

ΜΥΕ042 Τεχνολογίες Διαδικτύου

Ανακοίνωση: Παρασκευή, 8 Νοεμβρίου, Παράδοση: Παρασκευή, 29 Νοεμβρίου στις 21:00

Εργαστηριακή Άσκηση 1

Εφαρμογή ιστού για διαμοιρασμό φωτογραφιών στο Ruby-on-Rails

1. Εισαγωγή

Μια εφαρμογή ιστού κοινοχρησίας φωτογραφιών επιτρέπει στους χρήστες να ανοίξουν λογαριασμό και στη συνέχεια να εισέλθουν στην εφαρμογή προκειμένου να ανεβάσουν φωτογραφίες στον διακομιστή και να τις κάνουν προσβάσιμες σε άλλους χρήστες. Υποθέτουμε ότι οι λογαριασμοί είναι δημόσιοι. Κάθε χρήστης μπορεί να επιλέξει συγκεκριμένους χρήστες που επιθυμεί να ακολουθεί. Όταν ένας ακολουθούμενος χρήστης δημοσιεύει μια φωτογραφία, ο χρήστης που τον ακολουθεί βλέπει την φωτογραφία την επόμενη φορά που ανανεώνει την σελίδα του στην εφαρμογή. Οι φωτογραφίες του χρήστη και των ακολουθούμενων χρηστών εμφανίζονται σε αντίστροφη χρονολογική σειρά δημοσίευσης. Ένας χρήστης μπορεί να προσθέσει σχόλια σε μια φωτογραφία που δημοσιεύει ή σε φωτογραφίες των χρηστών που ακολουθεί. Τέλος, ένας χρήστης μπορεί να διαγράψει μια φωτογραφία που προηγουμένως δημοσίευσε με όλα τα σχόλιά της.

2. Προετοιμασία

2.1 Εκτέλεση της εικονικής μηχανής

Κατεβάστε την εικονική μηχανή [MYE042L1.zip](#) (1GB) και αποσυμπίεστε την στον τοπικό σας δίσκο (ή σε μνήμη USB με τουλάχιστο 13GB διαθέσιμο χώρο). Η εικονική μηχανή έχει εγκατεστημένη την διανομή Bookworm (v12.7) του Debian 64-bit Linux. Χρησιμοποιήστε πρόσφατη έκδοση του VMWare Workstation (π.χ., [VMware Workstation v17 για Windows](#) ή [Linux](#)) για να τρέξετε την εικονική μηχανή. Η μηχανή είναι ρυθμισμένη να χρησιμοποιήσει 2 πυρήνες και 2GB RAM, αλλά αυτές τις ρυθμίσεις μπορείτε να τις αλλάξετε μέσα από το μενού Settings της εικονικής μηχανής. Για να εκμεταλλευτείτε το γραφικό παραθυρικό περιβάλλον της μηχανής, εξασφαλίστε ότι έχετε βάλει την εικονική μηχανή σε κατάσταση πλήρους οθόνης πιέζοντας το αντίστοιχο εικονίδιο πάνω αριστερά. Μπορείτε να εισέλθετε ως απλός χρήστης με το όνομα mye042 (και το ίδιο ως συνθηματικό). Σε περίπτωση που χρειαστεί να εγκαταστήσετε επιπλέον πακέτα στο σύστημα Linux, μπορείτε να εισέλθετε ως χρήστης root (με συνθηματικό mye042) και να τρέξετε στην γραμμή εντολής `apt install <package>`. Η εικονική μηχανή είναι προετοιμασμένη με εγκατεστημένη τη γλώσσα Ruby (έκδοση 2.6.6) και τον φυλλομετρητή ιστού Firefox. Για διευκόλυνσή σας, μπορείτε να κάνετε αντιγραφή-επικόλληση (copy-paste) κείμενο ή αρχεία μεταξύ του λειτουργικού συστήματος επισκέπτη (guest) που τρέχει στην εικονική μηχανή (Debian Linux) και του λειτουργικού συστήματος φιλοξενίας (host), για παράδειγμα, προκειμένου να πάρετε τα αρχεία που αλλάξετε και να τα υποβάλλετε (turnin) για την εξέταση.

2.2 Η εφαρμογή ιστού rottenpotatoes

Στον κατάλογο `~mye042/src/rottenpotatoes` υπάρχει μια λειτουργική έκδοση της εφαρμογής rottenpotatoes που χρησιμοποιήσαμε στις διαλέξεις ως παράδειγμα. Ανοίξτε ένα τερματικό και αλλάξτε τον τρέχοντα κατάλογο (με `cd`) σε `~mye042/src/rottentpotatoes`. Ο κατάλογος αυτός είναι γνωστός και ως κατάλογος ρίζα της εφαρμογής. Εκεί εκτελέστε την εντολή `rails server` (ή `rails s`) για να ξεκινήσετε την εφαρμογή rottenpotatoes. Μετά εκκινήστε τον φυλλομετρητή Chrome, επιλέγοντας από την επιλογή Internet στο μενού Applications πάνω αριστερά. Εισάγετε τη διεύθυνση <http://localhost:3000> για να συνδεθείτε με την εφαρμογή. Η εφαρμογή επιτρέπει τον χρήστη να δει τη λίστα με τις υπάρχουσες ταινίες, να εμφανίσει την περιγραφή μιας ταινίας, να αλλάξει τις σχετικές

πληροφορίες, να προσθέσει νέες εγγραφές, και να διαγράψει μια ταινία. Ανά πάσα στιγμή, μπορείτε να τερματίσετε τον διακομιστή, πιέζοντας Ctrl-C στο τερματικό που τρέχει ο διακομιστής.

Στον κατάλογο ρίζα (`~mye042/src/rottenpotatoes`) θα βρείτε τον κατάλογο `app` που περιέχει τον κώδικα της εφαρμογής. Μπορείτε να δείτε τις διαδρομές (`routes`) της εφαρμογής εκτελώντας την εντολή `rake routes` στον κατάλογο ρίζα. Εξοικειωθείτε με το μοντέλο της εφαρμογής που εμφανίζεται στον κατάλογο `models`, τους ελεγκτές που εμφανίζονται στον κατάλογο `controllers` και τις προβολές που εμφανίζονται στον κατάλογο `views`. Η εφαρμογή χρησιμοποιεί πρότυπες προβολές γραμμένες σε γλώσσα `HamI` για να προετοιμάσει το αρχείο `HTML` που επιστρέφει σε μία αίτηση `HTTP` (αρχεία με επέκταση `.hamI` που περιέχονται στις προβολές). Η εφαρμογή χρησιμοποιεί τη βάση δεδομένων `sqlite3` (καθορίζεται στο αρχείο `db/database.yml`) αρχικοποιημένη με μια λίστα ταινιών, όπως καθορίζεται στο αρχείο `db/seeds.rb`. Οι ιδιότητες του πίνακα `movies` καθορίζονται στο αντίστοιχο αρχείο `Ruby` του καταλόγου `db/migrate`.

2.3 Η εφαρμογή ιστού `treegram`

Στον κατάλογο `~mye042/src/treegram` θα βρείτε τον κώδικα της εφαρμογής `treegram`. Η `treegram` είναι μια πολύ απλή εφαρμογή κοινοχρησίας φωτογραφιών που επιτρέπει σε έναν χρήστη να ανοίξει λογαριασμό, να εισέλθει και να ανεβάσει φωτογραφίες. Κάτω από μια φωτογραφία, υπάρχει ένα μενού με όλους τους εγγεγραμμένους χρήστες από τους οποίους ο χρήστης μπορεί να επιλέξει έναν για να τον προσθέσει ως ετικέτα (`tag`) στην φωτογραφία. Σταματήστε την εφαρμογή `rottenpotatoes` και ξεκινήστε την εφαρμογή `treegram` μέσα από τον κατάλογο ρίζας της. Μπορείτε να συνδεθείτε με την εφαρμογή εισάγοντας στον φυλλομετρητή την διεύθυνση <http://localhost:3000> όπως και πριν. Η εφαρμογή ορίζει τους πίνακες `photos`, `users` και `tags` σύμφωνα με τον κώδικα στο αρχείο `db/schema`, ενώ τα αντίστοιχα μοντέλα `photo`, `user` και `tag` υλοποιούνται στα αρχεία `Ruby` του καταλόγου `app/models`. Ο κώδικας της εφαρμογής είναι αποθηκευμένος στον κατάλογο `app/controllers`. Πειραματιστείτε με την εφαρμογή δημιουργώντας λογαριασμούς για μερικούς χρήστες και φορτώνοντας φωτογραφίες. Τα email των χρηστών δεν χρειάζεται να είναι πραγματικά (απλώς να μοιάζουν με email) ενώ μπορείτε να πειραματιστείτε με μια ενδεικτική φωτογραφία που υπάρχει στον κατάλογο `app/assets/images`. Όποτε εισέρχεστε ως ένας χρήστης, θα δείτε τις φωτογραφίες που ανέβασε ο συγκεκριμένος χρήστης και θα βρείτε ένα κουμπί πάνω δεξιά για να προσθέσετε μια νέα φωτογραφία και ένα άλλο κουμπί για να βγείτε από τον λογαριασμό.

3. Άσκηση

Σας ζητείται να επεκτείνετε την εφαρμογή `treegram` με λειτουργίες που επιτρέπουν σε έναν χρήστη να προσθέσει τίτλο σε μια φωτογραφία, να ακολουθεί άλλους χρήστες, και να δει τις φωτογραφίες που ανεβάστηκαν από τους ακολουθούμενους χρήστες. Η υλοποίηση θα γίνει με χρήση της γλώσσας `Ruby`, των βοηθητικών αρχείων και μεθόδων του `Rails`, και της γλώσσας `HamI`.

1. **Προσθέστε τίτλο σε μια φωτογραφία όταν την ανεβάζετε στον διακομιστή.** Θα χρειαστεί να επεκτείνετε το μοντέλο `photo` και τον αντίστοιχο πίνακα `photos` της βάσης δεδομένων με μια ιδιότητα `title` τύπου `string`. Επιπλέον, επεκτείνετε την φόρμα που χρησιμοποιείται για το ανέβασμα προκειμένου να συμπεριλάβετε τον τίτλο και να τον αποθηκεύσετε στον αντίστοιχο πίνακα. Όταν προβάλλονται οι φωτογραφίες ενός χρήστη, ο τίτλος θα εμφανίζεται κάτω από τη φωτογραφία.
2. **Επιτρέψτε στους χρήστες να καθορίσουν χρήστες που ακολουθούν.** Θα προσθέσετε ένα μοντέλο `follow` και έναν αντίστοιχο πίνακα που καθορίζει τα αναγνωριστικά του ακολουθητή (`follower`) χρήστη και των ακολουθούμενων χρηστών (`followed`). Θα προσθέσετε κουμπί για να εμφανίζονται όλοι οι εγγεγραμμένοι χρήστες. Δίπλα στο όνομα του κάθε χρήστη θα εμφανίζεται κουμπί

που επιτρέπει την επιλογή ακολούθησης. Όταν εμφανίζετε τις φωτογραφίες ενός χρήστη θα συμπεριλάβετε τις φωτογραφίες των ακολουθούμενων χρηστών με τον τίτλο της καθεμίας από κάτω. Οι φωτογραφίες θα πρέπει να είναι ταξινομημένες σε αντίστροφη χρονολογική σειρά της ώρας που φορτώθηκαν στο σύστημα.

3. **Επιτρέψτε τους χρήστες να προσθέσουν σχόλια σε φωτογραφίες.** Προσθέστε μοντέλο *comment* και αντίστοιχο πίνακα βάσης δεδομένων που να περιέχουν το αναγνωριστικό της φωτογραφίας, το αναγνωριστικό του χρήστη που σχολίασε την φωτογραφία, και το κείμενο του σχολίου. Όταν εμφανίζετε τις φωτογραφίες που μπορεί να δει ένας χρήστης (δικές του ή ακολουθούμενων), θα προσθέσετε μία φόρμα (ή κουμπί) δίπλα σε κάθε φωτογραφία που επιτρέπει τον χρήστη να προσθέσει σχόλιο στην φωτογραφία αυτή.
4. **Επιτρέψτε τους χρήστες να διαγράψουν σχόλια.** Ένας χρήστης επιτρέπεται να διαγράψει στις δικές του φωτογραφίες τα σχόλια που έχει προσθέσει ο ίδιος ή άλλοι χρήστες, ενώ στις φωτογραφίες των άλλων χρηστών μόνο τα σχόλια που έχει προσθέσει ο ίδιος. Προσθέστε κουμπί διαγραφής δίπλα σε κάθε σχόλιο που ένας χρήστης επιτρέπεται να διαγράψει και υλοποιήστε τη λειτουργία διαγραφής.

Για καθένα από τα παραπάνω χαρακτηριστικά, θα πρέπει να περιγράψετε τα πρόσθετα μοντέλα και πίνακες, με τον κώδικα στους ελεγκτές και τις προβολές της εφαρμογής.

4 Τί θα παραδώσετε

Θα ετοιμάσετε την λύση ατομικά ή σε ομάδες των δύο ατόμων. Η εκπρόθεσμη υποβολή μειώνει τον βαθμό κατά 10% ανά ημέρα μέχρι 50%. Υποβάλετε την λύση σας εγκαίρως με την εντολή **turnin lab1_24@mye042 README.pdf file1 ...** Το αρχείο **README.pdf** περιέχει το ονοματεπώνυμό σας, το ΑΜ και μια περιγραφή των μοντέλων, ελεγκτών, προβολών, και πινάκων που προσθέσατε. Εξηγήστε σε λογικό επίπεδο λεπτομέρειας τα βήματα που ακολουθήσατε κατά την υλοποίηση της λύσης καθώς και τις δοκιμές που πραγματοποιήσατε προκειμένου να επιβεβαιώσετε ότι η λύση σας δουλεύει σωστά. Το κείμενο της περιγραφής θα πρέπει να είναι γραμμένο στα ελληνικά και να είναι ευανάγνωστο. Υποβάλετε όλα τα αρχεία του πηγαίου κώδικα που προσθέσετε ή αλλάξατε. Η λύση σας θα πρέπει να μπορεί να εκτελεστεί στην εικονική μηχανή που σας δίνεται.

5 Λειτουργία για επιπλέον βαθμό (10%)

Για επιπλέον 10% στον βαθμό της άσκησης (μέγιστο 110/100), προσθέστε δίπλα σε κάθε φωτογραφία κουμπί που να επιτρέπει τον χρήστη που την ανέβασε να τη διαγράψει μαζί με όλα τα σχόλια και ετικέτες της.