COUNTRY SIMULATION – DATA ANALYSIS

The goal is to be able to store and visualise an approximate game-like state for the lives of all the citizens of a country (NI initially) over the modern era (last couple of hundred years). The challenge of the project is in creating a format for storing this information efficiently so that it can all be stored and queried on a single desktop PC.

# **1st week**

I began the project with an absolutely confused mind with no clue on where to start. I had different ideas coming up in mind, whether to use Flask, Django for a website interface to input the country details for country simulation but again, that wasn’t the main motive of this project. I was getting frustrated because I really wanted to get the work start going on the project but I just had no clue on where to begin from. This is a challenge that I greatly appreciate, but also a very frustrating one at that.

I first started learning mangoDB via w3schools. I downloaded mangoDB, installed it on my computer and created a couple databases and collections and fickled with different myDB commands such as retrieve, update and such. I never used mangoDB before so I had to know on how to use the basics of it. While going through the learning process, I onstantly thought of where to begin my project from. How to start, but I still couldn’t figure out. I then wondered upon taking help of Mr. Bustard to see if he can guide me through the start. But then I thought I should first focus on grasping mongoDB usage in Python first and then think clearly again before reaching out for Mr. Bustard.

So today I saw MongoEngine which is basically a library that provides higher abstraction and can be used for high-level projects, which I think suits the scenario right now. Now I’m learning on how to use MongoEngine with Python because I think it could be really useful considering we are potentially planning to have millions of data? I’m thinking of asking John if MongoEngine is going to be helpful but regardless, I will still learn how to use it.

It still bothers me that even after I learn on how to use MongoEngine, I still don’t know where to properly begin my project from. I need guidance from John though I feel like it would be a bit embarrassing considering I don’t even know where to begin from! But I will surely ask him. I don’t want to go down the wrong path in the beginning. I want to make sure before I begin on the project essentially, I get myself known with different libraries that are used for data analytics in Python to get a speedy progress on the project. I did a couple Google searches and I’m most definitely going to try pandas and NumPy. Although Matplotlib shows up as a recommended too but I doubt I’d need to visualise any data. So I’ll just stick to NumPy and pandas.

I had a bit of slow start today as I spent my most of the time today doing the practical for OOP and also all the additional exercises. It is honestly hard when you’re constantly switching programming languages. I always use python syntax in Java. But I’m really glad I’m part of the project to try something new even though it intimidates me with the complexity. During late night, I used online guides to learn about pandas and it really seems like one if not the most useful library for data science. It’s not so easy for me and its taking for me to learn but its mostly because with every stage of learning, I’m playing around with it and trying new inputs. Hopefully I will have some time left.

Today I spent most of my time tinkering around with pandas and also took quite a big break gaming. I was tired. Also I thought I should interact with my other group member so I sent Samual a message on discord to ask how he was going along with the project. He’s learning numpy and pandas as well so it makes me at least relieved that I’m not going too slow on the project. But it also makes me think if I need to learn numpy as well essentially.

So..I accidentally slipped the laptop out of my hands, it fell down and it’s been sent for repair. It’ll come back tomorrow but I’m quite angry. Not in a mood to do anything studies related.

So far for the past two days I have done literally nothing ever since I slipped the laptop. I think I’ll start back again from Fridays. Should never slack behind. Also John sent us an example diary and I’m wondering if I have to note in the diary daily or is it okay to sum up like I’m doing now? But hopefully in the meeting with John, I can get a clear shot at where to begin the project from.

**Websites used –**

* <https://www.analyticsvidhya.com/blog/2016/01/complete-tutorial-learn-data-science-python-scratch-2/>
* <https://www.w3schools.com/python/python_mongodb_getstarted.asp>
* <https://towardsdatascience.com/using-mongodb-with-python-bcb26bf25d5d>
* <https://docs.mongoengine.org/tutorial.html>
* <https://www.tutorialspoint.com/python_pandas/index.htm>

**Questions to ask –**

1. Any tips/guidance on where to begin the project from apart from learning how to use mongodb and such?
2. Do I need to record in the diary daily or can it be a weekly sum up?

# **2nd week**

**Saturday & Sunday**

I decided to give the project a break on Saturday and Sunday in order to finish other module’s practicals and extra assignments. I think if I do that, I’ll have the rest of the week free to work on the project specifically other than specific studying. Just wished I would be studying Python instead of Java and the project would be more seamless. But challenges it is.

**Monday**

Although I still was confused a bit on what sort of ‘template’ to use for in the project, a couple Google searches helped me dearly. I checked out the JSON templates and immediately knew that I was supposed to use this. I used w3schools and realpython to know how to use JSON templates in Java and it felt like for the first time, I knew where I was going! After a bit of learning, I then had to know how to integrate JSON with mongoDB. Luckily, I could spend my Monday playing around with JSON as I have not any classes, just practical. I sent my first piece of template to John for feedback. I was a bit worried because I thought it was too basic, but I just wanted to be sure I was on the right track at the start.

**Tuesday**

John’s feedback gave me a picture perfect situation of what I had to do. His attached file also really helped me in knowing how to arrange the different rooms, states and roles especially with naming.

MongoDB has a really great documentation on how to use JSON with MongoDB and it worked for me in trying to put into MongoDB my first database. Although sadly the first thing it encountered was an error, an answer to someone’s else question on StackOverflow (about insert\_many) method helped me. I guess this is the start of lots of error messages. But although my code wasn’t basic, I tried to differentiate between many and single data for insert method.

**Wednesday**

I made a thorough template for the rooms, states, object and roles. As I was making through it I thought, will I need to make hundreds of this? Or can I just make a function to make them based on the input? I hope I can do that early and I won’t have to manually make hundreds of these templates..its a bit tedious. Also I wondered how I’ll link these states and roles to each other. I should probably brush up on my array skills while I wait for feedback from John when I send it. One very huge problem for me during this was deciding on what to use for what object; like where should I use lists, dictionaries and arrays. It was all confusing to me but I tried my best to do it. I then sent it to John for feedback. Maybe I should look up at the definitions and usages of dictionaries and arrays to see what fits best? Got the feedback back from John and it helped me a bit on clearing my confusion; just a bit. But I could see what I had to do next after making templates – functions. It gave me some direction on what to proceed the project in.

**Thursday**

I re-read John’s feedback and honestly, it was a bit complicate for me. But I think its because I haven’t practised much and this is the very challenge that I need to solve myself. However I still was confused with which datatypes to use for, especially after John said to use array of dictionaries for room\_day\_template. I felt like I needed a break.

So I instead remembered that I had to use Flask as well for my project. I downloaded Flask to build a simple webpage and tinkered around with it. The documentation was really well described. I had used Django a while back for some basic web development and I kind of hoped we’d use it.

I used freeCodeCamp on knowing how to use flask and deployed it from my PC too! Oh boy, I need to use this to show those massive data. I can’t wait to be done with the templates and get to the functions part. I was looking forward to see examples of flask webpages with mongoDB but I was too tired so I called it a day.

**Questions to be asked –**

1. Am I going too slow? (I get a bit worried about my pace as I go through my project so an answer would really help me to focus)
2. Should I list websites used in my diary?

# **3rd Week**

**Friday**

On Friday, I took a break as most of my day was involved in spending on multiple classes throughout. I ultimately got some final feedback from John on the templates and how the involving JavaScript functions would work. I had an earlier question about using arrays and dictionaries in the template and now understood that basically the room\_day\_template will be an array of dictionaries of different rooms. These rooms will then have an array of time periods for room and object states. John even provided me with a sample template on the basis of which I’ll build up the template.

Also, I was reminded by John to keep asking questions if I have them instead of wasting time. I will keep that in mind closely.

I planned the tasks for the next week ahead:

1. Finish the template for rooms with one room, several objects at two different state of time and two different roles.
2. Start learning how to use Flask with Python and JavaScript and making some pages.
3. Begin writing the how-to guide in an HTML page for assessment one.

**Saturday & Sunday**

On Saturday, I summed up John’s feedback in an elaborate word document from last supervision meeting because it was the backbone of templates, the first initial step of the project. Then I proceeded on to finishing up the additional exercises of another module. I’m quite of a completionist myself so I don’t like any additional bit of exercise left undone especially if I’m already in a day.

Later on in the day I sat down to finish the room template. I picked up living\_room as the first room for a template and assigned it a unique ID. I gave the living room two different state of time, 3 AM and 3 PM. I assigned the time on a 24-hour clock and kept in mind from the last supervision meeting by what John said that in future we will use UNIX for time. I have heard that term before so I’ll closely look into how to implement it in my code soon. For each state of the room, I gave the living room same objects however different roles. Just like a real life example of how nobody would be at a living room at 3 AM in the night but would be eating or sitting on a chair at 3 PM. The objects had different state. Now, just like there was an array of dictionary for objectState, I did the same for roles. And each role had a unique role ID but, also with that had an object\_ID to relate if the role was doing an action with an object. If the role (person) was doing an action, I included an array for it as well.

I think this will suite well all the necessary information and nothing bloated since John said to make sure to minimise storage. However, I’m not sure if we have to add separate arrays under a role for the action as a necessity.

On Sunday, I couldn’t work on the project as I was drafting my CV/resume alongside applying for summer placements. It’s too much work for me for just this so I gave everything else a pause till I was done with it. But I’m not done with templates anytime soon. I need to write the template for people(roles) as well the next thing.

**Monday**

On Monday, after doing some practical classes I started my first learning on Flask. Following the Flask documentation (<https://flask.palletsprojects.com/en/1.1.x/quickstart/>) which was very extensive, I found it to be pretty simple and easy to setup and use for multiple pages. However for the beginner’s guide, the Flask documentation is very minimal and I for one needed to search for other guides alongside the documentation to get a better understanding of flask. One thing I was heavily invested in was the search for how to link Flask with Python & JavaScript. I followed the guide on <https://www.freecodecamp.org/news/how-to-build-a-web-application-using-flask-and-deploy-it-to-the-cloud-3551c985e492/> and it turned out to be really helpful for me alongside the documentation on using HTML & CSS too with Flask which would of course be needed for UI design.

**Tuesday**

On Tuesday, I first started to learn how to use Python and JavaScript by <https://www.jitsejan.com/python-and-javascript-in-flask.html>. Granted, it was really tough at start and I had to search up for the different GET/POST methods from search on Google. It took half of the time spent on the project today to actually understand what the JavaScript function was doing but I got a grasp of it. It was generating a canvas and allowing to draw dots on the screen. The example however had plenty exploration of JavaScript, HTML & CSS with minimal touch of Python, just for the POST request. I’m not sure if it was time well spent but I enjoyed going through it and trying it out for myself. I think I’ll need to use the same call method when fetching or altering results for my project. Also, I had almost completely forgotten about the how-to guide and thought to save it for tomorrow thinking it won’t take me too long.

**Wednesday & Thursday (How-To Guide)**

Before starting up the work on the how-to guide, I read John’s announcement on the how-to guide advice. I feel like I have underestimated it and it will take a lot more time than expected. Also I’m not really good at writing tutorials but at least this is the first time I’m writing up a tutorial and I’m actually happy that I’m doing something new! Especially if it helps someone else..

I started drafting up the guide and it took me a couple few hours already and I haven’t even been to the middle part of the guide yet! I will briefly explain how I made up the diary in a separate section below. I don’t think I particularly did something so far that must be hard for others to follow up but I’m happy to make a tutorial of what I have done if it helps others.

I divided my how-to guide in three different steps which I thought would be a good way for someone to learn who are new like me since it helped me know which step to begin from based on my previous knowledge. I first gave a brief introduction of the project, what this guide is and what is the goal of this project. I thought adding a description is a must to make sure the reader can get to know if the guide is exactly what they need, saving time.

For the first section of the guide, I listed all the possible technologies including languages, frameworks and libraries that are a must to know for advancing into the project and which are recommended to know if someone wants to do complex scenarios with this project, expanding on just rooms and country simulation. For example, I listed MongoEngine which is great if you want to expand on MongoDB for further complex scenarios. I added several links to these individual technologies making it easier for the reader to find the guides. At least this separate step will help them progress efficiently. If they don’t know certain ‘must’ languages, they can learn it first before delving into further steps and getting confused. I also ensured I listed other languages if someone wants to take this project to a next level.

The step is what I would say the most important step of the whole project. This very step lays of the entire framework or I’d say blueprint of the project, from what needs to be done at the start with templates to what needs to be done at the end in making sure the project is efficient enough to manage millions of rooms. I took extra care and details in describing the template and the start of the project because for me, it was the most difficult step to understand. I just didn’t had any clue on how to start on the project until John helped me out. I feel this will be very very helpful for people like me who are complete beginners. Also, I made sure I listed the feedback I received from John too which was beyond helpful and a saviour of the project for me. After explaining how the rooms, objects and roles will work in the guide, I stated the next steps for JavaScript functions and then Flask. I didn’t explain briefly about them as I haven’t had started working on them (in brief for the project) yet.

Step three of the project gave more insight into the templates and using MongoDB and JSON. I added a sample demo for the room templates to insert the data in a MongoDB Collection and view the output. For the entire demo, I added my code images alongside the whole code with descriptive guides links. I even showed the output to make sure the reader is doing everything right. Then I showed the template of room\_day\_template and explained the step one in a practical manner with numbers and names; explaining what the room state could be. I felt this was needed to get a practical grip of what stage one really meant and also help them visualise the rooms, as I did. In the end, I gave a small introduction of Flask and some guides as I had myself not used it much for the project.

I wrote the guide in plain word document, then converted it to HTML online. I downloaded a HTML template from <https://www.html5webtemplates.co.uk/templates.html> and used Komodo Edit to edit the HTML webpage to suit the project. The guide had various pictures and guide links attached.

This project will really help those who are seeking to simulate a country’s growth or activities over the years or simulate a game. In this field, I think this project will be very helpful for real-time simulation. And an easy UI to see the growth with just a slider will be helpful for case study or analysis of data. I made this with the help of searching about other guides on the net such as [https://www.pyimagesearch.com/2019/12/16/training-a-custom-dlib-shape-predictor/.](https://www.pyimagesearch.com/2019/12/16/training-a-custom-dlib-shape-predictor/)

Phew. It took me so much time to complete on the guide! But I’m glad I have done something like this for the first time. I really hope someone finds it useful and if not, I’m proud I still made it! Unfortunately due to this, it took so much of my time that I couldn’t progress further in the project. Time to get on the project in the next few days!

# **4th Week**

On Friday, I wrapped up the last of the how-to guide and submitted it. I couldn’t do much work on Friday since my whole day is spent on classes. However I planned the rest of my week ahead:

1. Learn to implement MongoDB in a Flask application to query results
2. Query the results from the actual project roomtemplate to basic Flask web application
3. Ask John if I should proceed with making the UI better and flexible once 1 & 2 is done or go back to making the room\_template more detailed to fit with the project
4. Added on Monday – Fix the how-to guide and modify it according to John’s feedback

**Saturday & Sunday**

Even though I read some guides on using Flask with Python and JavaScript, I thought they were of little use when it came to using MongoDB with Flask. The different guides uses different libraries such as MongoEngine or pyMongo. The guides aren’t really detailed much but the documentation is okay to continue with.

At first, I tried to use MongoEngine (<https://pythonbasics.org/flask-mongodb>) but the tutorial wasn’t explanatory enough so I used pyMongo. I spent a couple hours just troubleshooting with installation of pyMongo. The issue was with my pycharm IDE not automatically importing pymongo and after searching through Google for hours, I re-installed my IDE alongside Python and that solved the issue.

Though pyMongo is simple, I plan to use MongoEngine for the project instead. I found a detailed dev.to guide (<https://dev.to/paurakhsharma/flask-rest-api-part-1-using-mongodb-with-flask-3g7d>). I used pyMongo with a guide and with a sample small ongoDB database to show up on the browser.

I created a TODO Reminder which lists all the task, sorts the task with ‘completed’ and ‘uncompleted’ category. For this, as per the guide, I used different route definitions and HTML files for different usages. However tomorrow, I plan to add the updating of MongoDB data in this too.

**Monday & Tuesday**

Today, in the sample pyMongo project, I added the functionality to add a task which will also add data into the MongoDB database. This was fairly easy but MongoDB Compass was really helpful in checking the data types and using that data type in the code.

I received John’s feedback on my how-to guide and I think..I didn’t spend too much effort in the guide as I should have. So I’ll add another task on my to-do list for this week – remaking the how-to guide as per John’s feedback. As much as I’d like to stay on the programming side, I think its better if I keep my guides updated as I go along as well so I don’t feel overwhelmed at the last time.

At first, I changed the template of the guide and added various new images from the project. I followed on the feedback of every aspect John mentioned and also fixed some of the coding style (comments) and changing the theme of the guide from just creating templates to adding how to compress this extensive data alongside how it could be useful for others. I searched up on some related projects such as Sim City Glassbox engine.

After this took me hours and hours again, I went back to the project. I forgot that I had to import the JSON template into the MongoDB database. I did that for a sample project but not this project. So I did that and well..I encountered a TypeError that document must be an instance of dictionary. I didn’t have clue with this error so as pretty usual, I searched it up on Google. I tried some of the solutions offered such as skipping jso.dumps call and using bulk insert without dumping the dictionary but that was of no help. I thought I should post my own Stack Overflow question and see if someone can help me with it. I paused the progress of the project till I received an answer.

**Wednesday**

So..I didn’t get any reply in my stackoverflow question and instead my thread got locked. I couldn’t figure out any way to solve this I so sent John a message with this issue. I was a bit hesitant again but I think I should rather ask questions to him directly which I cannot find answer to without wasting time. He sent a different version of my code and I tried to run it too but unfortunately, it showed up another TabError of Inconsistent usage of tabs and indentation. I changed my IDE settings because someone suggested that in a similar StackOverflow question to tackle the problem but that was of no help.

Then I thought, I think I’m wrong for using bulk insert. My JSON had only a single document and not multiple documents so instead of using insert\_many, I should use just insert\_one? I did that and the code worked! I was so glad because it was the first time in this project I felt like I tackled a tough problem (at least for me). I really think I should put this on the guide as well as this could be a nice reference. Though it took me some time, I had my JSON template on the database and all was left was displaying it to the Flask application.

**Thursday**

My first goal of today was showing all the data from MongoDB database in the browser. I didn’t really want to sort it out as per the time, room or objects..but to display all the data, particularly in raw JSON form. I wanted to take it slow with one step at a time that’s why. I checked the data structure of the JSON template using MongoDB Compass and well..it was tough for me to contemplate how to show the data with various arrays and dictionaries. I first used MongoClient to print the data in raw format because I didn’t know how to proceed with arrays of arrays as in my JSON template. All the other guides on the web had very simple JSON data that was easy to fetch. MongoClient returned several errors upon execution and the same as couple days ago, document must be an instance of dictionary and the Flask web server itself wasn’t starting.

I google up to find a solution but mostly related answers were in pyMongo so I used just pyMongo and not MongoClient. As I was planning to fetch raw JSON data, I used jsonify to return the data. Luckily this time the web server started and the number of errors were reduced. But, I got another error.

‘TypeError: unhashable type: ‘dict’’ – This I couldn’t resolve my head around. I think the problem could be because I’m fetching a dictionary which is unhashable and cannot be members of a set. Is it because I’m using a dictionary inside a dictionary? I searched for some solutions to my problem but many comments regarding similar questions were that the JSON isn’t valid. But my JSON was pretty valid. Should I convert the dictionary to a hashable object like ‘tuple’ and use it as a key for dictionary or instead fetch individual data as separate keys instead of fetching the entire dictionaries? I think I’ll try both the solutions to see which one works best after meeting with John as I’m already overwhelmed with a whole week of errors.

# **5th Week**

The **tasks & updates** for this week are:

1. Starting to create the web-page summary of the project.
2. Partially updating how-to guide to make it better as per John’s recommendation.
3. Focusing on the above two to ensure I don’t slack behind for assessment all of a sudden and can work on the programming side of project easily.
4. Making UI of the webpage more user-friendly and good to look at.
5. Solving the past week’s problem of displaying database in a web page.
6. **Update** on project progress for this week – I expect to give less time to the project this week because of a group project to submit on Monday, and two tests (one from QUB and other for Stock Market Challenge) on Wednesday with one being next Monday.

**Friday to Tuesday:**

* After the meeting with John, I figured to make some changes in fetching the result of JSON output. That should help with the JSON fetching and also, as I outlined earlier the reason I’m unable to fetch all this data is because I’m trying to fetch everything as once. John said I should use HTML and fetch individual data at once. What he linked me (the formparser and jsontoform converter) seemed really cool to use. Using an HTML link to get data and adding data in links is very smart and I will use that approach too. I think with that help I will be able to display the database on the webpage and cross that milestone.
* For now for these couple of days, I mostly focused on the group project that I have to submit on Monday but still worked on the project at times.
* I spent some time searching for similar projects that I could take some inspiration from as John suggested but it was a bit tough getting a somewhat similar project. Simscity was the only one I could relate to and along with that, I chose an AI startup project as guidance.
* I have identified some issues that the project might really suffer with in the future – first with space, because at some time we will surely run out of it despite compression and also if we are trying to replicate real-life simulation, we would need procedurally generated objects/rooms to be created with the help of AI. We could do little manually with different variations.
* I think I have a bit of understanding why the project is useful; it can be used in country mapping, gaming and beyond but I’m still not sure how it could be useful for a new reader and to make it interesting for them.
* Beyond changing some technical things of the how-to guide such as template, coding and style, I now thought about at least beginning with the web summary of the project. I don’t want to end up rushing it like I did with the how-to guide at the last moment.
* I read up what the web summary of the project must be on the Canvas page and saw the example of it. To me, it sounds like the web summary is a summarized technical form of how-to guide except it is more about solution to current problem rather than visualising how it can be useful to others. But do I have to make it just one page as summaries usually are or more?
* I used the template of the how-to guide and created a separate page for the summary. It actually took me an hour to think of how to begin the summary and to make it different from the how-to guide. Is it alright to be similar?
* For the first two paragraphs, I just explained what the project is in technical terms, what it aims to be and what have I used. I stated that the SimCity Glassbox Engine was a motivation than an inspiration to work on the project, seeing its possible usage in game models.
* I plan to continue work on this next week, and got back to focusing on the projects I have to submit on Monday and the test on Wednesday.

**Wednesday to Thursday:**

* Having a good test actually gives me good motivation to work, weird as it sounds.
* So as I planned for this week, I need to display the database data to the website before I proceed further with the project. I actually plan to first show JSON and then once that’s done, I’ll use HTML to make a simple and clean UI to display the results from the database.
* What John forwarded to me in the last meeting, the parser is still what I’ll be using but I want to first show something on the webpage, be it basic before proceeding to the advanced stage.
* I plan to get the div element from the body, loop through every object in the JSON object and then append it to the HTML page. The issue with me last week was I was trying to display a dictionary inside another dictionary.
* I used individual keys to return their values and then used jsonify and have been able to print out the JSON data. Finally!! Really gives me some relief that I was able to do this. It kept me gripped for so long. I plan to explain this deep in the how-to guide. This was pretty hard for me and I imagine a dictionary inside a dictionary is not something people come by often so will be a good fit for the guide.
* Now that I can display the JSON data, I will show the plain data from HTML using JavaScript I think. But before that, I started to make my UI cleaner and more simple.
* I believe once I’m able to display data based on user input, I can see what else needs to be added to the template. John said that too. So I have added a simple slider with different values, representing the state of time to help showing what the state of the room will be during that time.
* I formed three columns to show the room, object and the roles present at that time. Seems too basic but it’s a good start for me. I’ve had trouble with going too advanced too quickly before. For showing the MongoDB data, I’ll loop through the results to show the data. I’m confident I’ll be able to do it but I’ll ask John if I encounter troubles to save time.

**Progress Completed this week:**

* Less progress than usual, mainly due to other deadlines of modules coming this week.
* Heavily modified the how-to guide to suit it for the project needs and to make it sound interesting for a reader.
* Started the web summary of the project. Still need to be done much more in the coming days.
* Was able to eradicate the problem of not able to show JSON data to a webpage! Can now show JSON data in webpage.
* Make a sleek UI of existing webpage for showing MongoDB data – made a slider and three separate columns for different values of the database.

# **6th Week**

The **tasks and update** for this week will be:

1. Displaying the database details in webpage in HTML format, rather than pure JSON.
2. Allowing to sort the data selectively. Showing the room state by inputting time.
3. Making the code more efficient with better coding style and adding comments before Assessment 2.
4. **Update on 1 and 2 –** I finished doing them by the end of Wednesday.

**Friday to Wednesday:**

* After the meeting with John, I’m glad that what I’m doing is correct and in the right path. It reassured me. Now what I plan to do next is fetch the results in an HTML page instead of displaying raw JSON data.
* I’m not sure if I should try to use the FormParser now and form a whole URL when displaying results or go with a plain basic format first and then use the former method.
* After a brief consideration, I thought I should first go with the basic method because I have to submit what I’ve done in a working manner next week. I don’t want to present a broken code so I’ll just fetch the results in a basic manner in a text field and then begin working on the advanced code. This way I’ll at least ensure that the code is running fine and well.
* Learning how to display a dictionary inside a dictionary in a complex JSON file like mine required considerable time searching up. I was honestly tired of it first and wanted to take a break even though I hadn’t achieved anything significant.
* One other part that distracted me when working was constantly thinking about the Databases test on Monday where I had to study for it as well. I guess this is what it is to balance different projects to get a good outcome out of both and prioritise what needs to be done first.
* I looked up for Flask API documentation for the request case and checked some related questions on stackoverflow which greatly helped me in figuring out which way to show a dictionary inside a dictionary, a set datatype.
* Finally, I used the jQuery code to use the getJSON() method that will allow me to fetch the data from a file’s location with the help of AJAX HTTP GET request. It took two arguments, the location of the JSON file and the function that contained the JSON data.
* In the code, I used the each() function to iterate through all the objects in the array which also took two arguments.
* I figure I should tell about these technical terms in the how-to guide more, not fill them with the diary much but regardless, I used an empty string to construct rows that contained the data from the JSON objects.
* The append() method was then used to append the string containing rows in the table.
* I tested the code and well, I got a few couple errors. Nowadays I feel like with every code that I make, there is always errors involved haha. The errors were mostly syntax errors and then a run-time error where I used the wrong dataset for the code. I fixed this up early even though it took me a fair amount of time, mostly with the help of StackOverflow for similar questions.
* Once I tackled the errors, I was able to display the data in the form of a table in an HTML page. It looked good and made me feel good for doing something worthwhile.
* Now was the task to also selectively show the state of a room, the objects and the roles within in a specific time. In my JSON code, I showed the room state at two different times so it should provide with enough support.
* I collectively used HTTP GET and HTTP POST request to show the state of room at a specific time. I used a table with different columns for the room, object, roles and state of object and role. For now, the time was to be input in a simple textfield using integers in 24-hour clock format. However in the future, I plan to instead use a drag slider making it convenient for the user to select the time.
* Also I kept in mind that I need to use ASCII code for time. I will implement all these changes the next week after I’m done with all the part necessary for Assessment 2.

**Thursday:**

* On Thursday, I was majorly busy with making the code efficient and also adding documentation in my code. This would really make it easier for people trying out the code for the first time.
* This included adding comments everywhere where it was needed in the code.
* Honestly I’m not used to commenting the code other than in the university assessments. I do comment but it’s a simple one line description of what a certain part of the code does. Writing whole descriptive comments for the entire program was a ‘chore’ for me and I felt so tired doing it already.
* Once I’m done with the commenting, I should just have to focus on the how-to guide and webpage summary for the next assessment submission, easily allowing the next week to go hand-in-hand with the progress of the project and the assessment submission.
* I have to put the slider I made last week for HTML in use for the fetching results. I should remember this instead of making a slider again out of scratch.

**Progress Completed this week:**

* Displayed the data in an HTML page instead of a raw JSON data.
* Personalised the JSON data in different tables and columns, a first step to making it user-friendly.
* Allowed the user to get the room of state at a specific time by inputting the time in a 24-hour clock in a textfield and then getting the results in a table.
* Documented the code by adding comments and such.

# **7th Week**

The **tasks and update** for this week will be:

1. Finishing the how-to guide and webpage summary for submission this week.
2. Uploading the project files into Github
3. Making a better HTML site with a landing page for the project

**Monday:**

* Today I created the landing page of the site. It will hopefully look cleaner and better for the readers and it provides one-stop navigation to the how-to guide, the web summary and the github library.
* Reading through the information about how to make a webpage summary and after seeking a response from John about what the summary should be, I got a clear vision for how I should make my webpage summary.
* I started with the Project Overview which for a brief lists out what the project is about, what it does and gives an interesting introduction to the reader on the summary about the project.
* Next, for the ease of reader, I divided the overview into different parts which inherently are the main focus of attracting someone to read this tutorial – What is the challenge of the project, how can the project be useful of them and how we aim to simulate a real life country in this project.
* The challenge of this project was simple to state and describe – simply minimising the storage to make sure that this database of a country can be queried and viewed in a simple desktop PC. This would hopefully give the reader an idea of what they are about to face as they go along the project. Could this also be the goal of the project? Possibly, but I think this should be the main challenge if not the goal. As the goal could be a mixture of making templates and showing in the database this country simulation.
* I researched into the SimCity’s Glassbox Engine article and couple other data analysis projects to see how their projects can be useful. This gave me an insight on how to make my project appear useful to others too. I knew for a reason that game developers could impart this idea for an open-world immersive sim games so game development was listed as well. But also, real-life companies, governments and organisations could use the data to see the trend of countries.
* For the next part of the project, it was ‘How will we carry on with the project’. This project delved into the technical terms of the project to give a basic idea of how we plan to do this project programmatically. I listed the technologies used in a quick paragraph and how the templates would be made up, then how we will go along this further.
* I again divided this into different parts of what I had done so far in the project that is already covered in the how-to guide and what I plan to do next with some brief descriptions of beyond what I initially planned. I also made sure to add about the ‘life simulation’ project that John told me would merge in the future to feed data in.
* Lastly, I added a link to the how-to guide.
* I think I might add another section for the technologies I believe in description? I don’t know and I’m not sure but I will get back to this on Friday.

**Tuesday & Wednesday:**

* These whole two days I just plan to make a complete overhaul of the how-to guide and finish it up.
* The how-to guide was changed significantly, looked much better and had every points that John had told me to focus on based on the feedback! I added a contents list at the top that linked to every heading in the how-to guide which would easily allow people to browse.
* The introduction was divided into two parts that basically told what the project is about and for whom is this guide for. This took me some time to research as I read every feedback of John that was to make this an interesting and captivating introduction to entice readers to see my guide. Accordingly, the project ‘about’ section included in a not-so technical terms the project overview but including what we aim to do further in the future with this project.
* The other section just included who was this guide for which had links to the SimCity’s Glassbox Engine. I made sure that for every example I listed, example game developers, I tried to list a small summary of why it’d be interesting for game developers specifically and so. I hope I do not sound too amibitious in this and my project’s original work appears short of it haha!
* Then was giving the previous ‘prerequisites’ a nice and neat look with two different sections for what was needed and what was extra which people could learn if they wanted to.
* Next was to get the idea of the project for the reader simply. This was a bit technical and a bit of the vision of the project. The backbone of the project of how we visualise things. One thing I changed this time from the last how-to guide was adding comparisons from the SimCity’s Glassbox Engine. I did not research for this in the last submission but this time after reading some articles, I found that there were quite some similarities in the datasets of the engine of how they used objects like rooms, roles etc for describing things in their game.
* Next up was visualising the template of the project on how we see it in use, then about storing it in such a manner where it doesn’t take too much space – which is the main challenge of this project. Finally for the end of this section, it was about how we aim to view this stored information and how we can expand this project further.
* The other sections were merely a reform of the previous how-to guide based on John’s feedback about adding focus on minimizing storage and making the steps more detailed.

**Thursday:**

* The submission is tomorrow and I’m kind of worried because I haven’t completed /everything/ yet. It’s not much but I believe it’s a lot. But I’ll finish it up, except it will take a lot of time.
* Today I focused more on the coding side of the project. I divided the HTML page for seeing just the database data and seeing room state based on the time to different pages. I used ‘search.html’ for rendering the data after the user inputs time.
* I used the POST request method when requesting the data from the search bar in HTML code to return the state of the room.
* I’m kind of debating whether to use the webpage for displaying the data or use an API like ‘website.com/api/change\_state/3/chair/is\_sitted\_on=true’ for changing the status of an object based on user input. Once the link is visited by the user, it will edit the state of the room.
* Turns out this debate will be done next week because it’s not so easy as I thought and I already have don’t enough time to work on this and the assignment submission for Friday. I will go through it after the submission so for now, I’ll just be showing the data from the database and querying it for the submission tomorrow. I encountered the error where Python doesn’t want to navigate to objectState with ‘data["room\_day\_template"][0]["stateAtTime"][1]["roomState"]’ and returns an error. I got this error after changing my JSON template structure, by placing the state of the room as an array under Time. I deleted this after the error because I think there was something I did horribly wrong. I will go through this fresh without changing the structure of the template which was the root cause.
* The code works pretty fine after testing it out and I think it’s ready for submission tomorrow! Pretty tired doing the coursework this week and I just want to submit the project. But I still have a lot to do tomorrow.

**Friday & Saturday:**

* After receiving feedback from John and making a couple of changes to the project, mainly some coding changes, I asked one of my friends who studied Computer Science as well to give their opinion based on the blog post.
* After asking about the goal of the project, he was able to understand that the goal of the project is country simulation via database and feeding of dataset. However he only thought of creating templates and feeding them into the database and then displaying as the main goal of the project. The challenge of the project i.e. storing in an efficient manner wasn’t clear to him.
* He was able to understand we were making templates for feeding data into the database and fetching it in a browser. However the blog didn’t make him understand how to do it ‘technically’.
* He was able to understand what needs to be done next however the ordering of the goals was incorrect as per him.
* The link for the how-to guide was attached at the bottom so he found it convenient to just go through it.

# **8th Week**

**Monday**

* After submitting the second assessment for the module, I think it was a time to step back and focus on what I should plan up ahead for the project first.
* I had two options to go by – 1) Making the template more complex and rich as in a real simulation first or 2) Add JavaScript functions that filter the requested queries much better like adding date and time or even year.
* Alongside that, I set the main **goals** for this week to be just focusing on one of the above two tasks and initially expanding the code. I don’t expect to give too much time to the project this week because honestly, I have dragged behind the projects of other modules because being too focused on this!
* I also really (hope) my code doesn’t throw up too many errors because recently I tried to use a cloud-based database of MongoDB and for some reason it just couldn’t connect, throwing 503 errors. Guess that’s not a priority for now.
* So I’m thinking of making my template more complex but I haven’t got any ideas so far so I decided to just switch to JavaScript functions for now. Where should I begin with? I couldn’t think something that I could easily pick up to work on but this module is all about challenges so I don’t think I’d encounter anything like that.
* Instead of going both the ways, I created test-cases for my so-far done code. I thought it’d be a good idea for valid test-cases in the final submission and as I had learnt from my OOP modules, test cases are almost a must during software development.
* I used try and catch to raise exceptions especially during the start of the webserver and during internal service errors that I frequently faced in the past.
* My browser had somehow cached the errors when starting the Flask webserver from my code and the TypeError code I had just couldn’t be shown up. Whenever I started the webserver, it’d simply show Internal Server 503 error and even with debugging on, no error would be displayed which would simply make it almost impossible for me to find the code.
* I google a lot about it and some of the suggestions were using logging and changing the level of logging to ‘debug’. I did that but it was of no use. I was tired of this so I called it a day.

**Wednesday & Thursday:**

* I started the webserver and this time the error started appearing which was a TypeError in the list indices. I figured the browser had simply cached the 503 internal server error and that’s why it showed up nothing else? I guess so..but at least I can try to fix the error now.
* I wanted to get started on expanding the templates so I asked John for guidance on how to expand them. In summary:

1. For the JavaScript functions, make code for rooms that allow client to submit a time/date and find closest record to it in the templates.
2. Simulate a day of a person and then judge what parameters do you need. Should have a ‘status’ of person state that causes an action, similar to what I had previously done in the template.
3. After John’s sharing of the examples of people’s routine, I figured the idea to make a smaller section of it.

* I did some modification to the template and where I had the actions for a role and the object associated with its state, I added the ‘state’ for the role which would basically be the cause of their action.
* I started imagining the life of an introvert college student and added the rooms for –

1. Room
2. Washroom
3. Kitchen
4. Road v1, v2(Assuming it’s a single lane road towards the college and it is not far)
5. Shop v1, v2, v3 (with minimal items) (Each version depicts different states of time)
6. College building reception
7. Lecture hall (with minimal people)
8. Computer Lab (with minimal people)

* After creating the templates for each, the version depicted the different times for the rooms. The road had two time for one when the student is going to college/shop and when they are returning home from it. The shop had three different times as well with minimal items and people.
* I created a bunch of couple people, not so many to fill in some of the rooms alongside objects.
* Some objects had actions attached to them and all of the roles had an action and also ‘state’ that is the cause of the action attached to them.
* I particularly plan to group all these templates in different groups based on their location and time just like the example code sent to me by John. This will make it easier to sort them.

# **9th Week**

**Monday**

* For this week, I plan to mostly focus on the feedback from John from last assignments.
* I structured the directory of my code in a feasible way and added a screenshot of it. Then in the how-to guide, I included the structure and gave small yet brief information about what each file did. Also I included in bold some important methods for the code that I used that the reader might find more important to catch up to.
* John said I need to add more screenshots of what is running. Does this mean the code itself or the output of the code? I assume it’s the output. I’ll keep this in mind the next time I work on the guide with the new steps to make sure the output is demonstrated well with screenshot.
* As for the old code, I ran it again and for each step that I considered crucial – mainly the templates, adding them into the database and then viewing them in the browser will be focused on.
* After going through the guide in the next couple hours, I found out I didn’t include the screenshots for the web browser code that displays the output in the browser !!!!
* I did this huge mistake and I feel dumb about it. I will add the screenshots of the output as a priority in the browser.

**Tuesday & Thursday**

* In my Python code, I changed the API structure and the ‘app.py’ majorly. Mostly a rehaul of what I had done previously.
* The previous code would usually render based on static values of the JSON code ‘time’ using if/else. This would be redundant if the template extended beyond just a few time slots and something which cannot be done manually.
* As I remember from the feedback, I need the code to see the state of an object that is nearest to specific time. I will implement this later in second stage.
* For stage one, what I did was take a time and then also a date – which I added into the template – then based on the exact date (first) it’d show up all the time and the state of the room in two separate columns. This was not based on static code which checks on the exact value of time and returns it instead of returning the state at that particular time.
* For the time however, I planned to use UNIX timestamp instead of regular time. Would make it easy to query and also serve as universal without worrying of different formats.
* I searched up for converting time to timestamp in Python and landed on this stackoverflow question - <https://stackoverflow.com/questions/9637838/convert-string-date-to-timestamp-in-python> which basically had various solutions to the problem.
* There was also trouble of timezone. I used UTC instead of local time for uniform time and then used Epoch converter to convert time into timestamp, not particularly date and then fetch the time based on the timestamp – which resulted in some errors of course as usual that I aim to fix.
* Was a bit rough but now I need to think of how to make it so I can find the closest time stamp. For simple time values, I wrote a small piece of code using binary search to find the closest time period and then return the state of that time. I think I could use binary search for timestamp too? I need to google a bit more and research on this.

**Saturday**

* I need to work on the code side more today as next week, I’m pretty sure I’ll be busy with finishing up the OOP module project with deadline on Friday. So I aim to work on templates slightly next week.
* For finding the closest timestamp in the JSON, I used the bisect module (<http://docs.python.org/library/bisect.html#module-bisect>) and specifically bisect\_left that was actually very useful as it takes the key/value pairs from the dictionary; just similar to the JSON dictionaries I had.
* I can find the closest timestamp so far however I have not used that in the browser yet. I just implemented in the console.

# **9th Week**

**Update for this week:**

* I don’t think I’ll be able to do much (or any) work till Saturday as all making my progress this week drastically less. My other project for OOP module is returning a lot of errors than it usually should since I started working on it pretty late, just a week before deadline and it’s taking all of my time to work on it.

**Tuesday & Wednesday**

* I’m going to expand upon the templates, maybe try to imagine how to make it like a real simulation? Going to take a thought on how to do that.
* I already have the rooms for a college student. How can I plan to expand on it to make it more real? I played around with the days in the template and made 3 different days with 3 different time periods.
* What I have in mind is – the 3 days with 3 different time periods will provide a unique activity of a college student to test out with. Once that’s done, I can try this out to find the nearest timestamp and then return the state of a room.
* I really hope that this goes smooth as I haven’t yet used the code to print in the browser with these values. Whenever I try to do something with fetching the results from database in browser, it takes a lot of time and also errors.
* It’s hard to work on two projects simultaneously throwing errors at me and where even the simplest errors are just out of sight.
* I tried to run the code to print the state of the room that is nearest to the timestamp. It basically converts the time to timestamp, and it works however I need to test more of it if it actually works because I just used a simple test case.

# **10th Week (29/04/2021)**

**Plans for this week:**

1. Restructure the coding for project, to make it work efficiently and properly as desired.
2. Focus on just the python part of the project.

**Monday:**

* I started off with restructuring the code from the bottom of the project. I had some basic errors during the stage of the project where I had to show the data of the database on webpage.
* To begin with, I created a simple Python file ‘upload.py’ which just opens the JSON template and uploads it into the MongoDB server. Easily done, with just a few minutes taken.
* I ensured that the data integrity was intact by double-checking as last time the database had different array of objects.
* Once that done and double-checked using MongoDBCompass, I created another file ‘webpage.py’. This file will be the main Python document that fetches the data from the MongoDB database.
* Using Flask and MongoClient from pymongo, because it’s quite easier and there is good reference guide for how to use it, I created a new decorator ‘/raw’ that would show the raw JSON data in the webpage. I looped the data in database, saving every data to a variable and then returning the JSON dump of it.
* The part of tutorial on how to print raw JSON data from database and upload JSON template into the database was done.

**Wednesday:**

* Now I started working on fetching data from the template. For the current project time, I just focused on the ‘room\_day\_template’. All the rooms template for the day. So in the code I had to use ‘room\_day\_template’ as a variable. In the future of course, we might expand on it to make it dynamic according to the template name.
* For the index page, I’d just show all the rooms, their objects and states. Fetching it from the database. And so, I parsed data from the MongoDB in a variable and then rendered it into the template ‘index.html’, passing data from the MongoDB.
* Of course I had yet to create the index.html so this was a risky case. I planned to work on the webpages next week and I didn’t know if this would work, but I was certain it should.
* The index page would only have the GET method as we’re not allowing any type of behaviour from the user.
* For the search method, I had to figure out how to actually make it so that it returns the closest time period of a room from the input value.
* I use a POST method but how do I get the data from the database?
* I spent the last night searching on different functions and libraries that I could use to allow me to do this.

**Friday:**

* Datetime library should work well for what I’m trying to achieve. I’ll rather explain this why as I go through how I implemented my search code because really..it’s hard for me to explain plain.
* Adding another URL decorator, ‘search’ would be the best reroute to perform search. First, I parsed data from MongoDB usual as done before. Then I ensured that executing search is only enabled if the search input is filled and submitted, thus using request method POST.
* For the search, I created an empty dictionary and as the user would input a time to search through the template, I got the time value of the input. Then I checked through the template to find the closest result that is 45 minutes closest to the provided input.
* The timedelta function of datetime library proved to be really helpful in it.
* Once that was done, I parsed the data from MongoDB, checking through the ‘stateAtTime’ to see if the time input is between the margin for the search.
* If yes, then the empty dictionary was added with the name of the room, the ID and the state at that time.
* The dictionary data was returned back to the search.html along with the time input back for use as value in the input.
* I also made sure to return nothing to search.html in case no data was inputted.

**Sunday:**

* I plan to add the final ‘add’ method to the project that will add objects and object states to a room. The URL will be specific to the room ID, to only add in that specific room. Using ‘www.webpage.com/add?id=1’ for example will allow adding object and states to room with room ID 1.
* First the user will select the room to modify by clicking on it, which will get the ID of it using the get request. After fetching the data from the database, we check if it’s the POST method and identify the time input for correct editing at stateAtTime.
* This was a bit of difficult step for me that first led to many errors so I hope the people reading the guide might find this really helpful.
* I validated the room Object and then fetched it, before looping the data from the database.
* What I did next is best explained in the comments in the code but I wanted to search the database with old data and replace it with new room objects. Of course, I had to save the old room objects too and update the original data with new room objects, before returning it to the template.
* This approach as I figured out was best, being basic and yet still to be working well.
* This same approach was for adding object State however, appending the state values with object action was a bit different.
* For the safer side, in case the user did not wanted to add anything, the frontend would just append the data matching the object ID that the user searched for.
* All of this was then returned back to the add.html, returning this data to frontend.

# **11th Week**

**Plans for this week:**

* Focusing on the frontend webpage, specifically the templates to show the data.

**Monday:**

* To begin the work, I created a separate folder for templates this time. Unlike last project files, this should be more segregated and allow cleaner viewing of project files. Making it only simply for others to understand and use.
* Starting with the index page, I created a simple yet clean looking HTML page using certain stylesheets.
* My index page basically had three important elements that I had in mind. First, the index page would show all the rooms, objects and states in the database. Basically everything the database has stored from the JSON template. I think it fits well to show something on the index page instead of nothing. Secondly, there will be a small button for ‘Search’ which upon clicking will redirect to the ‘search.html’ page for searching. Third, clicking on a room name will go to a separate URL followed by the room ID, allowing for room objects and states to be added.
* With the help of ‘webpage.py’, I created several tables with rows and columns to display the data used in the ‘webpage.py’. It should be noted that the webpage was already routed to from the Python code.
* I had to read some fairly popular guides on how to create web applications using Python to do this. I don’t think I need to link them individually but I might..now need to do this for index and search HTML page too.

**Wednesday & Friday:**

* Similarly for search page, I had to implement HTML code akin to index. However, I also added a navigation button back to the index page.
* I added a form and decorated it a bit with external spreadsheets. Using a POST forum, this input would be used to input the time in 24-hour clock. For the input tag, I converted the value to data[‘time’] to fit the template or by default, 12:00 value. Followed by a submit button.
* Similar to index, a table was created that feeds the data from the ‘webpage.py’ Python code.
* This required time but it wasn’t too complex and was merely done by testing and trying every time to fix errors. Also, I forgot to add that I looped over each value in the collection to print out all the details. This was done in index as well as search.
* I then went on to create the add.html page. Honestly, all these things I’m doing..I first thought that they are so difficult, but now that I’m trying them and getting used to them. They seem fun to play along with!
* For the add.html, the code should be fairly understandable to get a grasp of what’s happening. It just creates different columns showing the data, column for adding object, adding object state, showing room objects and object states that are currently in the room.
* Once the user adds an object state or object, it adds up to the current object (state) and also added to the database.
* As usual, this is done in loop.
* After testing out a sample JSON template and trying out the webpages, it worked well as expected and I think so far this seems perfect for the final submission. Now I need to just get on behind other stuff..I have two other courseworks to do along with this one and I have to get on with the other two as well!

**Sunday:**

# **12th Week**

**Monday:**

* I tried to implement the previous template I created of a college student life. However I soon realised that the template was far more ambitious than I initially thought to be made in the time frame for the final submission. There were far too many rooms, roles and objects that I had to consider in different times for a day.
* So unfortunately, I had to change the subject of the template as well to a more easy and simple one – something I could do completely within the project submission time period.
* I remember the last feedback John gave me about the project; he said to include the lifestyle of an introvert person who doesn’t go often much. I planned to use that in my project.
* I created different rooms – living room, bedroom, bathroom and kitchen with individual objects. A single person living in the house, that is, a single role ‘teenager’ was created in the template. I compared ‘teenager’ to a lazy person who wakes up late and spends most of his time on PC and TV during the day.
* The day starts with the teenager waking up late at 12 PM. Then afterwards, he goes to bathroom to brush, heading to kitchen for breakfast/lunch afterwards where he eats yoghurt and bunch of healthy food. He then goes to the living room, finishing his wood and watching TV. Then afterwards he goes to bedroom and at last, he goes to kitchen and back to living room to end his day for the project.
* The template involves breaks of hour and thirty minutes. I made sure to add the time state when the role isn’t in the rooms to signify that he’s not there, aka. There’s no roles, the object states are mostly false for objects like light is on and such.
* The role action helps knowing what the role is doing at that time alongside which object is he using/being attached to.
* This took me many hours to figure out the time and implement. JSON isn’t really fancy to play with, just too many brackets..
* I uploaded this JSON to the MongoDB and then ran the Flask webserver and it works perfect! I can add objects, can search through time and it really depicts all the data.
* I’m done with the coding part of the project. Onto the guides.

**Tuesday:**

* Today, I worked on going through the how-to guide and making major implementations to it. Since I changed the entire way of implementing my project, I had to make some major changes to the how-to guide. I also made sure that I followed in to John’s feedback on my guide – to add some images of the result displaying of my project working.
* First of all, I added the project file structure section on my how-to guide. This will really help people in not getting lost as I usually do when trying to start off my work. Also, I added a description of each file in the project to give an idea of what it is about and what is its functions. Hopefully this will save A LOT of time for readers to trim into the stuff they actually want to use.
* I knew I had the prerequisites and the basic idea of the project correct as my templates were largely unchanged. John gave an ‘okay’ feedback about it as well so I believe this will do fine. I changed minor things like adding that the ‘updation of data’ was done and modified future expansion section to only include ‘deletion of data’ and creating JavaScript function to create room templates based on simple instruction or AI. I could think of this as the most major expansion for the project. Of course there can be a multitude of expansions but I believe this to be the most ideal.
* A new section for understanding the template of the project was created which with a picture explains about the real data we’ll be using, the template and how the data and template fuse together to give an ideal real-life simulation! As previously stated, I’m using the teenager student lifestyle for the project so I made emphasis so readers can actually get a sense of how the template works, with an example.
* I changed the sample insertion code with thew new code and images alongside a proper code to print raw JSON code to the console regardless of the JSON template type. Previously the code ran into some errors with the printing of JSON data due to an issue with the ‘if’ statement trying to know the type of JSON data being stored/retrieved. I have fixed it to print/retrieve any JSON data stored.
* Comments were emphasised upon to give the reader what every function is doing, as per John’s feedback.
* I added an extra section for readers who might have a database username and password as well on how to make a connection with the database. Hopefully it helps all.
* Now for the last park for running Flask webserver, the first image gives a small snap of the webpage Python application. I couldn’t get the whole screenshot of the application so I added in the title to check the code. Also, I forgot to include. Every title of the images have links to the individual files for ease in downloading and navigation.
* For each HTML webpage – index, search, add and raw – I added real-time functioning browser images as told by John to give an idea to the user of how the application might work like.
* I can’t explain every code of what I have implemented in the guide, just to make it so longer. So I have made it clear note for the readers to go through the comments of the code which explains every section to get a grip of what does what.
* Added a small reference link on how the GET and POST requests work.
* The ‘What needs to be improved’ section was updated to add how we need an AI implementation to create rooms etc.
* After being done with the guide, I asked one of my friends to see what he thinks of the guide and what he sees from it without having any background knowledge of what is going on. This was recommended by John so I gave it a go. On his notes, he told me the project file structure gave him a clear view of what the files are doing. Moving on to the project view, he says he got some knowledge of what the project is about but when asked if it is about creating templates for rooms, etc or storing the data in an efficient manner – he said he couldn’t figure out and said it was for both. I think that’s a mistake on my end on making it a bit confusing as I focused on both sides..however I think this will do for the project. The images of the webapp working were of great help to just visualise what the webpage is doing without reading the text.
* I did take a couple notes from my friend to realise I could have improved upon for the ‘aim’ of the project. However I think that my guide focuses on both sides – template & efficient data so I’ll keep it as it is.

**Wednesday:**

* I uploaded my project files to the GitHub after arranging them nicely. I figured I would also upload my how-to guide and project summary to give a more than better understand of the project for beginners.
* I created requirements.md file in the github project which lists all the applications required for the project, the libraries used and the link to the download page of all of them.
* I created reade.md file in the github project which gives first a summary of what the project is about and a simple aim. Then a separate section of how this aim can be achieved. I tried my best to keep it short and not seem too long with the readme. That’s why I added the link to the how-to guide and the project summary in the readme so people who can’t get a better idea can get an in-depth guide of how to begin and understand the project without sacrificing the length of the readme. Then I just created a few simple lines on how to run the project, that is run certain python applications and you’re good to go!
* I went on to update my Project Summary page. I update what I had done – updating data, searching through it and what needs to be done for the future part of the project, whoever wants to progress onto it further.
* After that, I added in the ‘Further Research Models’ a brief view of how I think this project could be expanded by other; how they could use it in their daily- life. On the last meeting with John, he suggested me more about including how we can expand the project and also just giving some insight on what related could already be out there.
* Unfortunately I could not find much related information from the project I am doing. So I was honest in the section that the research on this sort of project is limited.
* Meanwhile I also added the template view image and a GIF of operating the webpage under ‘What has been done so far?’ to better summarize visually in how the webpage has been made for the project’s needs.

**Thursday:**

* I started working on the blogpost today. I don’t think I can sum up my project well in a twitter post due to character limitations so I thought for now, I’d prepare the blogpost as if it’s a reddit post.
* If I was a person who wants to see a similar data simulation project, I would be first attracted by how it can be useful to me at first glance. That is the beginning of the post. I would like to see a working image of the project to actually verify if this is a legit post or not. And also knowing how much work has been done, and how it can be taken further. Taking this imagination of how a person wants to see a post, I had to make such a blogpost.
* I created another HTML page for the blogpost. Before starting work on the blogpost, I thoroughly read John’s feedback from email from the first feedback where he gave some ideas on how to go ahead with the blog post. I’ll explain them as I go through further with my work.
* First, starting with the title. John said to research a title well to make sure it catches the eye of others. I went to programming subreddit and checked out some of the titles of different projects. Some of them were informal and I thought to make it formal as this is an academic project. Thinking of with just writing a simple title or making it a bit more than just a basic title, I came up with ‘A beginner step to efficient country simulation with Python’. Knowing my project well, it was just a basic project. Not too advanced. And the goal of the project was not just data analysis for country but also storage management so ‘efficient’ tag fits it well. Including the programming language might help in catching the eyes more.
* Based on John’s feedback, I tried to make the first few lines of the post interesting giving it a little sight on how simulation can be a bit daunting with storage, yet how it’s a really useful tool with some examples. Followed by that, I gave a slight introduction of the project and how much time it took for me to finish this project.
* Then I gave a small brief info on the technical part of the project from templates to the webpages and what major languages were used.
* John also said to attach images as people tend to look at posts with images more. So I included a GIF of the webpage in work to show the database.
* Followed by that, I shared what the goal of the project is as I explained in project overview already with focus on creating simulation and making sure it is resource efficient. At the end of the post, I included how this project can be expanded upon – and how it can actually be put to use. This will really help someone figure out if this project can be helpful to them and in fact, give them an idea on how they can take it further.
* As a minor edit at last moment, I added link to the landing page with just a small line explain what I had done so far to make it crystal clear.

**Friday:**

* I plan to create the testing template today. I used the testing template used in one of my modules – Object Oriented Programming. I figured that was the best testing template that I could use for the project.
* The template has test cases, objective for each test case, the preconditions, test data, expected result and if the result is pass or fail as primary fields.
* My project already had sample test data as ‘sample.json’ and ‘data.json’ so there was no need to create additional test data.
* I performed several tests on my project and noted every data in the testing template. It was a xlsx file which I named ‘Testing Template’.
* All of my tests passed which gave me a huge sigh of relief.
* I created a new webpage – ‘testing.html’ to reroute to the testing template, added a link to this webpage on the landing page of my project. Also, I made it clear in the new webpage that the testing data is already present in the project files.

**Saturday:**

* For the final work on the project, in the ‘websummary.html’ page under ‘Further Research Models’, I added further information about how the modules correlate to the project.
* I did some research on the SimCity Glassbox Engine because honestly those were the only two good articles I could find that is trying what I’m trying with my project with the time frame. I compared how the SimCity Glassbox Engine is a bit similar to my project and how it’s also different from my project. The Glassbox Engine at its lowest level was separating different parts of the games to different object types, similar to how I was separating different components of a day for a particular place into different object types.
* This comparison should hopefully people get more information on how it is useful!
* Still unclear on whether I should add webpages to my GitHub repository as well but to not take any risks, I’ll still add them.