

Εργασία

Πληροφοριακά Συστήματα Παγκόσμιου Ιστού
Α.Π.Θ. Τμήμα Πληροφορικής
Εαρινό Εξάμηνο 2024-2025

Διδάσκον: Θοδωρής Τολιόπουλος (tatoliop@csd.auth.gr)

Σύνοψη εργασίας

Ο σκοπός της εργασίας είναι να χωριστείτε σε ομάδες **3-4 ατόμων** και να δημιουργήσετε ένα απλό eshop με προϊόντα μίας κατηγορίας της επιλογής σας.

Η εργασία αποτελείται από 4 ενότητες όπως περιγράφονται στον παρακάτω πίνακα, με ξεχωριστή αξιολόγηση σε κάθε ενότητα.

Ενότητα	Τίτλος	Μονάδες αξιολόγησης
A	Ανάπτυξη στατικών ιστοσελίδων	1
B	Ανάπτυξη REST API	1.5
Γ	Ανάπτυξη δυναμικής ιστοσελίδας	1.5
Δ	Δημιουργία docker images για deployment	1 (bonus)
Συνολικές μονάδες		4 + 1 bonus

Προθεσμία παράδοσης της εργασίας είναι **02/06/2025**.

Η εκπρόθεσμη παράδοση θα έχει ως αποτέλεσμα την αφαίρεση 10% της συνολικής βαθμολογίας για κάθε μέρα που περνάει, πχ . -10% μέχρι και 03/06/2025, -20% μέχρι και 04/06/2025, κλπ.

Αρχεία εργασίας: Υποβάλετε στο [elearning](#) ένα **zip** αρχείο με όνομα **AEM1-AEM2-AEM3-AEM4.zip** που θα περιλαμβάνει την υλοποίησή σας.

Ο κώδικας θα ελεγχθεί ως προς την πρωτοτυπία του, και κάθε εντοπισμός λογοκλοπής -αντιγραφής κώδικα συνεπάγεται τον μηδενισμό της βαθμολογίας.

Επικοινωνία: Όλη η επικοινωνία σχετικά με απορίες για την εργασία θα γίνεται μέσω του φόρουμ στην σελίδα του μαθήματος στο elearning. Για οτιδήποτε άλλο σχετικά με την εργασία μπορείτε να στείλετε email με το πρόθεμα WIS_25 στον διδάσκον.

Ενότητα Α - Ανάπτυξη στατικών ιστοσελίδων

Στην ενότητα Α καλείστε να αναπτύξετε δύο στατικές σελίδες, την [homepage.html](#) και την [products.html](#), που είναι ορατές σε έναν χρήστη, από τη δομή, το περιεχόμενο, μέχρι τη μορφοποίηση και τη βασική αλληλεπίδραση με μενού, κουμπιά και τα υπόλοιπα διαδραστικά στοιχεία. Οι τεχνολογίες που απαιτούνται για τη συγκεκριμένη υλοποίηση είναι η HTML5 και CSS.

Και στις 2 σελίδες θα πρέπει να υπάρχει ένα header με το logo που θα επιλέξετε και κουμπιά για πλοήγηση στις σελίδες και ένα footer με τα ονόματα των μελών της ομάδας. Στο homepage.html θα πρέπει να υπάρχει στην αρχή της σελίδας το όνομα του eshop και από κάτω ένα slideshow με εικόνες που θα επιλέξετε. Στην υπόλοιπη σελίδα μπορείτε να βάλετε ότι ταιριάζει στο eshop σας. Στην σελίδα products.html θα πρέπει να έχετε στο πάνω μέρος μία μπάρα αναζήτησης και από κάτω τη λίστα με τα προϊόντα σας, που θα περιλαμβάνουν μία εικόνα του προϊόντος, το όνομά του και μία σύντομη περιγραφή.

Ενότητα Β - Ανάπτυξη REST API

Στην ενότητα Β καλείστε να αναπτύξετε ένα Flask REST API, που περιλαμβάνει όλη την αλληλεπίδραση ενός server με μία βάση δεδομένων για την αναζήτηση και παρουσίαση αποθηκευμένων δεδομένων, εισαγωγή νέων δεδομένων αλλά και τη δυναμική δημιουργία περιεχομένου, χρησιμοποιώντας Python 3.10. Ως βάση δεδομένων θα χρησιμοποιήσετε τη MongoDB (ανήκει στην κατηγορία NoSQL βάσεων δεδομένων). Η δημιουργία του API θα γίνει σε ένα αρχείο με όνομα [app.py](#), χρησιμοποιώντας **μόνο και αποκλειστικά** τις βιβλιοθήκες: [Flask](#), [Flask-PyMongo](#), [Flask-Cors](#), [numpy](#).

Προσοχή! Στην βάση δεδομένων καλείστε να εισάγετε τουλάχιστον 20 προϊόντα του eshop σας, με τουλάχιστον τα εξής πεδία:

- Όνομα προϊόντος
- Φωτογραφία
- Σύντομη περιγραφή
- Likes

Καθώς και ότι άλλο πεδίο θεωρείτε σημαντικό για την υλοποίησή σας.

Τα **endpoints** που θα αναπτύξετε είναι:

- **/search**: Αναπτύξτε ένα GET request που θα δέχεται μία παράμετρο με όνομα name με τύπο string και θα αναζητάτε ένα προϊόν στη βάση με βάση το όνομα και θα σας επιστρέφει το προϊόν ή τα προϊόντα. Σε περίπτωση που βρει περισσότερα από ένα προϊόν τότε τα προϊόντα που θα επιστρέφει θα πρέπει να είναι κατά φθίνουσα σειρά ταξινομημένα με βάση τη τιμή. Το request θα επιστρέφει μία λίστα με όλα τα προϊόντα που βρήκε σε μορφή JSON. Αν δεν βρει κανένα προϊόν τότε το endpoint θα επιστρέφει

κενή λίστα. Για παράδειγμα αν αναζητήσετε ένα προϊόν με όνομα Paper, τότε η αναζήτηση θα πρέπει να επιστρέφει όλα τα προϊόντα που έχουν τη λέξη Paper στο όνομα, π.χ. Paper A4, Paper A3. Ωστόσο αν αναζητήσετε ένα προϊόν με όνομα Paper A3, τότε η αναζήτηση πρέπει να επιστρέφει μόνο αυτό το προϊόν. Σε περίπτωση που η παράμετρος της αναζήτησης είναι ένα κενό string, τότε θα πρέπει να επιστρέφει όλα τα προϊόντα του eshop.

Προσοχή: Στο collection της βάσης που θα έχετε τα προϊόντα πρέπει να φτιάξετε έναν Index στο πεδίο name με τύπο text (θα εξηγηθεί στο εργαστήριο για τις NoSQL βάσεις δεδομένων).

- **/like:** Αναπτύξτε ένα POST request που θα δέχεται μία παράμετρο με το id ενός προϊόντος και θα ενημερώνει το πεδίο **Likes** του προϊόντος προσθέτοντας τον αριθμό 1 στην υπάρχουσα τιμή.
- **/popular-products:** Αναπτύξτε ένα GET request που θα επιστρέφει μία λίστα με τα top-5 πιο δημοφιλή προϊόντα με βάση των αριθμό των **Likes**.

Η πρόσβαση μεταξύ Flask και MongoDB θα γίνει **αποκλειστικά** με την βιβλιοθήκη Flask-PyMongo. Για να αποφύγετε προβλήματα σχετικά με την πολιτική CORS, θα χρειαστεί να ενεργοποιήσετε την CORS policy στο Flask API με την βιβλιοθήκη Flask-Cors. Το Flask API πρέπει να τρέχει στην IP 127.0.0.1 (localhost) και στη PORT 5000, δηλ. <http://127.0.0.1:5000>. Κάθε endpoint βαθμολογείται ξεχωριστά σύμφωνα με τις μονάδες που αναγράφονται στην επιμέρους βαθμολογία.

Ενότητα Γ - Ανάπτυξη δυναμικής ιστοσελίδας

Στην ενότητα Γ καλείστε να συνδέσετε την υλοποίηση σας στην ενότητα Α και Β χρησιμοποιώντας JavaScript, ώστε να αναπτύξετε μία δυναμική ιστοσελίδα. Η JavaScript επιτρέπει τη δημιουργία αλληλεπιδράσεων μεταξύ μιας στατικής σελίδας HTML και ενός REST API. Πιο συγκεκριμένα θα χρειαστεί να υλοποιήσετε τις παρακάτω αλληλεπιδράσεις σε ένα αρχείο JavaScript με όνομα products.js:

- Αλληλεπίδραση 1: Όταν ο χρήστης αναζητήσει ένα προϊόν (στην μπάρα αναζήτησης της σελίδας products.html), η ιστοσελίδα πρέπει να κάνει ένα GET request στο endpoint /search που αναπτύξατε στην ενότητα Β χρησιμοποιώντας το κείμενο που υπάρχει στο πλαίσιο αναζήτησης. Όταν ολοκληρωθεί το request τα αποτελέσματα της αναζήτησης πρέπει να εμφανίζονται στο πίνακα που υπάρχει στην products.html σελίδα. Σε κάθε νέα αναζήτηση τα αποτελέσματα της προηγούμενης πρέπει να σβήνονται από τον πίνακα.
- Αλληλεπίδραση 2: Όταν ο χρήστης κάνει αριστερό κλικ πάνω στη φωτογραφία ενός προϊόντος από την σελίδα products.html, θα πρέπει να γίνεται ένα POST request στο endpoint /like για την ενημέρωση των likes του συγκεκριμένου προϊόντος.
- Αλληλεπίδραση 3: Στο slideshow της σελίδας homepage.html θα πρέπει να φαίνονται οι φωτογραφίες και τα ονόματα των προϊόντων που επιστρέφονται από το GET request στο endpoint /popular-products.

Προσοχή: Για το POST request θα πρέπει να ορίσετε το "Content-Type" header σε "application/json".

Ενότητα Δ - Δημιουργία docker images για deployment (προαιρετικά)

Προαιρετικά μπορείτε μαζί με την υλοποίησή σας να συμπεριλάβετε και τα αρχεία Dockerfiles καθώς και το docker-compose.yml με το οποίο μπορεί να γίνει deploy η εφαρμογή σας. Θα πρέπει να περιλαμβάνει 3 αρχεία Dockerfile:

1. Τον web server με τα αρχεία html, css και js
2. Το python application του Flask
3. Την MongoDB