

Συνοπτική Περιγραφή της Διαδικτυακής Εφαρμογής

M102 – Εφαρμογή Αναζήτησης Video με το API του YouTube

Στα πλαίσια της εργασίας για το μάθημα M102, δημιουργήθηκε μια διαδικτυακή εφαρμογή στην οποία κάποιος μπορεί να αναζητήσει και να αναπαράγει video που είναι ανεβασμένα στο Youtube (<https://www.youtube.com>) .

Για την αναζήτηση των αποτελεσμάτων χρησιμοποιείται το αντίστοιχο API του Youtube, με κλήσεις POST μέσω AJAX, όπως θα δούμε πιο αναλυτικά παρακάτω.

Η εφαρμογή έχει ανέβει και λειτουργεί στον υπερσύνδεσμο <https://sotos.re/ourtube>

Περιέχει τις σελίδες login.html για σύνδεση και δημιουργία λογαριασμού και την searchPage.html για την αναζήτηση video. Η σύνδεση στην εφαρμογή δεν είναι υποχρεωτική, μπορεί δηλαδή ο χρήστης απευθείας και χωρίς να είναι συνδεδεμένος να αναζητήσει το video που επιθυμεί και στην συνέχεια να το αναπαράγει.

Μέσω της διαδικτυακής αυτής εφαρμογής οι χρήστες θα μπορούν να αναζητούν βίντεο της αρεσκείας τους με γρήγορο και αποτελεσματικό τρόπο. Επιπλέον, οι χρήστες θα έχουν την δυνατότητα να ορίζουν κριτήρια αναζήτησης, όπως τη διάρκεια των βίντεο, το πλήθος των αποτελεσμάτων που θα εμφανίζονται ανά σελίδα και το είδος ταξινόμησης/εμφάνισης των αποτελεσμάτων. Έπειτα από την αναζήτηση και εύρεση των βίντεο παρουσιάζονται χρήσιμες πληροφορίες για κάθε ένα από αυτά, όπως περιγραφή, διάρκεια, πλήθος προβολών κ.ά. Για την προβολή του κάθε βίντεο οι χρήστες επιλέγουν είτε το thumbnail, είτε τον τίτλο του και στη συνέχεια προβάλλεται το βίντεο στην ίδια καρτέλα. Προκειμένου να μην υπάρχουν ανακατευθύνσεις ή πολύπλοκες διαδικασίες για την προβολή των βίντεο, χρησιμοποιείται το FancyBox.

Για την χρήση της διαδικτυακής εφαρμογής δεν απαιτείται σύνδεση στο σύστημα. Ωστόσο εάν οι χρήστες επιλέξουν να συνδεθούν στο σύστημα, είτε δημιουργώντας έναν νέο λογαριασμό, είτε χρησιμοποιώντας τον ήδη υπάρχον λογαριασμό που διαθέτουν στο Facebook, θα έχουν πρόσβαση σε κάποιες επιπλέον λειτουργίες.

Για τη σχεδίαση και την ανάπτυξη της εφαρμογής αυτής έχουν ληφθεί υπόψη οι συνηθέστερες ανάγκες και απαιτήσεις των χρηστών, όπως είναι:

- Η ευκολία στη χρήση.
- Η γρήγορη αναζήτηση.
- Ο σύντομος χρόνος απόκρισης της εφαρμογής.
- Η ακεραιότητα στα αποτελέσματα.
- Η αδιάλειπτη λειτουργία της εφαρμογής.
- Η δυνατότητα αναζήτησης βάσει των διαφορετικών χαρακτηριστικών των βίντεο, π.χ. τη διάρκειά τους.
- Η ταξινόμηση και η εμφάνιση των αποτελεσμάτων με βάση την επιθυμία των χρηστών, π.χ. με βάση τη σχετικότητα, την ημερομηνία, την βαθμολογία, το πλήθος προβολών.
- Η απλή και εύκολη πλοήγηση.
- Η σύγχρονη και αισθητικά ευχάριστη εμφάνιση της εφαρμογής.
- Η προαιρετική είσοδος στο σύστημα για την χρήση των υπηρεσιών.
- Η δυνατότητα εισόδου στο σύστημα με διάφορους τρόπους, όπως π.χ. μέσω της δημιουργίας νέου λογαριασμού ή με τη χρήση κάποιου ήδη υπάρχοντος λογαριασμού (π.χ. Facebook).
- Η προστασία των προσωπικών στοιχείων των χρηστών.
- Η δυνατότητα να υλοποιούνται όλες οι λειτουργίες στο ίδιο παράθυρο, ώστε να μην υπάρχουν ανακατευθύνσεις σε άλλες καρτέλες ή σε μη έμπιστους εξωτερικούς συνδέσμους.
- Η ικανοποιητική υποστήριξη και η μεγάλη αποκρισιμότητα της εφαρμογής σε όλα τα μεγέθη οθονών και τους διάφορους browsers.
- Επιπλέον, πέρα από όλα όσα αφορούν τους χρήστες, ορισμένες εξίσου σημαντικές απαιτήσεις που έχουν ληφθεί υπόψη για την αποτελεσματικότητα της εφαρμογής είναι οι εξής:
- Η εφαρμογή να λειτουργεί και να συμπεριφέρεται σύμφωνα με τις καταγεγραμμένες απαιτήσεις χρηστών.

- Εύκολη τροποποίηση της εφαρμογής.
- Δυνατότητα επαναχρησιμοποίησης και επεκτασιμότητας.

Αντίστοιχες Υπηρεσίες

Μερικές από τις πιο διαδεδομένες διαδικτυακές εφαρμογές αναζήτησης βίντεο είναι οι εξής:

YouTube

Link: <https://www.youtube.com/>

Το YouTube είναι ο πιο συχνά επισκεπτόμενος ιστότοπος για την αναζήτηση βίντεο και δημιουργήθηκε το 2005. Επιτρέπει στους χρήστες να ανεβάζουν, να προβάλλουν, να βαθμολογούν και να μοιράζονται βίντεο. Προσφέρει μια μεγάλη ποικιλία από ερασιτεχνικά και επαγγελματικά βίντεο.

Vimeo

Link: <https://vimeo.com/>

Το Vimeo είναι μία ταχέως αναπτυσσόμενη πλατφόρμα, η οποία ιδρύθηκε το 2004 και το 2007 έγινε η πρώτη διαδικτυακή εφαρμογή που επέτρεπε τους χρήστες να ανεβάζουν βίντεο σε HD ποιότητα. Εστιάζει κυρίως σε περιεχόμενο υψηλής ποιότητας, ωστόσο διαθέτει και μεγάλη ποικιλία από user-generated βίντεο.

DailyMotion

Link: <http://www.dailymotion.com/>

Είναι ένας γαλλικός ιστότοπος που επιτρέπει τους χρήστες να παρακολουθούν και να ανεβάζουν βίντεο. Η αναζήτηση γίνεται με τη χρήση των tags, κατηγοριών, καναλιών κ.ά. Το upload των βίντεο περιορίζεται σε 2GB μέγεθος, σε διάρκεια 60' λεπτών και σε ποιότητα HD-Ready (1289-720p).

Metacafe

Link: <http://www.metacafe.com/>

Το Metacafe είναι ένας από τους με μεγαλύτερους διαδικτυακούς τόπους αναζήτησης και προβολής βίντεο σύντομου περιεχομένου. Περιλαμβάνει αποκλειστικά και

πρωτότυπα βίντεο διαφόρων κατηγοριών (όπως π.χ. ψυχαγωγίας κ.ά.).

Veoh

Link: <http://www.veoh.com/>

Το Veoh, που περιγράφεται ως “Internet TV company”, είναι μία πλατφόρμα αναζήτησης βίντεο. Διαθέτει εκατομμύρια βίντεο, τόσο ερασιτεχνικά, όσο και επαγγελματικά, εστιάζοντας, ωστόσο, κυρίως στην δεύτερη κατηγορία. Οι χρήστες έχουν τη δυνατότητα να ανεβάζουν βίντεο οποιασδήποτε διάρκειας και με διαφορετικές μορφές αρχείων.

Η Εφαρμογή

Η κεντρική σελίδα της εφαρμογής (<https://sotos.re/ourtube/searchPage.html>) περιέχει αρχικά μόνο ένα πεδίο για την αναζήτηση του video και τρεις επιλογές για τα αποτελέσματα τα οποία θα επιστραφούν.

Επιλογές Αναζήτησης

Στον χρήστη παρέχονται τα παρακάτω κριτήρια για την εμφάνιση των αποτελεσμάτων:

1. **Sort by** (σειρά κατάταξης): Ο χρήστης μπορεί να διαλέξει με ποια σειρά θα του εμφανιστούν τα αποτελέσματα, επιλέγοντας ένα από τα παρακάτω 4 κριτήρια:

- **Relevance** (σχετικότητα): τα πιο σχετικά με την αναζήτηση video θα εμφανιστούν πρώτα στη λίστα. Ο καθορισμός της σχέσης μεταξύ της αναζήτησης και του κάθε αποτελέσματος γίνεται εσωτερικά από το API. Αυτή είναι και η default επιλογή.
- **Date** (ημερομηνία): τα πιο πρόσφατα video θα εμφανιστούν πρώτα.
- **Rating** (βαθμολογία): τα video με την υψηλότερη βαθμολογία των χρηστών θα εμφανιστούν πρώτα.
- **Views** (προβολές): εδώ η εφαρμογή θα επιστρέψει πρώτα τα video με τις περισσότερες προβολές χρηστών στο Youtube.

2. Video length (διάρκεια video): Δίνεται η επιλογή στο χρήστη να του εμφανιστούν μόνο τα video με τη διάρκεια που επιθυμεί, ή όλα:

- **Any** (οποιαδήποτε διάρκεια): θα εμφανιστούν όλα τα video, ανεξαρτήτως διάρκεια. Είναι και η default επιλογή για το συγκεκριμένο κριτήριο.
- **Short** (σύντομο): εμφανίζονται μόνο τα video με μικρή διάρκεια.
- **Medium** (μέση διάρκεια)
- **Long** (μακρά διάρκεια)

3. Number of results (πλήθος αποτελεσμάτων): Ο χρήστης μπορεί επίσης να επιλέξει πόσα αποτελέσματα θα του εμφανίζονται σε κάθε σελίδα. Παρέχονται κουμπιά για επόμενη/προηγούμενη σελίδα, όπως θα δούμε και παρακάτω. Οι επιλογές για αυτό το κριτήριο είναι: 5 (default), 10, 15 και 25.

Αναζήτηση

Αφού θα έχει ορίσει τα κριτήρια για την εμφάνιση των αποτελεσμάτων όπως περιγράφηκε παραπάνω ή επιλέξει να αφήσει τις default τιμές τους, ο χρήστης μπορεί να αναζητήσει το video που επιθυμεί. Η διαδικασία είναι πολύ απλή και διαισθητική, αρκεί δηλαδή να εισάγει στο πεδίο “Search YouTube...” το κείμενο με βάση το οποίο επιθυμεί να αναζητήσει το video και να κάνει κλικ στο εικονίδιο του φακού ή να πατήσει το πλήκτρο Enter. Η εφαρμογή θα αποστείλει στο API ένα HTTPS POST request με τα στοιχεία της αναζήτησης και θα περιμένει το αντίστοιχο HTTP respond.

Αποτελέσματα

Αν και εφόσον το HTTP respond επιστρέψει αποτελέσματα, η εφαρμογή θα τα εμφανίσει με τη σειρά που επέστρεψαν, η οποία εξαρτάται από τα κριτήρια αναζήτησης. Κάθε αποτέλεσμα καταλαμβάνει μία σειρά, με την προεπισκόπηση του video αριστερά και λεπτομέρειες για το video (meta-data) στα δεξιά του. Η διάταξη αυτή βέβαια αλλάζει και γίνεται πάνω και κάτω αντί για αριστερά και δεξιά σε μικρότερες αναλύσεις.

Για την προεπισκόπηση δεν χρειάζεται κάποια ανάλυση, πέρα από το είναι clickable και με το κλικ ξεκινά η αναπαραγωγή. Στα μετα-δεδομένα τώρα, υπάρχει πρώτα ο τίτλος του video όπως είναι ανεβασμένο στο Youtube και είναι και αυτός clickable και ξεκινά την αναπαραγωγή του video αν πατηθεί. Από κάτω υπάρχουν τα στατιστικά σε δύο γραμμές. Στην πρώτη έχουμε τη διάρκεια και τις συνολικές προβολές του video από χρήστες του Youtube. Στη δεύτερη γραμμή είναι το πλήθος των likes, των dislikes και των σχολίων (comments). Ακόμη μια γραμμή πιο κάτω βρίσκεται συμπτυγμένη η περιγραφή του video, η οποία μπορεί να είναι και πολύ μεγάλη. Για το λόγο αυτό, προστέθηκε HTML anchor (<a> element) για toggle μεταξύ εμφάνισης και απόκρυψης της πλήρους περιγραφής. Με τον τρόπο αυτό οι πληροφορίες που μπορεί να θεωρούνται περιττές από κάποιους χρήστες δεν εμφανίζονται γεμίζοντας την οθόνη, μπορούν όμως με ένα κλικ να τις εμφανίσουν.

Δημιουργία Λογαριασμού

Η ιστοσελίδα που χειρίζεται τη δημιουργία του λογαριασμού βρίσκεται στο URL <https://sotos.re/ourtube/login.html>. Εκεί ο χρήστης μπορεί στα δεξιά της σελίδας (ή στο κάτω μέρος αν η εφαρμογή έχει τρέξει σε μικρή οθόνη ή μικρό παράθυρο browser) να δημιουργήσει έναν λογαριασμό χρήστη, εισάγοντας τα κλασικά στοιχεία e-mail, όνομα χρήστη και κωδικό πρόσβασης (δύο φορές για επιβεβαίωση) στα πεδία της φόρμας Create Account. Η διαδικασία κι εδώ ακολουθεί την κλασική τακτική που ακολουθείται σχεδόν παντού, οπότε είναι αρκετά εύκολη για το χρήστη. Τα στοιχεία αποστέλλονται στο server και αποθηκεύονται στη βάση δεδομένων. Φυσικά, δεν μπορεί να εισαχθεί χρήστης με κάποιο e-mail ή κάποιο όνομα χρήστη που είναι ήδη καταχωρημένο στη βάση δεδομένων.

Σύνδεση Χρήστη στην Εφαρμογή με Λογαριασμό Χρήστη

Στην ίδια ιστοσελίδα ο χρήστης μπορεί να χρησιμοποιήσει τα στοιχεία εισόδου που δήλωσε στο βήμα δημιουργίας λογαριασμού που περιγράφηκε παραπάνω, για να κάνει είσοδο στην εφαρμογή. Συμπληρώνοντας τα στοιχεία στα πεδία της φόρμας Login (username και password) και πατώντας το κουμπί App Login, η εφαρμογή θα ελέγξει

την εγκυρότητα των στοιχείων στη βάση δεδομένων και, αν είναι σωστά, θα ανακατευθύνει το χρήστη στη σελίδα αναζήτησης.

Σύνδεση Χρήστη με Διαπιστευτήρια Facebook

Πέραν της κλασικής μεθόδου εισόδου με διαπιστευτήρια της εφαρμογής, το OurTube δίνει την ευκαιρία στο χρήστη να εισαχθεί στην εφαρμογή και με το όνομα χρήστη και τον κωδικό του στο Facebook. Αν ο χρήστης επιθυμεί κάτι τέτοιο, τότε πατάει στο κουμπί Log in with FB. Θα εμφανιστεί το παράθυρο εισόδου του Facebook, ώστε ο χρήστης να δώσει εκεί (και όχι στη φόρμα login της εφαρμογής) τα στοιχεία του και να διαπιστευθεί από το ίδιο το Facebook.

Για το λόγο αυτό η εφαρμογή αποθηκεύει σε ξεχωριστούς πίνακες το e-mail από ότι τα στοιχεία εισόδου της εφαρμογής, όπως θα δούμε παρακάτω στην ανάλυση της βάσης δεδομένων.

Τεχνολογίες – Κώδικας – Βάση Δεδομένων

Front-end

Στο front-end χρησιμοποιήθηκε HTML με Javascript και CSS για την εμφάνιση και τη λειτουργικότητα στην πλευρά του browser.

HTML

Δεν χρησιμοποιήθηκε κάτι ιδιαίτερο στην HTML. Προσπαθήσαμε να κρατήσουμε το κομμάτι της παρουσίασης του markup όσο γίνεται πιο καθαρό, αφού το περιεχόμενο της σελίδας δημιουργείται δυναμικά, με βάση τα αποτελέσματα της αναζήτησης του χρήστη. Χρησιμοποιήθηκε HTML 5, αν και δεν υπήρξε κάποιος ιδιαίτερος λόγος.

Javascript

Τα αρχεία javascript τα διαχωρίσαμε για περισσότερη ευκολία τόσο στην ανάπτυξη όσο και στη συντήρηση του κώδικα. Κάθε αρχείο έχει συγκεκριμένο σκοπό και έτσι μπορούμε εύκολα να προσθέσουμε κι άλλη λειτουργικότητα στο σημείο του κώδικα που

πρέπει. Προσπαθήσαμε να κρατήσουμε δηλαδή τον κώδικα καθαρό όσο γίνεται, τόσο για maintainability, όσο και για scalability και code reusability. Επίσης, χρησιμοποιήθηκαν design patterns στα scripts. Προσπαθήσαμε να κάνουμε όσο το δυνατόν λιγότερες global variables, καθώς θεωρούνται, και είναι φυσικά, bad practice.

Κάποια αρχεία χρησιμοποιούν closures, δηλαδή self-invoking functions. Κάποιες από αυτές επιστρέφουν ένα αντικείμενο το οποίο περιέχει encapsulated members και functions. Οι εσωτερικές functions (που σε κλασικές γλώσσες προγραμματισμού θα δηλωνόντουσαν ως private) έχουν εμβέλεια μόνο στο scope του closure, κάτι που τις κάνει στην ουσία private, αφού δεν μπορούν να κληθούν από κανένα άλλο σημείο του κώδικα. Βλέπουμε δηλαδή ότι επειδή η Javascript είναι δυναμική και χωρίς κλάσεις (δεν χρησιμοποιήθηκαν οι κλάσεις της έκδοσης ECMA 6) δεν σημαίνει ότι θα πρέπει να είναι ταυτόχρονα και κακογραμμένη ή εντελώς procedural, με global functions και variables από εδώ κι από κει που μπορεί να τις καλέσει όποιος θέλει από οπουδήποτε. Για παράδειγμα, στο script “renderer.js” υπάρχουν δηλωμένες τρεις functions, αλλά επειδή στο τέλος το closure επιστρέφει ένα αντικείμενο μόνο με δύο από αυτές, η τρίτη δεν μπορεί να κληθεί από πουθενά στον κώδικα, όπως συμβαίνει με τις κλασικές global function. Με τον ίδιο τρόπο μπορούν να αποκρυφτούν και variables. Υπάρχουν και scripts, όπως το “handlers.js”, όπου επιστρέφεται ως αντικείμενο ολόκληρος ο κώδικας που περιέχεται μέσα (όλες οι functions) και πρέπει να κληθούν όπως γίνεται η κλήση των static μεθόδων στις γνωστές αντικειμενοστρεφείς γλώσσες, δηλαδή με τη μορφή Handlers. changePage().

jQuery - AJAX

Για τις κλήσεις στο API του Youtube αλλά και στα server scripts της εφαρμογής χρησιμοποιήθηκαν XML HTTP Requests (XHR) μέσω της βιβλιοθήκης jQuery, με τη μέθοδο POST. Η jQuery χρησιμοποιήθηκε και για άλλες λειτουργίες της εφαρμογής στον κώδικα, όπως η κατανάλωση των HTTP responses είτε από το API του Youtube είτε από τα δικά μας server scripts. Η δυνατότητα της jQuery να δίνονται κλάσεις σε HTML elements μόνο και μόνο για την ευκολότερη διαχείριση τους, σχέση με το αν

αυτές οι κλάσεις θα περιέχουν όντως style, είναι εξαιρετικά βολική. Μπορεί δηλαδή να δηλώνονται κλάσεις σε HTML elements, αλλά αυτές οι κλάσεις να μην υπάρχουν δηλωμένες σε κανένα CSS αρχείο, αλλά να χρησιμοποιούνται μόνο για τη χρήση selectors μέσω της jQuery, ώστε να μπορούμε να χειριζόμαστε συγκεκριμένες ομάδες στοιχείων.

Styles

Για το styling και την εμφάνιση του περιεχομένου στις ιστοσελίδες χρησιμοποιήθηκαν Cascading Style Sheets (CSS) 3. Όλο το styling της σελίδας αναζήτησης βρίσκεται σε ένα αρχείο (styles/searchPage.css) ενώ της σελίδας login σε δικό της ξεχωριστό αρχείο (styles/login.css). Να σημειωθεί ότι σε κανένα στοιχείο στο markup, δηλαδή σε κανένα HTML element, δεν υπάρχει inline το property “style”. Και από αυτή την άποψη δηλαδή, το markup (HTML κομμάτι) είναι πεντακάθαρο.

Bootstrap

Για το responsiveness των ιστοσελίδων χρησιμοποιήθηκε το πασίγνωστο Bootstrap. Με τη χρήση του Bootstrap μπορούμε να ορίζουμε το περιεχόμενο σε γραμμές και στήλες, ανάλογα με το μέγεθος του παραθύρου του client. Σε μικρότερες δηλαδή αναλύσεις, είτε είναι παλιότερες οθόνες είτε mobile devices, προσπαθήσαμε να σχεδιάσουμε έτσι τις ιστοσελίδες ώστε να μην εμφανίζονται οι δύσχρηστες μπάρες οριζόντιας κύλισης (horizontal scrollbars). Το Bootstrap είναι λίγο περίεργο μέχρι να το συνηθίσει κανείς, αλλά άπαξ και γίνει αυτό, μεταμορφώνεται σε πανίσχυρο εργαλείο στα χέρια του designer. Μπορεί να κάνει οποιαδήποτε σελίδα 100% responsive, αρκεί βέβαια να συμπεριληφθούν όλα τα στοιχεία του περιεχομένου στις αντίστοιχες κλάσεις. Επίσης, πέρα από το responsiveness, δίνει και πολλές utility CSS κλάσεις, για χρήση με συχνά χρησιμοποιούμενα HTML elements, όπως buttons. Χρησιμοποιήθηκαν αρκετές από αυτές τις κλάσεις στις ιστοσελίδες της εφαρμογής.

Fancybox

Για την embedded προβολή του video χρησιμοποιήθηκε το εργαλείο Fancybox, που δίνει τη δυνατότητα παραμετροποιήσιμης αναπαραγωγής inline στην ιστοσελίδα αναζήτησης.

Back-end

Στο back-end χρησιμοποιήθηκε η server-side γλώσσα PHP, μαζί με βάση δεδομένων Maria DB (παραλλαγή της MySQL). Πριν περάσουμε στην ανάλυση του back-end, να σημειωθεί ότι το site τρέχει πίσω από HTTPS πρωτόκολλο και όχι απλό HTTP, χρησιμοποιείται δηλαδή SSL.

Server-side Κώδικας

Στην πλευρά του server προτιμήθηκε να έχουμε καθαρή PHP, ώστε να έχουμε και καθαρή HTML. Δηλαδή δεν υπάρχει κάποιο αρχείο PHP που να περιέχει HTML markup. Όλα τα server scripts επιστρέφουν δεδομένα τα οποία με τη χρήση Javascript εμφανίζονται και στην οθόνη του χρήστη. Σε καμία περίπτωση δηλαδή ο χρήστης δεν «χτυπάει» URL που να δείχνει σε αρχείο PHP στον browser του. Όλα αυτά έχουν γίνει στην προσπάθεια διαχωρισμού της εμφάνισης περιεχομένου με τον κώδικα που εκτελείται από πίσω.

Για την διαχείριση των sessions χρησιμοποιήθηκε το Fat-Free Framework (F3) που είναι γραμμένο σε PHP και δίνει πάρα πολλές δυνατότητες απλοποίησης. Εμείς το χρησιμοποιήσαμε αποκλειστικά για τη διαχείριση συνεδριών σε περίπτωση που ο χρήστης αποφασίσει να χρησιμοποιήσει την εφαρμογή συνδεδεμένος.

Τα PHP scripts αναλαμβάνουν να εκτελέσουν τις διαδικασίες εισόδου και εξόδου (login-logout) από την εφαρμογή, τη διαχείριση του session του χρήστη και την αποθήκευση στατιστικών στοιχείων ανά χρήστη και video, που μπορούν να χρησιμοποιηθούν αργότερα από κάποιο νέο script για την εμφάνισή τους σε μελλοντικό update. Το βασικό σημείο όπου δόθηκε έμφαση στον κώδικα PHP ήταν τα prepared statements με τα οποία εκτελούνται όλες οι ενέργειες στη βάση δεδομένων. Χωρίς τα prepared

statements ο κώδικας είναι ευάλωτος σε SQL injection. Οι κλήσεις που γράφτηκαν προς τη βάση δεδομένων χρησιμοποιούν το object-oriented style (αντικειμενοστρεφής τρόπος), με τη δήλωση αντικειμένου τύπου mysqli και χρήση των μεθόδων του και όχι το παλιό κλασικό procedural στυλ.

Βάση Δεδομένων

Χρησιμοποιήθηκε Maria DB με πίνακες για τη συγκέντρωση των στοιχείων των χρηστών και των στατιστικών.

Οι βασικοί πίνακες της εφαρμογής περιλαμβάνουν τους παρακάτω:

- **user_base**: εδώ αποθηκεύεται το μοναδικό ID του χρήστη, καθώς και το e-mail τους. Υπενθυμίζουμε ότι ο χρήστης έχει τη δυνατότητα να συνδεθεί και μέσω Facebook, οπότε σε αυτή την περίπτωση παίρνει ένα μοναδικό ID που αντιστοιχίζεται με το e-mail του. Την επόμενη φορά που θα μπει με τον ίδιο λογαριασμό Facebook, η εφαρμογή θα τραβήξει το e-mail του από τα στοιχεία του Facebook και θα δει αν υπάρχει το e-mail στη βάση δεδομένων. Αν υπάρχει, θα συνεχίσει με το ID χρήστη που του αντιστοιχεί. Σε αντίθετη περίπτωση, θα εισάγει νέα εγγραφή.
- **user_login_info**: εδώ αποθηκεύονται τα στοιχεία εισόδου (διαπιστευτήρια) του χρήστη που επέλεξε όταν δημιούργησε νέο λογαριασμό στην εφαρμογή. Ο πίνακας αυτός δείχνει στον παραπάνω συσχετίζοντας το όνομα χρήστη με το μοναδικό ID του και το e-mail του. Με αυτό τον τρόπο, μόνο οι χρήστες που επέλεξαν να κάνουν λογαριασμό στην εφαρμογή θα έχουν περαστεί σε αυτό τον πίνακα, ενώ οι υπόλοιποι που προτιμούν τη χρήση του Facebook login, θα υπάρχουν μόνο στον πρώτο πίνακα. Έτσι οι δύο πίνακες διατηρούνται «καθαροί» από nullable στήλες και σχετικά ανεξάρτητοι (πέρα από τη συσχέτιση ξένου κλειδιού που έχουν).
- **sessions**: εδώ αποθηκεύονται από το F3 οι μεταβλητές του session.
- **user_video_watch_history**: στον πίνακα αυτό θα κρατούνται στατιστικά για κάθε video που παρακολουθεί ο χρήστης. Κάθε video έχει ένα μοναδικό αναγνωριστικό

στο Youtube, επομένως εδώ μπορούμε να κρατάμε πληροφορίες σχετικά με το πλήθος προβολών του χρήστη από την εφαρμογή.

Μελλοντική Στόχοι – Επεκτασιμότητα

Όπως προαναφέρθηκε, η κατανάλωση των στατιστικών χρήσης δεν έχει πραγματοποιηθεί. Στα πλαίσια παρουσίασης αυτών των στατιστικών στο χρήστη, μπορεί να γραφτεί ένα νέο PHP script που θα επιστρέφει αυτά τα στατιστικά στο χρήστη και με τη χρήση Javascript να παρουσιάζονται στον browser του χρήστη, πιθανότατα με τη χρήση κάποιου popup-like CSS window.

Κάτι που προσπαθήσαμε και δεν πέτυχε ήταν η χρήση και δεύτερου εναλλακτικού τρόπου σύνδεσης, αυτή τη φορά μέσω Google account. Δυστυχώς στον host που μεταφερθήκαμε τώρα πήγαμε πριν λίγες ημέρες και στον παλιό δεν έγινε εφικτό να ρυθμιστεί η εφαρμογή στο αντίστοιχο Google account μας.