# ITX 2000 Remote Hosts and Webservers

## COURSEWORK 2 REPORT

**Student ID: M00650922**

**Name: Shriya Narang**

**CAMPUS: Dubai Campus**

**Module Coordinator :**

**Ms. Engie Bashir**

# Project Report- StuBooks

StuBooks is a website catering to students of Middlesex University under the school of Engineering and Information Sciences. It provides the lists of all core and recommended reading material from each module and the links to the same.

In addition to that, for each book on StuBooks, reviews have been scraped using BeautifulSoup from GoodReads.com. The scrapings include reviewer's name, profile picture, review, rating and review date. The data scraped has been put in StuBooks with each book.

# Scraping Code:

## The scraping function:

```
def _scrape(link, book_id):
    i = 1
    page = requests.get(link)
    soup = BeautifulSoup(page.content, 'html.parser')
    allreviews = soup.find_all(class_="friendReviews elementListBrown")
    for eachreview in allreviews:
        reviewer = eachreview.find(class_="user").get_text()
        review = eachreview.find(class_="readable")
        img = eachreview.find("img")
        img = img["src"]
        rating = eachreview.find({'title': True, 'span':
True}).find_next_sibling().get_text()
        reviewDate = eachreview.find(class_="reviewDate createdAt
right").get_text()
        a = review.findChild().get_text()
        _updateReviewer(i, reviewer, img)
        _updateRating(book_id, rating, i)
        if len(a) > 1:
            print(reviewer)
            _updateReview(i, a, book_id, reviewDate)
        else:
            b = eachreview.find(class_="readable").get_text()
            print(reviewer)
            _updateReview(i, b, book_id, reviewDate)
        i += 1
    query = "select reviewer_name, img, review, date, rating from
reviewer, review, rating where
reviewer.reviewer_id=review.reviewer_id=rating.reviewer_id and
review.book_id='"+book_id+"';"
    cursor = db.cursor()
    cursor.execute(query)
    global pagecontent
    pagecontent= cursor.fetchall()
    if len(pagecontent)==0:
        flash('There are no reviews for selected book on GoodReads! Leave
your review on StuBooks!')
    else:
        print(pagecontent)
        cursor.close()
```

The above code is being used to scrape data from GoodReads.

As the link differs based on which book the user is looking at, the _scrape function takes the specified link and book_id and scraped data for that book.

## Using _scrape function:

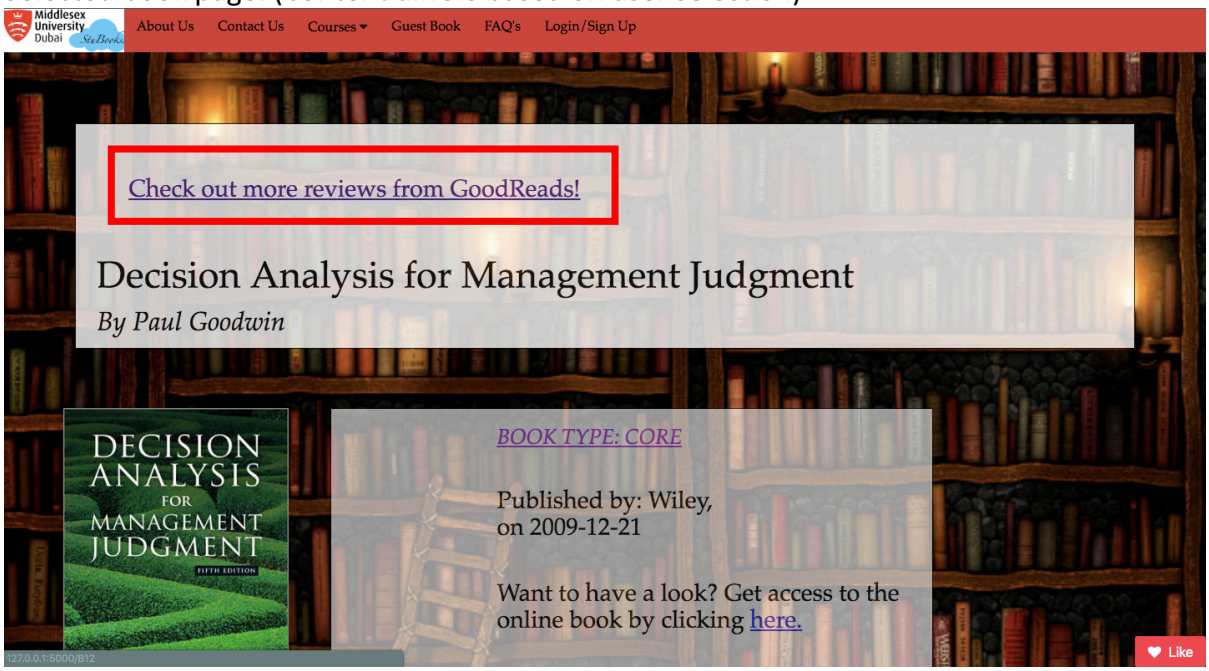
```
@app.route('/B1')
def B1():
    try:

_scrape("https://www.goodreads.com/book/show/845795.Practical_Packet_Analysis",
"B1")
        return render_template('reviews.html', pagecontent=pagecontent)
    except Exception as e:
            print("type error: " + str(e))
```

# Website Screenshots:

1. Home Page:

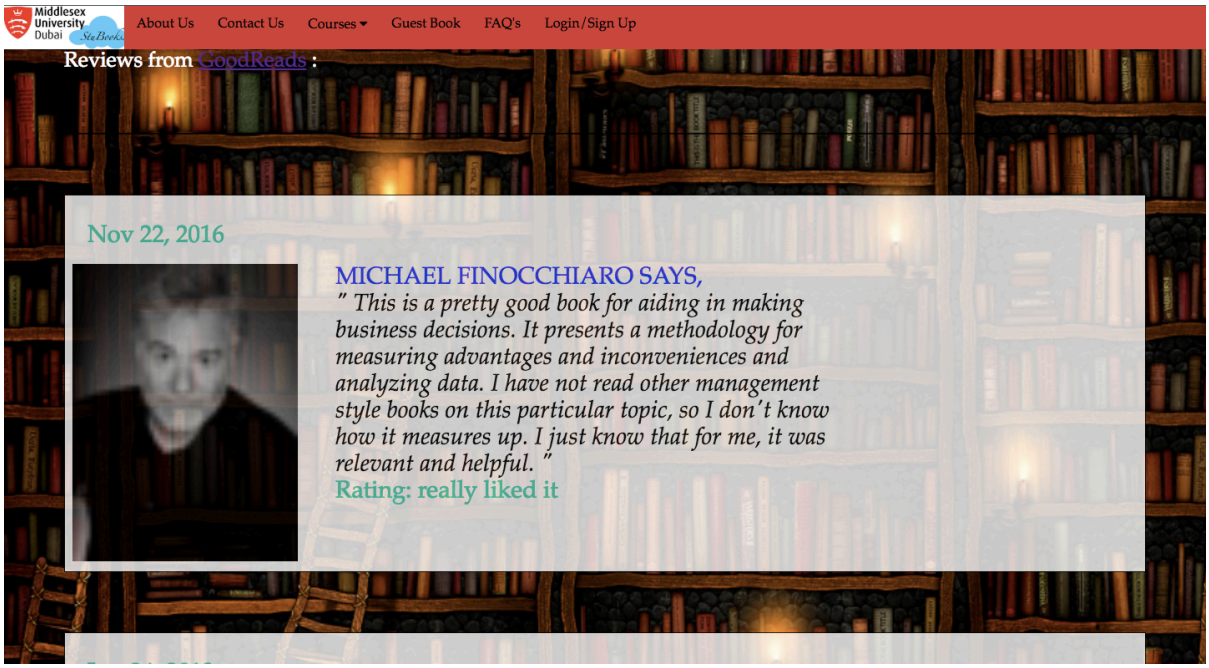
Home page displays the book of the month and the username of the user logged.

2. Selected book page: (Content differs based on user selection)

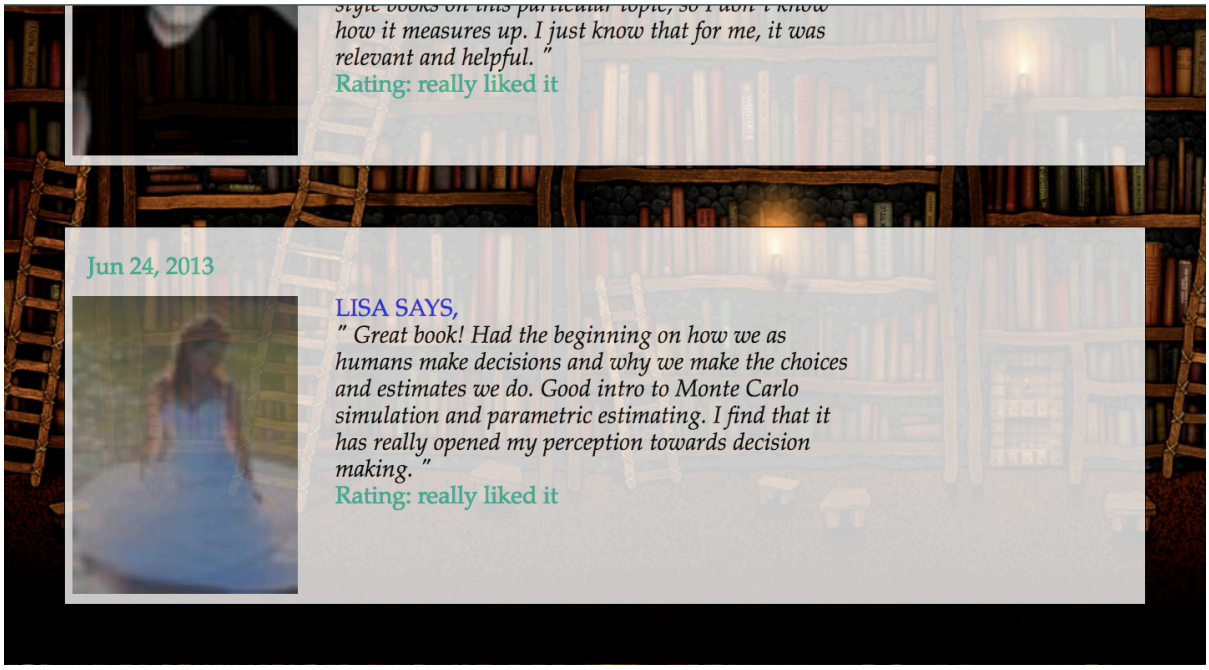



The book information and pictures and the StuBooks user reviews are extracted from the database. Reviews from GoodReads can be viewed by clicking on the link inside the red box.

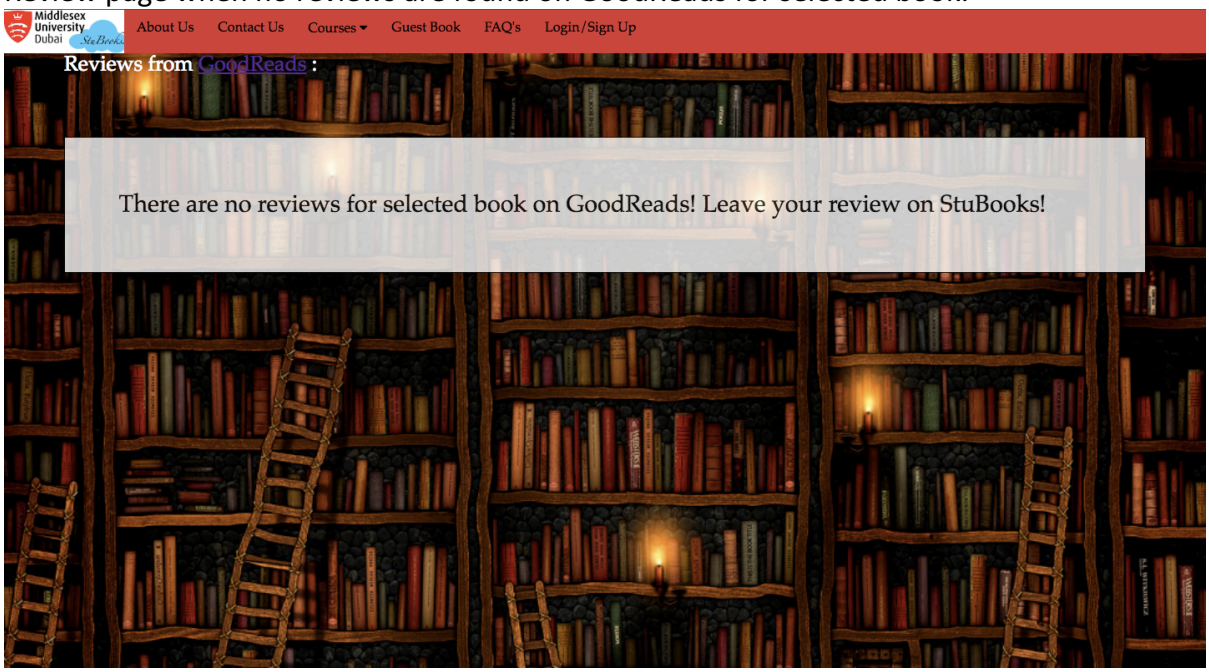
3. Reviews from GoodReads page: (Content differs based on user selection)




The review date, Reviewer name, reviewer picture, review and rating have been scraped from GoodReads, stored in the database and updated each time the page loads.



4. Review page when no reviews are found on GoodReads for selected book:




Every time the page loads, a counter function checks the number of records found. If no records are found, a flash message is displayed on the page.

# BIBLIOGRAPHY:

All information and pictures are taken from Amazon.

Amazon.com. (n.d.). *Amazon.com: Online Shopping for Electronics, Apparel, Computers, Books, DVDs & more*. [online] Available at: https://www.amazon.com [Accessed 20 Dec. 2018].

All reviews were scraped from Goodreads

*Goodreads* (2019). Available at: https://www.goodreads.com/ (Accessed: 19 March 2019).