



# Υπηρεσίες τον Υπολογιστικό Νέφος και την Ομίχλη (ΠΛΗ 513)

Αναφορά 1ης φάσης

Νοέμβριος 7, 2024

Δήμας Χρήστος, 2021030183  
cdimas@tuc.gr

# 1 Περιγραφή της εργασίας

Για την υλοποίηση της 1ης φάσης της εργαστηριακής εργασίας ζητήθηκε να υλοποιηθεί ένα eshop της επιλογής μας με full stack development (frontend - backend).

Το backend της εφαρμογής γράφτηκε στη γλώσσα προγραμματισμού **Python** και μέσω του **FlaskAPI** επιτεύχθηκε η επικοινωνία του με το frontend. Για αυτό το λόγο, δημιουργήθηκαν δύο φάκελοι με ονόματα **FlaskAPI** και **Order\_service** αντίστοιχα, και μέσα σε αυτούς δημιουργήθηκε python environment στο οποίο εγκαταστάθηκαν τα απαραίτητα πακέτα. Στον φάκελο **FlaskAPI**, βρίσκεται επίσης ο φάκελος **app** που περιέχει το **server.py**, ενώ στον αντίστοιχο φάκελο του **Order\_service** βρίσκεται το **main.py**. Τα εν λόγω αρχεία είναι υπεύθυνα για τα API requests, τα οποία είναι απαραίτητα για την επικοινωνία των βάσεων δεδομένων με το frontend.

Στο frontend της εφαρμογής χρησιμοποιήθηκε React. Δημιουργήθηκαν οι σελίδες όπως ζητήθηκαν από την εκφώνηση:

- Αρχική σελίδα
- Products
- Orders
- MyProducts
- Cart

Ακόμα έγιναν οι απαραίτητοι έλεγχοι για τη διασφάλιση της ομαλής λειτουργίας το Eshop, όπως ο έλεγχος διαθεσιμότητας για την ενέργεια Add To Cart και την ολοκλήρωση του Order, αλλά και ο έλεγχος για τις κατάλληλες τιμές στα πεδία των Products.

Οι βάσεις των προϊόντων και των παραγγελιών είναι **MongoDB**, των οποίων το ID χρησιμοποιείται για να διασφαλιστεί η μοναδικότητα του, που διαβάζουν απευθείας JSON. Επίσης, το καλάθι αποθηκεύεται με τη χρήση **Cookie** για να μην "χάνεται" με πιθανό refresh της σελίδας. Έχει δημιουργηθεί **docker-compose.yml** αρχείο που δημιουργεί τα απαραίτητα containers για την εφαρμογή. Υπάρχουν συνολικά 5 containers: ένα για το frontend service, ένα για το products service, ένα για το orders service και δύο για τις βάσεις δεδομένων.

Η εφαρμογή τρέχει κάνοντας unzip τον φάκελο, μετά ανοίγοντας το terminal στον φάκελο και τρέχοντας την εντολή **docker compose up --build**. Όταν ολοκληρωθεί η διαδικασία στον Docker, η εφαρμογή ανοίγει στο <http://localhost:5173>.