

Eric Son
Progress Report 1
CSE 4500-01
David Turner

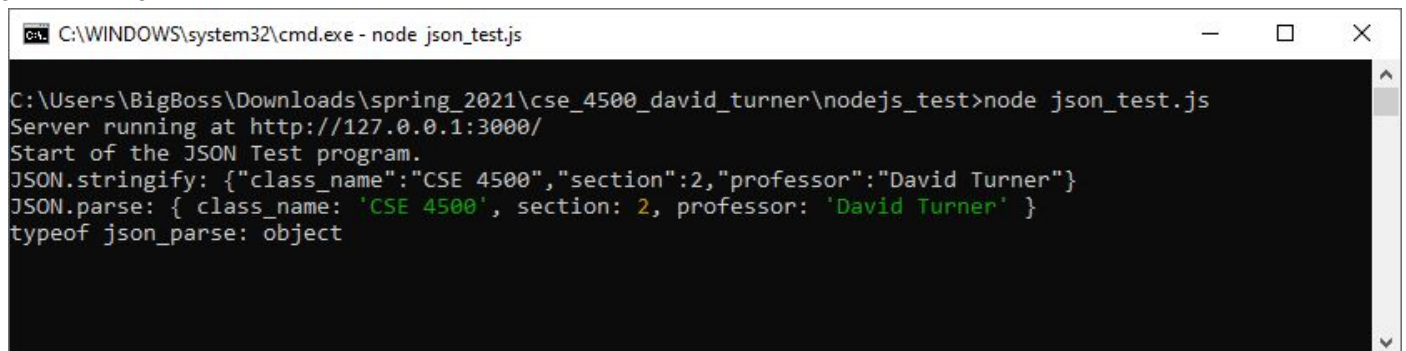
Core Assignment 1:

A private git repository on github will be maintained. Composed of mainly the basic Unity project files as well as a folder for the core assignments and a folder for all progress reports in PDF format.

Core Assignment 2:

A simple nodejs server is used to run code that demonstrates the usage of `json.stringify` and `json.parse`. The server code is modified from the official "hello world" tutorial on [nodejs.org](https://nodejs.org/en/docs/guides/getting-started-guide/).
<https://nodejs.org/en/docs/guides/getting-started-guide/>

json_test.js Output:



```
C:\WINDOWS\system32\cmd.exe - node json_test.js

C:\Users\BigBoss\Downloads\spring_2021\cse_4500_david_turner\nodejs_test>node json_test.js
Server running at http://127.0.0.1:3000/
Start of the JSON Test program.
JSON.stringify: {"class_name":"CSE 4500","section":2,"professor":"David Turner"}
JSON.parse: { class_name: 'CSE 4500', section: 2, professor: 'David Turner' }
typeof json_parse: object
```

json_test.js Source Code:

```
const http = require('http');
const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('CSE 4500-01 JSON Test Program');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);

  console.log("Start of the JSON Test program.");
});
```

```

const json_test = {
  class_name: 'CSE 4500',
  section: 2,
  professor: 'David Turner'
};

const json_string = JSON.stringify(json_test);
console.log("JSON.stringify: " + json_string);
const json_parse = JSON.parse(json_string);

process.stdout.write("JSON.parse: ");
console.log(json_parse);
console.log("typeof json_parse: " + typeof(json_parse));

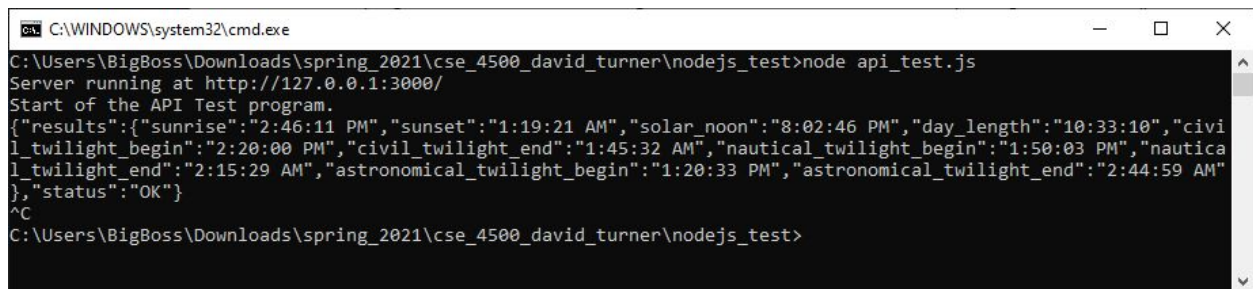
});

```

Core Assignment 3:

This program demonstrates the usage of a function from the node-fetch package called fetch(). fetch() is used to interact with the api from sunrise-sunset.org, in which the latitude and longitude can be specified from modifying the api URL. The text of the response (res) is extracted and subsequently outputted to the console.

api_test.js Output:



```

C:\WINDOWS\system32\cmd.exe
C:\Users\BigBoss\Downloads\spring_2021\cse_4500_david_turner\nodejs_test>node api_test.js
Server running at http://127.0.0.1:3000/
Start of the API Test program.
{"results":{"sunrise":"2:46:11 PM","sunset":"1:19:21 AM","solar_noon":"8:02:46 PM","day_length":"10:33:10","civil_twilight_begin":"2:20:00 PM","civil_twilight_end":"1:45:32 AM","nautical_twilight_begin":"1:50:03 PM","nautical_twilight_end":"2:15:29 AM","astronomical_twilight_begin":"1:20:33 PM","astronomical_twilight_end":"2:44:59 AM"},"status":"OK"}
^C
C:\Users\BigBoss\Downloads\spring_2021\cse_4500_david_turner\nodejs_test>

```

api_test.js Source Code:

```

const http = require('http');
const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('CSE 4500-01 API Test Program');
});

```

```
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);

  console.log("Start of the API Test program.");

  const fetch = require('node-fetch');

  //san bernardino coords - Latitude: 34.115784, Longitude: -117.302399
  fetch('https://api.sunrise-sunset.org/json?lat=34.115784&lng=-117.302399')
    .then(res => res.text())
    .then(text => console.log(text))

});
```

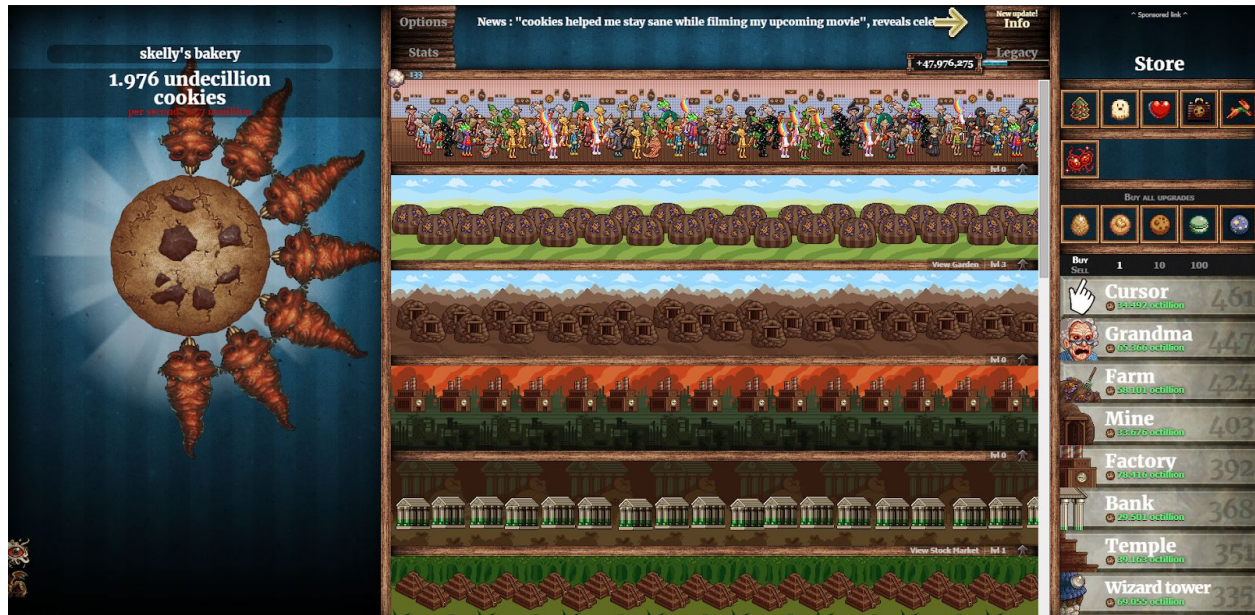
Project Assignment:

The project will be a Unity game with Firebase support.

- Unity project
- Android / Windows 10
- Firebase packages for login authentication, leaderboards, cloud saves, etc
- Genre: Idle/Incremental Game

The game itself will be inspired by the likes of Cookie Clicker and similar Idle/Incremental games in the genre. In these games, the goal is for the player to simply acquire large amounts of a resource through clicking an object, generating passive income from purchased buildings, and other gameplay interactions.

Screenshot of the original game that popularized the genre
(<https://orteil.dashnet.org/cookieclicker/>):



This project will be similar, albeit with different resources and buildings to acquire. Instead of acquiring cookies, the goal is to acquire cheese.

Project Progress for Week 1:

- cheese sprite and background image
- functionality for acquiring cheese through clicking
- functionality for buying a basic building and gaining a passive cheese per second (CPS)
- ui text that reflects changes in game data
- set up firebase console and imported firebase.auth package for later

First prototype of the “Cheese Clicker” UI:

Buyables:

Rats:
Amount: 65
Cost: 20
Unit CPS: 10
Total CPS: 650

Buy x1

Cheese: 8369
Total CPS: 650

