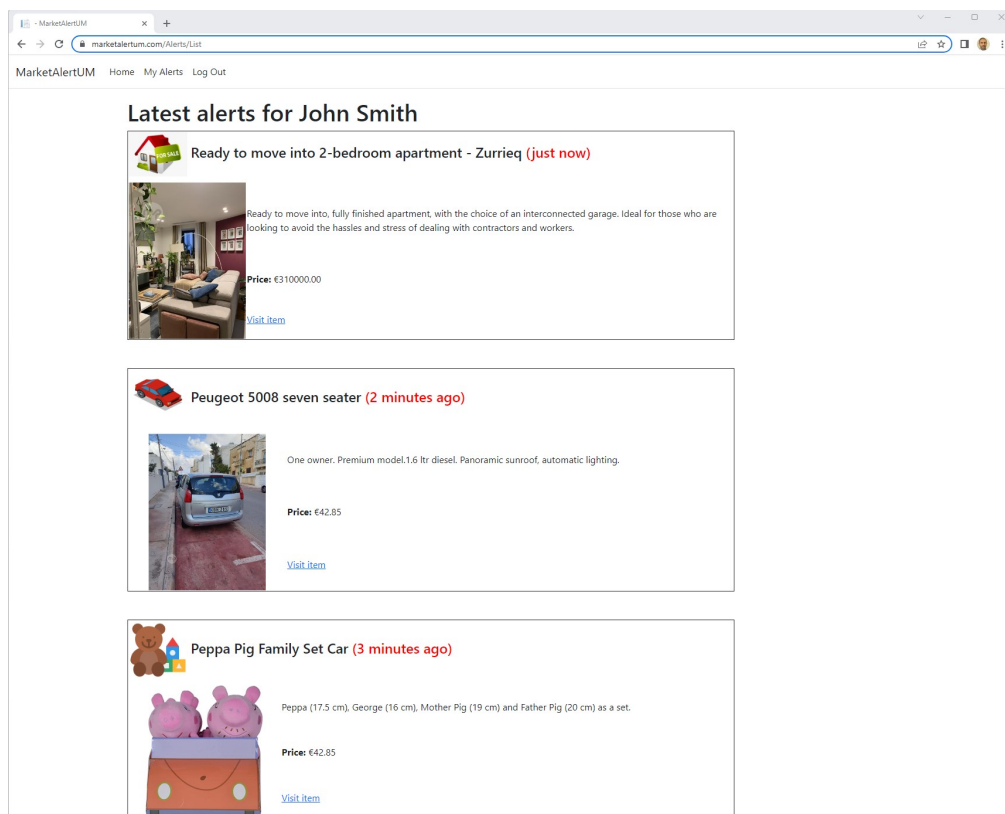


1 Instructions

- This is the first part of a three-part assignment.
- This assignment will be marked out of 40 and accounts for 40% of the study unit mark. It is estimated that it will take you between 20 and 25 hours to complete this assignment.
- The work is to be done on an *INDIVIDUAL BASIS*, not in groups.
- The assignment is to be submitted via VLE by noon of 15th November 2022.
- A random sample of students will be interviewed about their assignment to reduce the risk of collusion and/or plagiarism.
- Your submission should be a PDF file documenting your efforts and decisions. Code and video recordings should be made available via an appropriate repository (e.g. Github) and file sharing service (e.g. Google Drive) respectively.
- **IMPORTANT:** All questions regarding the assignment should be asked on the assignment forum in the VLE. Questions sent by email will be ignored. Before you ask a question, check if it has already been asked on the forum.

2 Market Alert UM Website

MarketAlertUM.com is a website that provides customised alerts to users when their favourite online stores or marketplaces list a product that may be of interest to them. You have been hired as a software engineer to help develop and test the system.



2.1 User Interface

The website can be accessed at <https://www.marketalertum.com>. The website allows you to log in, view alerts customised to you, and log out. Thihs is achieved as follows:

Logging In To log in, you need to click on the Log In link on the navigation bar at the top of the screen and enter your unique user ID.

Viewing Alerts Click on the My Alerts link in the navigation bar. If you are not logged in, you will be asked to do so. The alerts page shows you the **5 latest alerts**.

Logging Out To log out, you need to click on the Log Out link on the navigation bar at the top of the screen and enter your unique user ID.

2.2 REST API

As a developer, you have access to the REST API that controls the system's functioning. For the time being, there is only two API endpoints that are of interest to you.

2.2.1 Creating Alerts

To create an alert, you need to make a POST request to <https://api.marketalertum.com/Alert> and in the body include a JSON object as follows:

```
{
  "alertType": 6,
  "heading": "Jumper Windows 11 Laptop",
  "description": "Jumper Windows 11 Laptop 1080P Display,12GB RAM 256GB SSD",
  "url": "https://www.amazon.co.uk/Windows-Display-Ultrabook-Processor-Bluetooth",
  "imageUrl" : "https://m.media-amazon.com/images/I/712Xf2LtbJL._AC_SX679_.jpg",
  "postedBy": "7ca5f131-0ff0-42cd-85e8-cae25a4ee41f",
  "priceInCents": 24999
}
```

The alertType property indicates the type of alert being generated and can be one of the following:

```
Car = 1
Boat = 2
PropertyForRent = 3
PropertyForSale = 4
Toys = 5
Electronics = 6
```

The heading and description properties tell the website what heading and description to display about a product.

The url and imageUrl properties point to the original product website and an image of the product respectively.

The postedBy property should contain your unique user id whilst the priceInCents property contains the price of the item in cents.

If successful, the call will return a 201 status code and the response body will contain a representation of the newly created object:

```
{
  "id": "6f14f0e3-2d42-4beb-86b0-674127b1be29",
  "alertType": 6,
  "heading": "Jumper Windows 11 Laptop",
  "description": "Jumper Windows 11 Laptop 1080P Display,12GB RAM 256GB SSD",
  "url": "https://www.amazon.co.uk/Windows-Display-Ultrabook-Processor-Bluetooth",
  "imageURL": "https://m.media-amazon.com/images/I/712Xf2LtbJL._AC_SX679_.jpg",
  "postedBy": "7ca5f131-0ff0-42cd-85e8-cae25a4ee41f",
  "priceInCents": 24999,
  "postDate": "2022-10-11T21:38:54.3080651Z"
}
```

At this point the alert should appear on the website.

2.2.2 Purging Alerts

If you need to delete all your alerts, make a DELETE request to `https://api.marketalertum.com/Alert?userId=<userId>`, replacing `<userId>` with your own unique user ID.

3 Your Tasks

You have been hired as a software engineer to help develop and test the system. Two tasks are being asked from you at this stage.

3.1 Task 1: Develop a screen scraper to generate alerts

You are to:

1. Email `mark.micallef@um.edu.mt` and ask for your unique user ID. You will use this to log on and identify yourself with the system. Do not share it with anyone.
2. Select an e-commerce website of your choice and implement a program that visits the website, searches for a particular product (through search or navigation), screen scrapes the results and uploads 5 alerts to `marketalertum.com` using the REST API. You may use Webdriver or any other technology of your choice to perform screen scraping.
3. Ideally using a test-driven approach, you are to develop a unit testing suite that tests the functionality of your program being sure to test its behaviour for different possible behaviours by depended-on components (the e-commerce website and the REST API).
4. Document your efforts showing screenshots of coverage analysis, number of tests, etc. Document any key decisions you made regarding test doubles including the use of any specific test patterns. Be sure to state any assumptions you made and also explain why you did not manage to achieve 100% coverage, should that turn out to be the case.

3.2 Task 2: Web Test Automation

You have been asked to write a number of tests using Cucumber (or similar) and Webdriver (or similar) technologies. The following test scenarios are required.

Test 1: Valid Login

Given I am a user of marketalertum
When I login using valid credentials
Then I should see my alerts

Test 2: Invalid Login

Given I am a user of marketalertum
When I login using invalid credentials
Then I should see the login screen again

Test 3: Alert layout

Given I am an administrator of the website and I upload 3 alerts
Given I am a user of marketalertum
When I view a list of alerts
Then each alert should contain an icon
And each alert should contain a heading
And each alert should contain a description
And each alert should contain an image
And each alert should contain a price
And each alert should contain a link to the original product website

Test 4: Alert limit

Given I am an administrator of the website and I upload more than 5 alerts
Given I am a user of marketalertum

When I view a list of alerts
I should see 5 alerts

Test 5: Icon check

Given I am an administrator of the website and I upload an alert of type <alert-type>

Given I am a user of marketalertum

When I view a list of alerts

Then I should see 1 alerts

And the icon displayed should be <icon file name>

Alert Type	File name
1	icon-car.png
2	icon-boat.png
3	icon-property-rent.png
4	icon-property-sale.png
5	icon-toys.png
6	icon-electronics.png

You are to:

1. Develop a test suite using Cucumber (or similar) and Webdriver (or similar) technologies to implement the above tests.
2. Record a video of your final test suite, upload it to Google drive and share a link to it in your documentation.
3. Document your efforts explaining key design decisions that you made.
4. Comment on the testability of the website and make any recommendations you deem fit.

4 Dealing with issues

Due to the size of the class and in the interest of fairness, problems and queries related to the assignment should not be handled by email. Instead you are required to:

1. Visit the assignment forum on the VLE.
2. Search to make sure that no one else has asked a similar question.
3. Post your question on the forum.

It is in your best interest to monitor questions posted on the forum by your peers. The system and specification may change and **I will consider the forum as a living specification of the system** going forward.

5 Marking Scheme

This part of the assignment accounts for 40% of the marks for this study unit. This will be allocated as follows.

- Construction of screen scraper - **5 marks**
- Unit testing of screen scraper, including appropriate use of test doubles and justification of design decisions - **15 marks**
- Appropriate setup and use of Cucumber (or similar) technologies for Task 2 - **5 marks**
- Appropriate use of Webdriver (or similar) technologies including appropriate design decisions - **10 marks**
- Mature and informed commenting about the testability of the website - **5 marks**