UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA
Denkleiers • Leading Minds • Dikgopolo tša Dihlalefi

COS 216 Practical Assignment 2

- Date Issued: **11 March 2024**
- Date Due: **8 April 2024** before **08:00**
- Submission Procedure: **Upload to the web server** (`wheatley`) **+ clickUP**
- This assignment consists of **5 tasks** for a total of **95 marks**.

1 Introduction

During this practical you will be creating a site to view property listings and get in contact with a property agent. The idea is to give users of the site the ability to look at different listings and see the characteristics of each listing, helping them make the right choice of which property to buy or rent. Users of the site can choose to view listings, view real-estate agents, and even calculate financial indicators to inform them if they can afford the property or not.

After successful completion of this assignment you should be able to create a web page which complies to the HTML5 and JavaScript Standards. The specific web page for this assignment will showcase the following functionality:

- Javascript animations
- Loading Screen
- Retrieving data from APIs
- Populating your templates with the retrieved API data
- Implementing the View listing page
- Implementing the Calculator page

2 Constraints

1. You must complete this assignment individually.
2. You may ask the Teaching Assistants for help but they will not be able to give you the solutions.
3. You must produce all of the source files yourself; you may not use any tool to generate source files or fragments thereof automatically.
4. Your assignment will be viewed using Chrome or Brave Web Browser (`https://brave.com/`) so be sure to test your assignment in this browser. Nevertheless, you should take care to follow published standards and make sure that your assignment works in as many browsers as possible.
5. You may utilise any text editor or IDE, upon an OS of your choice, again, as long as you do not make use of any tools to generate your assignment. (**This includes ChatGPT!!**)
6. All written code should contain comments including your name, surname and student number at the top of each file.
7. Your assignment must work on the `wheatley` web server, as you will be marked off there. However you are free and encouraged to **Develop** using a local tool like XAMP and later move over to wheatley however it must work off wheatley during marking.

8. **You must only use JavaScript for this assignment (no libraries with the exception of the point below).**
9. **You may use JQuery ONLY for DOM manipulation. Use of JQuery for AJAX calls will result in no marks for that section.**

3 Submission Instructions

You are required to upload all your source files (e.g. HTML5 documents, any images, etc.) to the web server (`wheatley`) and clickUP in a compressed (zip) archive. Make sure that you test your submission to the web server thoroughly. All the menu items, links, buttons, *etc.* must work and all your images must load. Make sure that your practical assignment works on the web server before the deadline. No late submissions will be accepted, so make sure you upload in good time. The server will not be accepting any uploads and updates to files from the stipulated deadline time until the end of the marking week (Thursday at 3pm).

The deadline is on Sunday but we will allow you to upload until Monday 8am. After this NO more submissions will be accepted. FTP to wheatley will also be disabled during this time.

Note, `wheatley` is currently available from anywhere. But do not rely that outside access from the UP network will always work as intended. You must therefore make sure that you `ftp` your assignment to the web server. Also make sure that you do this in good time. A snapshot of the web server will be taken just after the submission was due and only files in the snapshot will be marked.

Practicals are marked by demonstrating your practical to a tutor during the allotted marking weeks. **If you do not demonstrate your practical you will receive 0 for the practical**. You will only be marked in the practical session that you have booked on the cs portal, if you miss it, you will receive 0 (Unless special permissions have been granted by the lecturer. i.e. You were sick and able to provide a sick note).

4 Online resources

HTML5 - `https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5`
`http://www.w3schools.com/`

CSS - `http://www.w3.org/Style/Examples/011/firstcss`

Javascript - `https://www.w3schools.com/js/`
`https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/JavaScript_basics`

REST - `https://en.wikipedia.org/wiki/Representational_state_transfer`

JSON - `https://www.w3schools.com/js/js_json_syntax.asp`

HTTP Methods - `https://www.w3schools.com/tags/ref_httpmethods.asp`

AJAX - `https://www.w3schools.com/js/js_ajax_intro.asp`

5 Rubric for marking

API Functionality	
AJAX	12
JSON manipulation	4
Listings/Agencies have the correct image pulled from the API	4
Listing page	
Data population	5
Searchable	3
Select Buy or Rent	2
Filterable	5
Sort-able	2
Loading Screen	3
DOM manipulation	5
Agents Page	
Data population	5
Loading Screen	3
DOM manipulation	2
View Listing Page	
Correct property details displayed	5
Image Carousel	5
Additional Property Info	5
DOM Manipulation	5
Calculators page	
Financial Indicators Functional	10
DOM Manipulation	5
Validation	
HTML	3
JS	2
Upload	
Does not work on `wheatley`	**-20**
Not uploaded to clickUP	**-95**
Not Demoed	**-95**
Bonus	10
Total	**95**

6 Assignment Instructions

Task 1: Populate from APIs .... (55 marks)

This task requires you to pull information from a **Wheatley API** and populate all the templates (HTML layouts) you have created in PA 1 for the Listings and Agents page. You should retrieve data from the Wheatley API for at least 30 Listings and 10 Agencies.

All API calls must be done through JavaScript **AJAX XMLHttpRequest** (`https://www.w3schools.com/js/js_ajax_intro.asp`). You must choose between synchronous and asynchronous calls and provide reasons for your choice.
NB: Your reasoning should be included at the top of your JS file/s as a comment.

For this practical you will use an API Hosted off Wheatley. You must use this API and no others. The documentation and usage of said API can be found at `https://wheatley.cs.up.ac.za/api/doc.html`.

You will have to navigate to: `https://wheatley.cs.up.ac.za/api/generate.html`, to create your API key. You will have to use this key and your student number to use the API!

The following functionality needs to be implemented:

- You need to **replace your mock data** from PA 1 with data from the API that will load dynamically when the user views the page.
- You must find a way to make sure all listings and agencies have the correct corresponding picture.
- For all dynamic content you need to have a **loading screen/animation** to show that the data is being retrieved/processed.This can be a GIF/APNG/SVG loader for each listing/agency or a loading screen for the entire page (it will disappear once all the data on that page has loaded), Decide which is best. You can get resources from (`https://loading.io/`) however you are not limited to that resource.
- The following should also be completed on the listings page:
  - Search from the previous practical must be implemented using the API. Use the search bar to search for a specific location.
  - Buy or rent selector from previous practical must be implemented using the API.
  - Sort from the previous practical must be implemented using the API.
  - Filter from the previous practical must be implemented using the API.
  - You should also implement a way to indicate to the user if no results were found from a search and/or filter.

Task 2: Create the "View" page .... (20 marks)

The view listing page from the previous practical should be implemented. Have a look at the PA 1 specification to remind you what should be present on this page. Since the page should be implemented, remove mock data (if present) and check if the page works correctly. You will have to implement a method of your choice on the "Listings" page to view the details of that specific property. Examples can be by clicking on the image, having a more info button/icon they can click on.
It is important that even when the page refresh the information displayed do not change. The following is important for this page:

- Image carousel: You should be able to cycle through the images related to that property.
- The detailed description for the property.
- Two additional characteristics not on the "Listings" page.

Task 3: Create the "Calculators" page .... (15 marks)

The Calculators page from the previous practical should be implemented. Allow the user to enter the parameters of the indicator they want to calculate and display the result of their calculation. Remember two indicators are required. You can find good examples of formulas you can use at: `https://www.luxurypresence.com/blogs/real-estate-math-formulas/`. It will not be necessary to ask the user to input the "Prime Lending Rate", you can set it as **11.75**. You are welcome to do it like: `https://www.property24.com/calculators/bond`, where they set it as a default value for the input of the "Interest Rate".

Task 4: Validation .... (5 marks)

All HTML and JS should be valid and comply with standards.

Task 5: Bonus .... (10 marks)

Bonus marks will be given for additional functionality and awarded based on the difficulty level. These marks will only be awarded for notable additions (simply adding a background image is not enough).

The following, among others, may earn you extra marks:

- View property location on a map. Have a look at leaflet (You will be allowed to include the leaflet library).
- Calculate the amount of properties sold by each agency.
- Calculate a property score based on a certain criteria. Note: The complexity of your scoring algorithm will earn you your bonus marks.