.ObjectId;
var friendSchema = new Schema({
    name : String,
    age : Number,
    sex : String,
    education : String,
    continent : String,
    city : String,
    email : String,
    friends : [Schema.Types.ObjectId]
});
module.exports = mongoose.model('friends', friendSchema);
```

Source Code #2 (/server.js)

```
server.post('/find', function (req, res, next) {
var path = [];
friendsVisited = []; // reset friends array
findNext(initId, function (resObj) {
    if (resObj.data.length != 0 && path.length == 0) {
        path = resObj.data;
        path.push(initId);
        path.reverse();
        // change id strings into object
        for (var i = 0; i < path.length; i++) { path[i] = mongoose.Types.ObjectId(path[i]); }

        var pathDetails = [];
        getDoc(0);
        function getDoc (index) {
            friends.findOne({ _id: path[index] }, function (err, doc) {
                pathDetails.push(doc);
                if ((index + 1) < path.length) getDoc(++index);
                else { res.render('index', { result : pathDetails }); }
            });
        }
    } else if (path.length == 0) { res.render('index', { result : [] }); }
});    });


function findNext (id, callback) {
friendsVisited.push(id); // add current id to visited array
friends.findById(id, function (err, userDoc) {
    if (err || userDoc == undefined) {
        callback({ response: "error", data: [] }); // cannot find id
        return;
    } else {
        var friendsToVisit = [];
        if (userDoc.friends != undefined &&
        Array.isArray(userDoc.friends) && userDoc.friends.length != 0) {
            // process each friend
            var friendsProcessed = 0;
            userDoc.friends.forEach( (curFriend, index, array) => {
                // check if already visited
                var isFriendVisited = false;
                for (var i = 0; i < friendsVisited.length; i++) {
                    if (friendsVisited[i].toString() == curFriend.toString()) {
                        isFriendVisited = true;
                        break;
                    }
                }
                if (!isFriendVisited) { friendsToVisit.push(curFriend); }
                friendsProcessed++;
                if (friendsProcessed == array.length) {
                    determineRoute();
                    return;
                }
            });
        } else {
            callback({ response: "success", data: [] });
            return;
        }

        function determineRoute () {
            if (friendsToVisit.length != 0) {
                var hasFound = false;

                // visit each friend
                var routesChecked = 0;
                friendsToVisit.forEach( (friend, index, array) => {
                    // target friend found
                    if (friend == targetId) {
                        friendsVisited.push(friend); // add final id to visited array
                        callback({ response: "success", data: [friend]});
                        return;
                    }
                    // target friend not found
                    else {
                        // search friends of each friend (asynchronously)
        setTimeout(function () {
        findNext(friend, function (resObj) {
            if (resObj == undefined || !Array.isArray(resObj.data) || resObj.data.length <= 0) {
                // do nothing, hasFound is still false
            }
            // friend found in friends of a friend
            else if (resObj.data.length > 0) {
                hasFound = true;
                resObj.data.push(friend); // add id to the back (reverse array later)

                    callback({ response: "success", data: resObj.data });
            }
            // all friends searched, but no target friend to be found
            routesChecked++;
            if (routesChecked == array.length && !hasFound) {
                callback({ response: "success", data: [] });
                return;
            }
        }); }, 0);
                    }
                });
            } else {
                // dead end
                callback({ response: "success", data: [] });
                return;
            }
        }
    }
}); }
```

Brief Output

Path from start node to target node:



Submit

StartID:

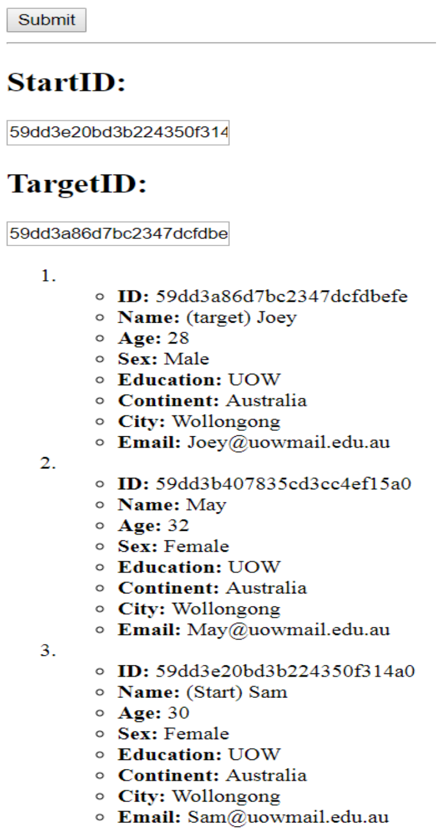
59dd3e20bd3b224350f314

TargetID:

59dd3a86d7bc2347dcfdbe

1.
   - **ID:** 59dd3e20bd3b224350f314a0
   - **Name:** (Start) Sam
   - **Age:** 30
   - **Sex:** Female
   - **Education:** UOW
   - **Continent:** Australia
   - **City:** Wollongong
   - **Email:** Sam@uowmail.edu.au
2.
   - **ID:** 59dd3b407835cd3cc4ef15a0
   - **Name:** May
   - **Age:** 32
   - **Sex:** Female
   - **Education:** UOW
   - **Continent:** Australia
   - **City:** Wollongong
   - **Email:** May@uowmail.edu.au
3.
   - **ID:** 59dd3a86d7bc2347dcfdbefe
   - **Name:** (target) Joey
   - **Age:** 28
   - **Sex:** Male
   - **Education:** UOW
   - **Continent:** Australia
   - **City:** Wollongong
   - **Email:** Joey@uowmail.edu.au

Path from target node to start node (Reverse):



Submit

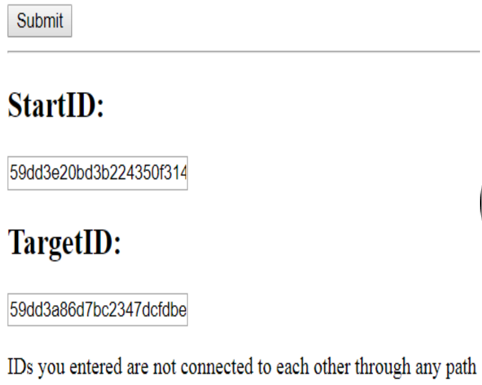
StartID:

59dd3e20bd3b224350f314

TargetID:

59dd3a86d7bc2347dcfdbe

1.
   - **ID:** 59dd3a86d7bc2347dcfdbefe
   - **Name:** (target) Joey
   - **Age:** 28
   - **Sex:** Male
   - **Education:** UOW
   - **Continent:** Australia
   - **City:** Wollongong
   - **Email:** Joey@uowmail.edu.au
2.
   - **ID:** 59dd3b407835cd3cc4ef15a0
   - **Name:** May
   - **Age:** 32
   - **Sex:** Female
   - **Education:** UOW
   - **Continent:** Australia
   - **City:** Wollongong
   - **Email:** May@uowmail.edu.au
3.
   - **ID:** 59dd3e20bd3b224350f314a0
   - **Name:** (Start) Sam
   - **Age:** 30
   - **Sex:** Female
   - **Education:** UOW
   - **Continent:** Australia
   - **City:** Wollongong
   - **Email:** Sam@uowmail.edu.au



Submit

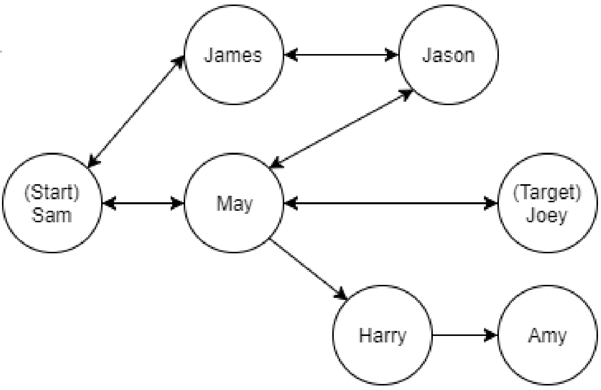
StartID:

59dd3e20bd3b224350f314

TargetID:

59dd3a86d7bc2347dcfdbe

IDs you entered are not connected to each other through any path



Errors Encountered

An error encountered was that the correct path wasn’t displayed in order as model.find() did not find according to the order of array supplied, which was solved by retrieving the document sequentially. Also, there was an error when comparing document id as it cannot be compared directly, which is because they are objects and not string, therefore, it was solved by casting to string with toString() function before comparing.

Task 6B

Source Code #1 (/Public/js/script.js)

```
updateTile = function (curElement) {
pos = curElement.id.split("~");
if (pos.length == 2) {
    if ( pos[0] == 0 && pos[1] == 0 ) { //starting node
    } else if ( pos[0] == (row-1) && pos[1] == (col-1) ) { //end node
    } else {    // change to true - blocked
        if (mazeArray[pos[0]][pos[1]] == false) {
            mazeArray[pos[0]][pos[1]] = true;
            curElement.setAttribute('class', 'tileBlocked');
            curElement.classList.remove('tilePath');
        } // change to false - not block
        else if (mazeArray[pos[0]][pos[1]] == true) {
            mazeArray[pos[0]][pos[1]] = false;
            curElement.classList.remove('tileBlocked');
        }
    }
}   }

findPath = function () {
msgElement.innerHTML = '';
socket.emit('find', mazeArray, startTile, endTile, function (pathArray) {
    for (var row_i = 0; row_i < row; row_i++) {
        for (var col_i = 0; col_i < col; col_i++) {
            var curId = row_i + '~' + col_i;
            var curElement = document.getElementById(curId);
            if (curElement != null) { curElement.classList.remove('tilePath'); }
        }
    }
    if (pathArray.data.length == 0) {
        msgElement.innerHTML = 'No Solution';
    } else if (pathArray.data.length > 0) {
        for (var i = 0; i < pathArray.data.length; i++) {
            var targetId = pathArray.data[i][0]+'~'+pathArray.data[i][1];
            var targetElement = document.getElementById(targetId);
            targetElement.setAttribute('class', 'tilePath');
        }
    }
}); }

resetPath = function () {
msgElement.innerHTML = '';
socket.emit('init', row, col, function (mazeArrayInitialized) {
    mazeArray = mazeArrayInitialized;
    if (mazeArray != undefined) {
        var resultingTable = createTableContent();
        while (divElement.hasChildNodes())
        {   divElement.removeChild(divElement.lastChild); }
        divElement.appendChild(resultingTable);
    }
}); }
```

Source Code #2 (server.js)

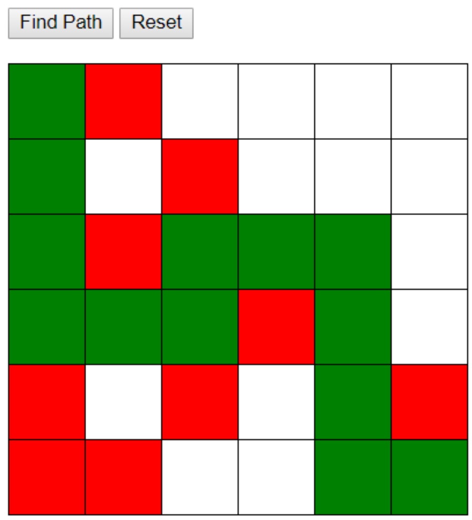
```
socket.on('find', function (mazeArrayUpdated, startTile, endTile, fn) {
    socket.mazeArrayUpdated = mazeArrayUpdated;
    // check valid start and end tile
    if ( (startTile[0] >= 0 && startTile[0] < socket.mazeArrayUpdated.length) &&
     (startTile[1] >= 0 && startTile[1] < socket.mazeArrayUpdated[0].length) ) {
        socket.startTile = startTile;
        socket.endTile = endTile;
    }
```

```
        socket.tileVisited = []; // reset tileVisited
        var path = [];
        findNext(socket.startTile, function (resObj) {
                if (resObj.data.length != 0 && path.length == 0) {
                        path = resObj.data;         path.push(socket.startTile);          path.reverse();
                        fn({ response: 'found', data: path });
                } else if (path.length == 0) { fn({ response: 'blocked', data: [] }); }
        });
});

function findNext (tile, callback) {
        if (tile.length != 2) { callback({ response: "error", data: [] }); return; }
        socket.tileVisited.push(tile); // add current tile to visited array
        var tilesToVisit = [ ];

        // add top, right, bottom, left tiles to check
        var tileTop = [ tile[0] - 1, tile[1] ],
        tileRight = [ tile[0], tile[1] + 1 ],
        tileBottom = [ tile[0] + 1, tile[1] ],
        tileLeft = [ tile[0], tile[1] - 1 ]
        var tilesToValidate = [ tileTop, tileRight, tileBottom, tileLeft ];

        // ensure the above 4 direction tiles are all valid
        tilesToValidate.forEach( (curTileToValidate, tileToValidateIndex, tilesToValidateArray) => {
                // check if valid tile (not out of range)
                if ( (tilesToValidate[tileToValidateIndex][0] >= 0) &&
                (tilesToValidate[tileToValidateIndex][0] < socket.mazeArrayUpdated.length) &&
                (tilesToValidate[tileToValidateIndex][1] >= 0) &&
                (tilesToValidate[tileToValidateIndex][1] < socket.mazeArrayUpdated[0].length) ) {
                        // check if tile is not blocked (true - blocked, false - not blocked)
                        if ( socket.mazeArrayUpdated[ tilesToValidate[tileToValidateIndex][0] ]
                                [ tilesToValidate[tileToValidateIndex][1] ] == false ) {
                                // check if tile is not visited yet
                                var isTileVisited = false;
        for (var tileVisitedIndex = 0; tileVisitedIndex < socket.tileVisited.length; tileVisitedIndex++) {
                if ( (socket.tileVisited[tileVisitedIndex][0] == tilesToValidate[tileToValidateIndex][0]) &&
                    (socket.tileVisited[tileVisitedIndex][1] == tilesToValidate[tileToValidateIndex][1]) )
                    { isTileVisited = true;     break; }
        }
        if (!isTileVisited) { tilesToVisit.push(tilesToValidate[tileToValidateIndex]); }
                        }
                }
        });
        if (tilesToVisit.length == 0) { callback({ response: "deadend", data: [] });   return; }

        // visit each valid adjacent tile
        var tilesChecked = 0;
        var hasFound = false;
        tilesToVisit.forEach( (tile, tileIndex, tilesToVisitArray) => {
                // target tile found
                if ( tile[0] == socket.endTile[0] && tile[1] == socket.endTile[1] ) {
                        socket.tileVisited.push(tile); // add final tile to visited array
                        callback({ response: "found", data: [tile] });
                        return;
                } // target tile not found
                else { // search each adjacent tiles of adjacent tile (asynchronously)
                        setTimeout(function () {
                                findNext(tile, function (resObj) {

                        if (resObj == undefined || !Array.isArray(resObj.data) || resObj.data.length <= 0) {
                                // do nothing, hasFound is still false
                        } // target tile found in adjacent tiles of an adjacent tile
                        else if (resObj.data.length > 0) {
                                hasFound = true;
                                resObj.data.push(tile); // add id to the back (reverse array later)

                                callback({ response: "found", data: resObj.data });
```

```
                                        } else { console.log('not found'); }

                                        // all friends searched, but no target friend to be found
                                        tilesChecked++;
                                        if (tilesChecked == tilesToVisitArray.length && !hasFound)
                                        { callback({ response: "notfound", data: [] });   return; }
                                                });
                                        }, 3500);
                                }
                        });
}
```

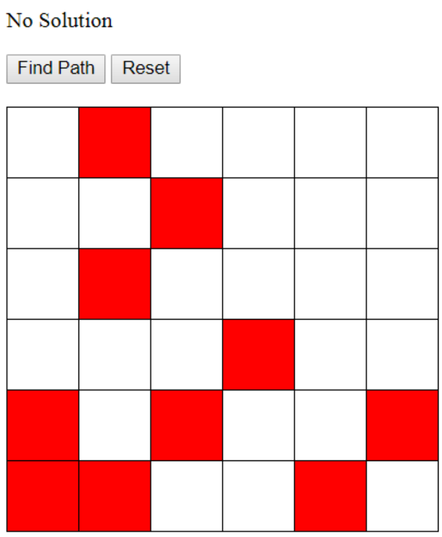
Brief Output

Path Found:



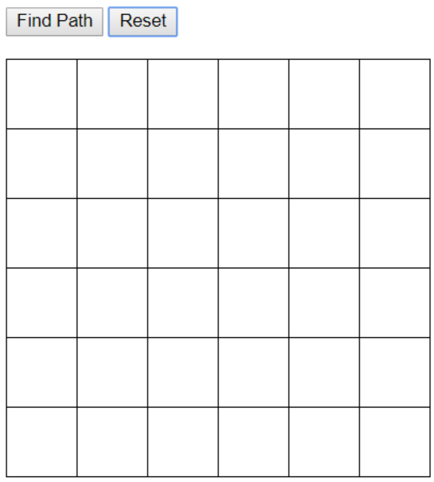
🟩	🟥				
🟩		🟥			
🟩	🟥	🟩	🟩	🟩	
🟩	🟩	🟩	🟥	🟩	
🟥		🟥		🟩	🟥
🟥	🟥			🟩	🟩

Path Not Found (Blocked):


No Solution

	🟥				
		🟥			
	🟥				
			🟥		
🟥		🟥			🟥
🟥	🟥			🟥	

Path Reset / Original Situations:



Errors Encountered

An error was encountered when resetting the maze, as the existing path was not cleared, which was solved by removing the class that gives the green colour at the respective reset areas.

Task 6C

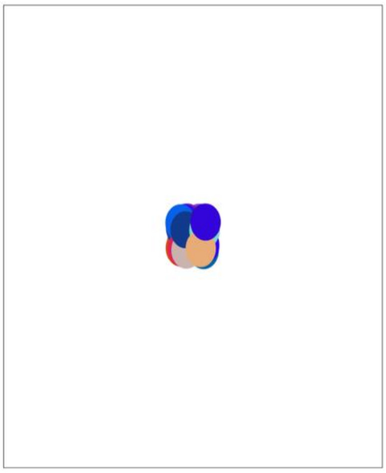
Source Code (/Public/js/script.js)

```
window.onload = function () {
		initCircle();
		setInterval (updateCircle, intervalTimer);
}

function initCircle () {
clearCanvas();
for (var circleNumIndex = 0; circleNumIndex < numOfCircles; circleNumIndex++) {
		var circle = {
				posX		: centerX,
				posY		: centerY,
				radius	: defaultRadius,	// fixed
				color		: getRandomColor(),		// random
				speed	: getRandomSpeed()		// random
		};
		circles.push(circle);

		ctx.beginPath();
		ctx.moveTo(centerX + defaultRadius, centerY);
		ctx.arc(centerX,centerY,defaultRadius,0,2*Math.PI);
		ctx.fillStyle = circle.color;
		ctx.fill();
		ctx.closePath();
}		}

function updateCircle () {
clearCanvas();
for (var circleNumIndex = 0; circleNumIndex < circles.length; circleNumIndex++) {
		// update position
		var updatedPosX = circles[circleNumIndex].posX + circles[circleNumIndex].speed[0];
		var updatedPosY = circles[circleNumIndex].posY + circles[circleNumIndex].speed[1];
		var updatedSpeed = circles[circleNumIndex].speed;

		if ( updatedPosX < (0 + circles[circleNumIndex].radius) ) {
				updatedPosX = 0 + circles[circleNumIndex].radius;
				updatedSpeed = getRandomSpeed();
		} else if ( updatedPosX > (canvasWidth - circles[circleNumIndex].radius) ) {
				updatedPosX = canvasWidth - circles[circleNumIndex].radius;
				updatedSpeed = getRandomSpeed();
		}

		if ( updatedPosY < (0 + circles[circleNumIndex].radius) ) {
				updatedPosY = 0 + circles[circleNumIndex].radius;
				updatedSpeed = getRandomSpeed();
		} else if ( updatedPosY > (canvasHeight - circles[circleNumIndex].radius) ) {
				updatedPosY = canvasHeight - circles[circleNumIndex].radius;
				updatedSpeed = getRandomSpeed();
		}

		var updatedCircle = {
				posX : updatedPosX,
				posY: updatedPosY,
				radius: circles[circleNumIndex].radius,
				color: circles[circleNumIndex].color,
				speed: updatedSpeed
		}

		// updated existing circle
		circles[circleNumIndex] = updatedCircle;

		ctx.beginPath();
		ctx.moveTo(updatedCircle.posX + updatedCircle.radius, updatedCircle.posY);
```

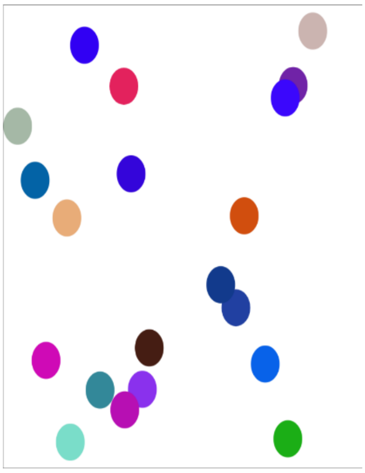
```
        ctx.arc(updatedCircle.posX, updatedCircle.posY, updatedCircle.radius, 0, 2*Math.PI);
        ctx.fillStyle = updatedCircle.color;
        ctx.fill();
        ctx.closePath();
}       }
```

Brief Output

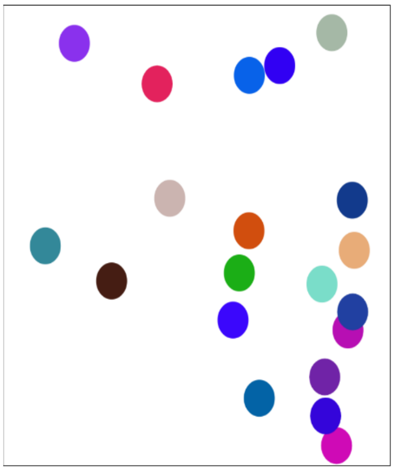
20 Circles from centre



Circles moving at different speed



Circles at different positions and bouncing off of borders



Errors Encountered

An error was encountered when the generated circles were all connected by a line, which was solved by calling ctx.beginPath() before drawing each circle.

Task 6D

Source Code #1 (/Model/users)

```
var mongoose = require('../node_modules/mongoose')
		, Schema = mongoose.Schema
		, ObjectId = Schema.ObjectId;
var userSchema = new Schema({
		first : String,
		last : String,
		email : { type: String, unique: true },
		password : { type: String, index: true },
		authorization : { type: String }
});
module.exports = mongoose.model('users', userSchema);
```

Source Code #2 (/Model/pages)

```
var mongoose = require('../node_modules/mongoose')
		, Schema = mongoose.Schema
		, ObjectId = Schema.ObjectId;
var pageSchema = new Schema({
		pageNum : { type: Number },
		pageName : { type: String, index: true },
		pageContent : String,
		pageColor : [String]
});
module.exports = mongoose.model('pages', pageSchema);
```

Source Code #3 (server.js)

```
server.post('/editContent', function (req, res, next) {
var submittedName = req.body['pageName'].trim();
var submittedContent = req.body['pageContent'].trim();

users.findById(req.session.loggedIn, function (err, doc) {
		if (err) return next(err);
		if(doc.authorization == 'change-content' || doc.authorization == 'change-format') {
				pages.update({ _id: mongodb.ObjectId(req.cookies.curPage._id) }, { $set: { pageName :
submittedName, pageContent : submittedContent } }, function (err, doc) {
						res.redirect('/page/'+req.cookies.curPage.pageNum);
				});
		}
});		});

server.post('/editFormat', function (req, res, next) {
var submittedColor = req.body['pageColor'];
users.findById(req.session.loggedIn, function (err, doc) {
		if (err) return next(err);
		if(doc.authorization == 'change-format') {
				pages.update({ _id: mongodb.ObjectId(req.cookies.curPage._id) }, { $set: { pageColor :
submittedColor } }, function (err, doc) {
						res.redirect('/page/'+req.cookies.curPage.pageNum);
				});
		}
});		});

server.get('/page/:pageNum', function (req, res, next) {
if (req.session.loggedIn) {
		var targetPage = req.params.pageNum;
		pages.findOne({ pageNum: targetPage }, function (err, doc) {
				if (err) return next(err);
				if (!doc) {
						res.cookie('curPage', null);		res.redirect('/');
				} else {
						res.cookie('curPage', doc);		res.render('page', { pageDetail: doc });
```

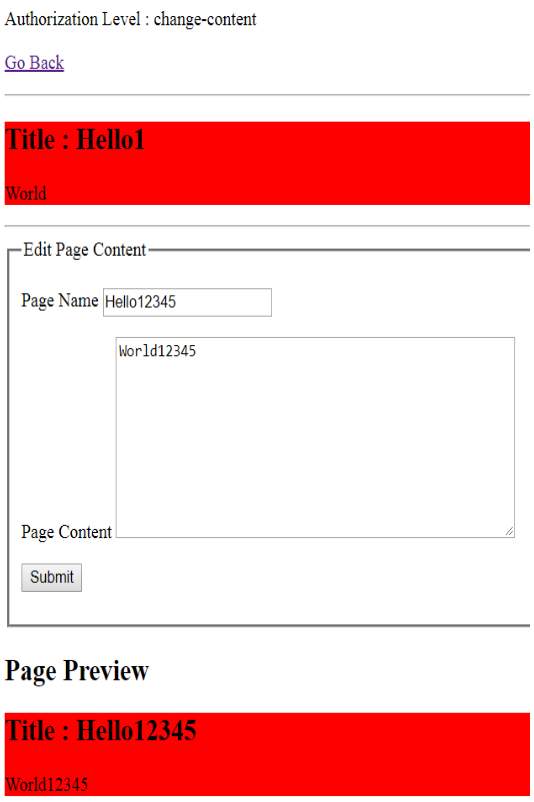
```
                    }
          });
} else { res.redirect('/'); }     });
```

Source Code #4 (page.ejs)

```
<div id="wrapper" ng-app="myApp" ng-controller="myCtrl"
           ng-init="modelName='<%= pageDetail.pageName %>';
                    modelContent='<%= pageDetail.pageContent %>';
                    modelColor='<%= pageDetail.pageColor %>';">

<% if (locals.me.authorization == "change-content" || locals.me.authorization == "change-format") { %>
          <h2>Page Preview</h2>
          <div style="background-color: {{ modelColor }}">
                    <h2>Title : {{ modelName }}</h2>
                    <p>{{ modelContent }}</p>
          </div>
<% } %>
```

Brief Output

Authorization (User 2 - Change Content) and Preview Text Change using AngularJS: (In addition to Read-Only)



Authorization Level : change-content

Go Back

Title : Hello1

World

Edit Page Content

Page Name Hello12345

World12345

Page Content

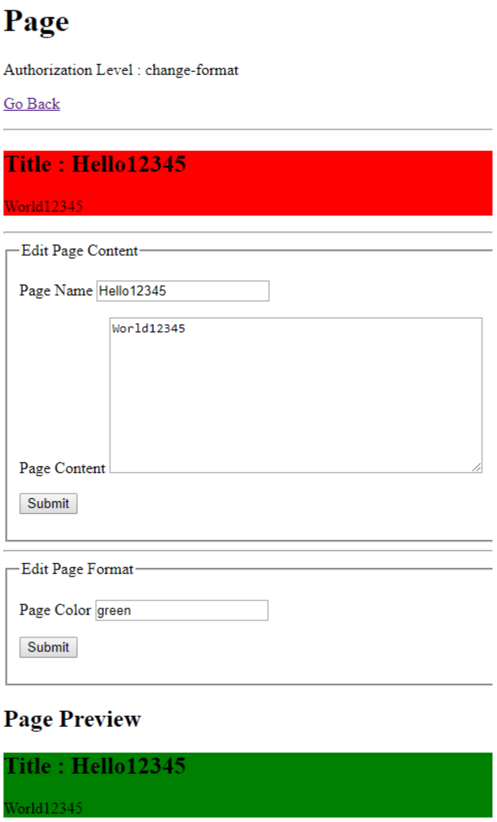
Submit

Page Preview

Title : Hello12345

World12345

Authorization (User 3 - Change Format) and Preview Color Change using AngularJS: (In addition to Read-Only & Change Content)



Page

Authorization Level : change-format

Go Back

Title : Hello12345

World12345

Edit Page Content

Page Name Hello12345

World12345

Page Content

Submit

Edit Page Format

Page Color green

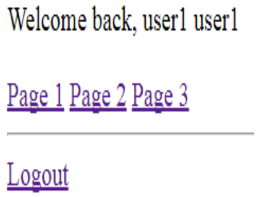
Submit

Page Preview

Title : Hello12345

World12345

3 Pages:



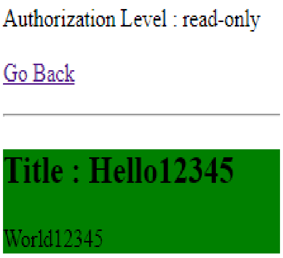
Welcome back, user1 user1

Page 1 Page 2 Page 3

Logout

Authorization (User 1 - Read-only)

Page 1:



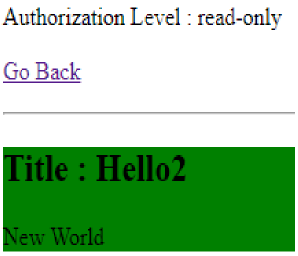
Authorization Level : read-only

Go Back

Title : Hello12345

World12345

Page 2:



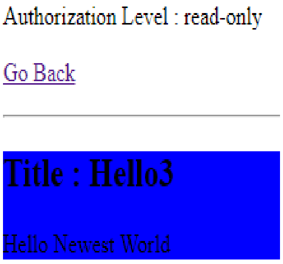
Authorization Level : read-only

Go Back

Title : Hello2

New World

Page 3:



Authorization Level : read-only

Go Back

Title : Hello3

Hello Newest World

Errors Encountered

An error was encountered when attempting to update document as it did not execute, which was solved by passing a callback function so that it executes and uses the callback function for further processing.