

Ατομική Διπλωματική Εργασία

**ΔΗΜΙΟΥΡΓΙΑ ΠΡΟΦΙΛ ΜΗΧΑΝΙΚΩΝ ΛΟΓΙΣΜΙΚΟΥ ΑΠΟ
ΚΟΙΝΩΝΙΚΑ ΔΙΚΤΥΑ**

Παναγιώτης Φούτρος

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2016

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Δημιουργία προφίλ μηχανικών λογισμικού από κοινωνικά δίκτυα
Παναγιώτης Φούτρος

Επιβλέπουσα Καθηγήτρια
Γεωργία Καπιτσάκη

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2016

Ευχαριστίες

Πρώτα απ' όλα θέλω να ευχαριστήσω την επιβλέπουσα καθηγήτρια μου Δρ. Γεωργία Καπιτσάκη για την υποστήριξη, τη βοήθεια, την καθοδήγηση αλλά και την υπομονή της καθ' όλη τη διάρκεια εκπόνησης της ατομικής διπλωματικής εργασίας. Μου έδωσε όλες τις απαραίτητες κατευθυντήριες γραμμές και έδειξε κατανόηση, ως αποτέλεσμα να έχουμε μια πολύ καλή συνεργασία.

Επίσης θα ήθελα να ευχαριστήσω την οικογένεια μου για την υποστήριξη τους, όχι μόνο στη παρούσα εργασία αλλά καθ' όλη τη διάρκεια των σπουδών μου.

Τέλος θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες στην φίλη μου Νάταλυ Ανδρέου που με στήριξε σε όλες τις δυσκολίες που βρήκα μπροστά μου και μου ανέβαζε το ηθικό και την αυτοπεποίθηση, όπως και όλους τους φίλους και συμφοιτητές μου που ήταν δίπλα μου.

Περίληψη

Στις μέρες μας παρέχεται ένας μεγάλος αριθμός εφαρμογών ελέγχου εκδόσεων, κοινωνικής δικτύωσης και ερωταπαντήσεων στις οποίες μπορούν να συμμετέχουν διάφοροι μηχανικοί λογισμικού και να συνεισφέρουν στην κοινότητα ανοικτού λογισμικού. Ενδεικτικά αναφέρονται τα: GitHub, Bitbucket, Stack Overflow και Twitter. Σκοπός της παρούσας διπλωματικής εργασίας είναι η μελέτη των παραπάνω εφαρμογών όσον αφορά τη δυνατότητα συμμετοχής των χρηστών, την ανάλυση των πληροφοριών που αποθηκεύονται για κάθε χρήστη και την εξαγωγή ενός κατάλληλου προφίλ χρήστη που συνδυάζει πληροφορίες από διάφορες πηγές.

Μέρος της εργασίας αποτελεί ο ορισμός του προφίλ του μηχανικού λογισμικού. Το προφίλ που θα εξαχθεί από αυτή την πληροφορία θα μπορεί να αναπαρασταθεί σε μια διαφορετική μορφή (π.χ. διαγραμματική), καθώς και να μπορεί να χρησιμοποιηθεί στη συνέχεια για την εξαγωγή συμπερασμάτων για τις δραστηριότητες, την εμπειρία και τις ικανότητες του μηχανικού λογισμικού όπως και κάποιες βαθμολογίες, όσο αφορά γλώσσες προγραμματισμού και τεχνολογίες.

Ο χρήστης θα μπορεί να δημιουργήσει το προφίλ του, στην ιστοσελίδα που δημιουργήθηκε στα πλαίσια της εργασίας με το όνομα Expertise Analyzer, δίνοντας σαν είσοδο το όνομα χρήστη για τις προαναφερθείσες εφαρμογές. Επίσης όταν ο χρήστης έχει είδη δημιουργήσει μια φορά το προφίλ του, σε κάθε επόμενη επίσκεψη στη σελίδα, θα μπορεί να ανακτά το προφίλ που δημιούργησε την τελευταία φορά ή να το δημιουργεί ξανά ώστε να ενημερωθεί, εφόσον υπήρξαν κάποιες αλλαγές σε αυτό.

Περιεχόμενα

ΚΕΦΑΛΑΙΟ 1 ΕΙΣΑΓΩΓΗ	1
1.1 Κίνητρο	1
1.2 Στόχοι	2
1.3 Δομή	3
ΚΕΦΑΛΑΙΟ 2 ΠΗΓΕΣ ΠΛΗΡΟΦΟΡΙΩΝ	5
2.1 Συστήματα Ελέγχου Εκδόσεων	5
2.1.1 Κεντροποιημένα Συστήματα Εκδόσεων	6
2.1.2 Κατανεμημένα Συστήματα Εκδόσεων	6
2.1.3 GitHub	7
2.1.4 Bitbucket	8
2.2 Ιστοσελίδες Ερωταπαντήσεων	8
2.2.1 Stack Overflow	9
2.3 Μέσα Κοινωνικής Δικτύωσης	9
2.3.1 Twitter	10
2.4. Λόγοι Επιλογής των Πηγών	11
2.5 Ετικέτες(Tags) Stack Overflow	12
ΚΕΦΑΛΑΙΟ 3 ΕΞΟΡΥΞΗ ΔΕΔΟΜΕΝΩΝ	13
3.1 Προγραμματιστικά Πλαίσια (API)	13
3.2 GitHub API	15
3.3 Bitbucket API	17
3.4 Stack Overflow API	18
3.5 Twitter API	20
ΚΕΦΑΛΑΙΟ 4 ΣΤΑΤΙΣΤΙΚΑ – ΒΑΘΜΟΛΟΓΙΕΣ ΧΡΗΣΤΗ	23
4.1 Στατιστικά από GitHub	23
4.2 Στατιστικά από Bitbucket	24
4.3 Στατιστικά από Stack Overflow	24
4.4 Στατιστικά από Twitter	25
4.5 Βαθμολογίες Χρήστη	26
4.6 Σύγκριση με Προηγούμενες Προσεγγίσεις	31

ΚΕΦΑΛΑΙΟ 5 ΥΛΟΠΟΙΗΣΗ ΚΑΙ ΛΕΙΤΟΥΡΓΙΑ ΤΗΣ ΙΣΤΟΣΕΛΙΔΑΣ	35
5.1 Περιορισμοί	36
5.2 Περιπτώσεις Χρήσης	36
5.2.1 Χρήστες	36
5.2.2 Περιγραφή Χρηστών	38
5.2.3 Περιγραφή Περιπτώσεων Χρήσης	39
5.3 Αρχιτεκτονική	40
5.4 Βάση Δεδομένων	41
5.5 Διαπροσωπείες Συστήματος(Interfaces)	44
5.5.1 Διαπροσωπεία Χρήστη	44
5.5.2 Διαπροσωπεία Διαχειριστή	46
5.6 Λειτουργία Ιστοσελίδας	48
5.6.1 Λειτουργία Ιστοσελίδας Χρήστη	48
5.6.2 Λειτουργία Ιστοσελίδας Διαχειριστή	49
5.7 Εργαλεία που χρησιμοποιήθηκαν	50
ΚΕΦΑΛΑΙΟ 6 ΑΠΟΤΕΛΕΣΜΑΤΑ ΧΡΗΣΗΣ	51
6.1 Ερωτηματολόγιο	51
6.2 Αποτελέσματα	52
ΚΕΦΑΛΑΙΟ 7 ΣΥΜΠΕΡΑΣΜΑΤΑ	57
7.1 Συμπεράσματα	57
7.2 Μελλοντική Εργασία	59
ΒΙΒΛΙΟΓΡΑΦΙΑ	I
ΠΑΡΑΡΤΗΜΑ Α	III

Κεφάλαιο 1

Εισαγωγή

1.1 Κίνητρο	1
1.2 Στόχοι	2
1.3 Δομή	3

1.1 Κίνητρο

Ζούμε σε μια εποχή όπου τα μέσα κοινωνικής δικτύωσης έχουν καθιερωθεί στο διαδίκτυο και έχουν κατακτήσει τις ζωές μας ανά το παγκόσμιο. Πολλές ώρες ξοδεύονται καθημερινά από τους ανθρώπους στα συστήματα αυτά, γεγονός που τα κατατάσσει ως ένα κυρίαρχο ρεύμα της κοινωνίας. Ο κάθε χρήστης δημοσιεύει τις δικές του πληροφορίες που αναμφίβολα κάποιες από αυτές είναι χρήσιμες και χαίρουν μελέτης και ανάλυσης.

Επίσης υπάρχουν κοινωνικές εφαρμογές που αφορούν την επιστήμη της πληροφορικής και χρησιμοποιούνται από πολλούς μηχανικούς λογισμικού.

Ανάμεσα σε αυτές κατατάσσονται τα συστήματα ελέγχου εκδόσεων (version control systems) όπως είναι το GitHub [1] και το Bitbucket [2]. Σε αυτά τα συστήματα οι χρήστες μπορούν να δημιουργήσουν το δικό τους αποθετήριο ώστε να αποθηκεύουν και να αναζητούν διάφορα συστατικά κώδικα τα οποία αποτελούν μέρη ενός έργου ανάπτυξης. Επιπρόσθετα μπορούν να δημιουργούν και να διαχειρίζονται διάφορες εκδόσεις του έργου ανάπτυξης, ενώ υπάρχει το στοιχείο της κοινωνικής δικτύωσης. Τα συστήματα ελέγχου εκδόσεων βοηθούν στη συνεργασία των μηχανικών λογισμικού που εργάζονται σε ομάδες αφού εφαρμόζεται μια κοινή προσέγγιση όσο αφορά την τεκμηρίωση, αναζήτηση και ανάκτηση των συστατικών.

Μια άλλη κοινωνική εφαρμογή είναι οι ιστοσελίδες ερωταπαντήσεων (q&a sites) όπως είναι το Stack Overflow το οποίο αποτελεί την πιο δημοφιλή σελίδα του Stack Exchange. Αυτού

του είδους οι ιστοσελίδες προσφέρουν στους χρήστες, αφού εγγραφούν, την ικανότητα να ρωτούν και να απαντούν ερωτήσεις, όπως επίσης να σχολιάζουν και να ψηφίζουν θετικά ή αρνητικά τις ερωτήσεις και τις απαντήσεις άλλων χρηστών και γενικά να τις επεξεργάζονται με ένα τρόπο παρόμοιο με τα wikis [3]. Εκείνο που έχει ιδιαίτερη σημασία για τις ιστοσελίδες ερωταπαντήσεων είναι ότι περιέχουν άφθονη γνώση την οποία μπορούν να μοιραστούν οι χρήστες μεταξύ τους και να βοηθήσουν ο ένας τον άλλο.

Χρειάζεται επίσης να σημειωθεί ότι ανάμεσα στα πιο δημοφιλή κοινωνικά δίκτυα κατατάσσεται και το Twitter. Είναι ένα κοινωνικό δίκτυο μέσα από το οποίο μπορεί ο κάθε χρήστης να εκφράσει τις απόψεις του, τα ενδιαφέροντά του, να μοιραστεί γεγονότα και οτιδήποτε άλλο τον απασχολεί, κάνοντας αναρτήσεις και σχολιάζοντας αναρτήσεις άλλων χρηστών. Με τα πιο πάνω δεδομένα δεν εκπλήσσει το γεγονός ότι ο χρήστης εκφράζει τη δική του συμπεριφορά, όπως και κάποιες δραστηριότητες που αφορούν τον ίδιο [4]. Γίνεται, επομένως, αντιληπτό ότι το Twitter παρέχει χρήσιμα δεδομένα για εξαγωγή συμπερασμάτων σχετικά με τους χρήστες που το χρησιμοποιούν.

Οι παραπάνω λόγοι αποτέλεσαν το κίνητρο μελέτης των μέσων κοινωνικής δικτύωσης και των υπόλοιπων αναφερθέντων εφαρμογών ώστε μέσω ενός εργαλείου να δημιουργείται το προφίλ, συγκεκριμένα για μηχανικούς λογισμικού. Ταυτόχρονα υπάρχει η ανάγκη δικτύωσης των μηχανικών λογισμικού και προβολής της δουλειάς τους αλλά και της γνώσης που κατέχουν σχετικά με διάφορα προγραμματιστικά πεδία. Έτσι μέσα από τη προβολή των προφίλ τους, πιθανοί εργοδότες μπορούν να τους προτιμήσουν σε θέσεις εργασίας. Το γεγονός αυτό είναι ένας επιπλέον λόγος για τη συγκεκριμένη μελέτη. Επομένως αφού πρώτα αναλυθούν οι λογαριασμοί που διατηρεί ο μηχανικός λογισμικού, όσο αφορά το GitHub, Bitbucket, Stack Overflow και το Twitter έπειτα θα γίνει μια ανάλυση και παρουσίαση των δεδομένων.

1.2 Στόχοι

Κύριος στόχος της παρούσας διπλωματικής είναι η μελέτη των λογαριασμών GitHub, Bitbucket, Stack Overflow και Twitter, τους οποίους διατηρεί ο χρήστης, ώστε να δοθεί μια συνολική εικόνα χρήσης των κοινωνικών αυτών δικτύων από μηχανικούς λογισμικού. Πιο συγκεκριμένα η ανάλυση κάποιων παραμέτρων για τον κάθε λογαριασμό και η εξαγωγή συμπερασμάτων. Αυτό θα επιτυγχάνεται με τη δημιουργία ιστοσελίδας με το όνομα Expertise Analyzer την οποία θα χρησιμοποιεί ο χρήστης ώστε να δημιουργήσει το προφίλ του.

Ως τώρα δεν υπάρχει συγκεντρωμένη η πληροφορία που αφορά τους διάφορους λογαριασμούς του χρήστη. Αντιθέτως, η πληροφορία βρίσκεται διεσπαρμένη στα διάφορα κοινωνικά δίκτυα. Οπότε συλλέγοντας τα δεδομένα του χρήστη από τις διάφορες πηγές, θα γίνεται και μια συσχέτιση των προφίλ που διατηρεί ο χρήστης, μέσω των ονομάτων χρήστη. Δηλαδή για παράδειγμα θα είναι γνωστό ότι ο μηχανικός λογισμικού «Α» έχει λογαριασμό στο GitHub, με το όνομα χρήστη «ΓΑ» και λογαριασμό στο Twitter με το όνομα χρήστη «ΤΑ».

Επιπλέον ο χρήστης θα έχει το δικαίωμα να δει το προφίλ του στο διαδίκτυο, δίνοντας στο σύστημα ως είσοδο τα ονόματα χρήστη που διατηρεί για τους τέσσερις προαναφερθέντες λογαριασμούς (ή για κάποιους από αυτούς στους οποίους διατηρεί λογαριασμό). Έτσι θα δημιουργείται το προφίλ του και θα του εμφανίζονται διαγραμματικά και λεκτικά κάποια αποτελέσματα που αφορούν τη δραστηριότητα του αλλά και τη γνώση του σε πεδία της πληροφορικής, όπως και κάποιες βαθμολογίες οι οποίες υπολογίζονται από το σύστημα.

Παρ' όλα αυτά το προφίλ θα χρειάζεται κάποιο χρονικό διάστημα ώστε να δημιουργηθεί, για αυτό λοιπόν εφόσον το δημιουργήσει μια φορά ο χρήστης, θα του δίνεται η δυνατότητα να το ανακτά και να βλέπει την ίδια πληροφορία, χωρίς να χρειάζεται να το δημιουργήσει ξανά. Ωστόσο πρέπει να αναφερθεί ότι κάθε φορά που θα δημιουργεί το προφίλ του, αυτό θα ενημερώνεται με τις αλλαγές που έχουν γίνει στα προφίλ του χρήστη στο διάστημα που μεσολάβησε από την τελευταία δημιουργία του συνολικού προφίλ.

1.3 Δομή

Στο κεφάλαιο 1 περιγράφεται το κίνητρο και γενική περιγραφή κάποιων παραγόντων που ώθησαν στην εκπόνηση της παρούσας διπλωματικής εργασίας, όπως και τους στόχους και το τι προορίζεται να υλοποιηθεί κατά τη διάρκεια της.

Όσο αφορά το κεφάλαιο 2 παρέχονται λεπτομέρειες και επεξήγηση σχετικά με τις πηγές πληροφοριών που χρησιμοποιήθηκαν για την υλοποίηση. Επίσης αναφέρονται και οι λόγοι οι οποίοι οδήγησαν στην επιλογή των συγκεκριμένων πηγών πληροφόρησης.

Συνεχίζοντας στο κεφάλαιο 3 περιγράφονται αναλυτικά τα προγραμματιστικά πλαίσια. Επιπρόσθετα αναλύεται και ο τρόπος με τον οποίο χρησιμοποιήθηκαν τα συγκεκριμένα προγραμματιστικά πλαίσια και άλλες προαπαιτούμενες ενέργειες που χρειάστηκε να γίνουν.

Προχωρώντας στο κεφάλαιο 4 γίνεται επεξήγηση των βαθμολογιών και των στατιστικών στοιχείων που υπολογίζονται για κάθε χρήστη, μαζί με τα απαραίτητα δεδομένα που χρειάζεται το καθένα από αυτά και ο τύπος που υπολογίζονται.

Στο κεφάλαιο 5 παρουσιάζεται η υλοποίηση του συστήματος. Η υλοποίηση εκτός από τους περιορισμούς που λήφθηκαν υπόψη αφορά και την αρχιτεκτονική του συστήματος, την σχεδίαση της βάσης δεδομένων και τον τρόπο με τον οποίο λειτουργεί το σύστημα.

Ακολουθεί το κεφάλαιο 6 στο οποίο γίνεται αναφορά στη μέθοδο με την οποία αξιολογήθηκε το σύστημα, η οποία είναι με τη χρήση ερωτηματολογίων. Αναφέρονται οι σχετικές ερωτήσεις των ερωτηματολογίων και τα αποτελέσματα από τους χρήστες που χρησιμοποίησαν το σύστημα.

Τέλος στο κεφάλαιο 7 περιγράφονται τα συμπεράσματα τα οποία εξήχθησαν από την μελέτη της παρούσας διπλωματικής και από τα ερωτηματολόγια των χρηστών. Επίσης αναφέρεται και η μελλοντική εργασία που θα ήταν δυνατό να υλοποιηθεί σε μεταγενέστερα στάδια ώστε να επεκταθεί το σύστημα.

Κεφάλαιο 2

Πηγές Πληροφοριών

2.1 Συστήματα Ελέγχου Εκδόσεων	5
2.2 Ιστοσελίδες Ερωταπαντήσεων	8
2.3 Μέσα Κοινωνικής Δικτύωσης	9
2.4. Λόγοι Επιλογής των Πηγών	11
2.5. Ετικέτες(Tags) Stack Overflow	12

Πιο κάτω θα ακολουθήσουν σύντομες περιγραφές των εφαρμογών που χρησιμοποιήθηκαν στη διπλωματική αυτή.

2.1 Συστήματα Ελέγχου Εκδόσεων

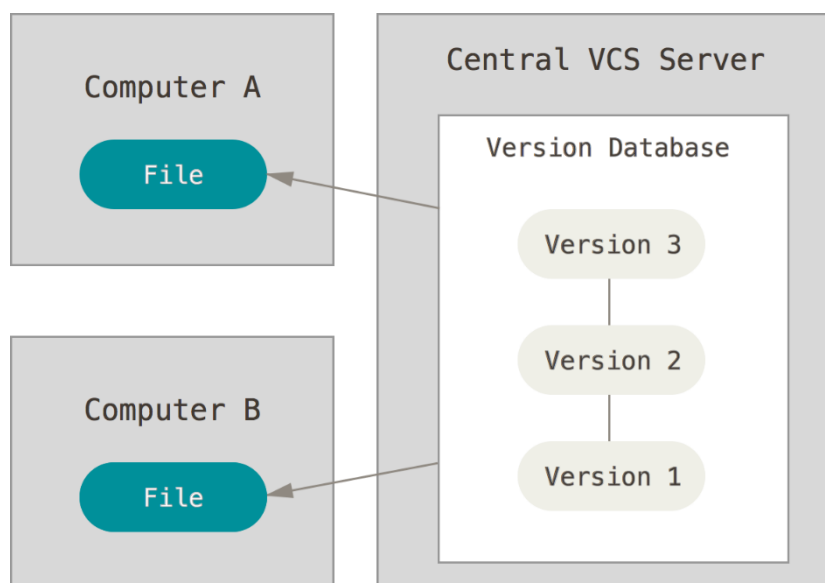
Τα συστήματα ελέγχου εκδόσεων παρατηρούν τις αλλαγές στα αρχεία κατά τη διάρκεια της ανάπτυξης ενός συστήματος και έχουν την δυνατότητα να ανακαλούν παλαιότερες εκδόσεις. Επιτρέπει στο χρήστη δηλαδή να ανακτά αρχεία που βρίσκονταν σε διαφορετική μορφή, παλαιότερες χρονικές στιγμές ή και ακόμα ολόκληρο το σύστημα. Εκτός από αυτό ο χρήστης έχει την ικανότητα να συγκρίνει αλλαγές σε αρχεία κατά την πάροδο του χρόνου και να γνωρίζει από ποιο δημιουργήθηκε το αρχείο και ποιος έχει κάνει αλλαγές σε αυτό.

Αξίζει να σημειωθεί ότι κάποια σύγχρονα διαδικτυακά συστήματα ελέγχου επιτρέπουν στον χρήστη να διατηρεί το δικό του προφίλ και να μπορεί να ακολουθεί προφίλ άλλων χρηστών, όπως επίσης και να έχει τους δικούς του ακόλουθους. Ακόμα μπορεί να δει κάποια στατιστικά σχετικά με τη συνεισφορά του στο σύστημα και τα έργα που έχει δουλέψει.

Τέλος υπάρχουν δυο βασικές κατηγορίες συστημάτων ελέγχου εκδόσεων οι οποίες είναι τα κεντροποιημένα συστήματα εκδόσεων (centralized version control systems) και τα κατανεμημένα συστήματα εκδόσεων (distributed version control systems). Στην παρούσα διπλωματική πρόκειται να ασχοληθούμε με δυο κατανεμημένα συστήματα εκδόσεων τα οποία είναι το GitHub και το Bitbucket.

2.1.1 Κεντροποιημένα Συστήματα Εκδόσεων

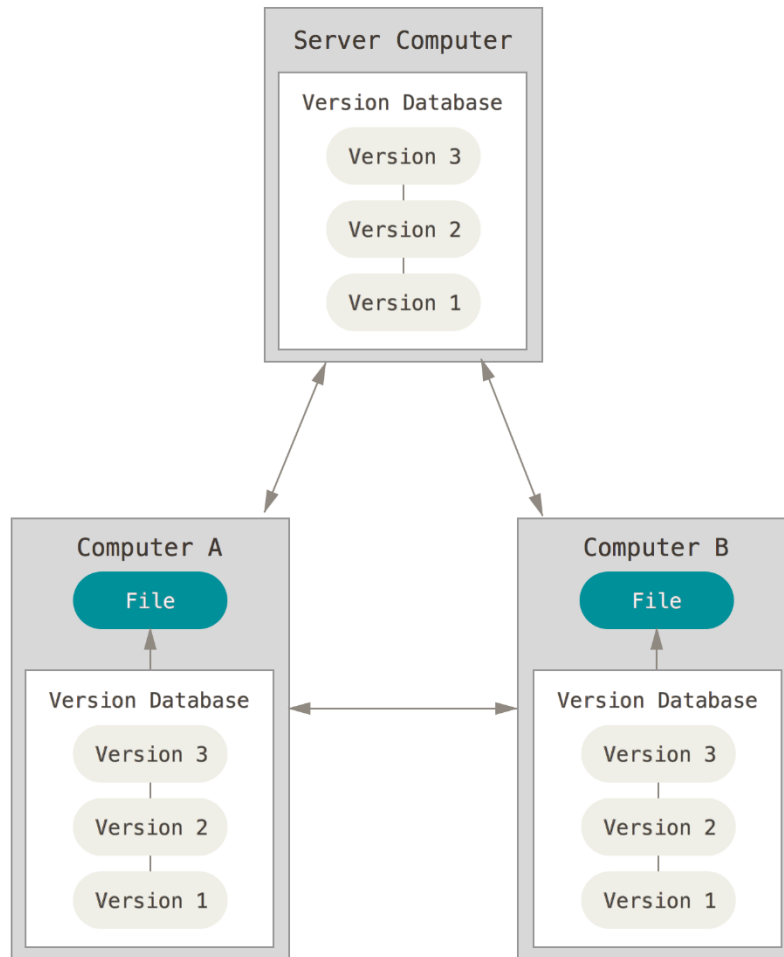
Τα κεντροποιημένα συστήματα έχουν ένα μόνο εξυπηρετητή ο οποίος περιέχει όλες τις εκδόσεις των αρχείων, όπως και ένα αριθμό χρηστών οι οποίοι μπορούν να ανακτήσουν αυτά τα αρχεία. Έτσι με αυτό τον τρόπο ο κάθε μηχανικός λογισμικού μπορεί να γνωρίζει σε ποια αρχεία έχουν δουλέψει οι υπόλοιποι μηχανικοί που βρίσκονται στην ομάδα ανάπτυξης και γενικά γίνεται ευκολότερη η διαχείριση ανάπτυξης του έργου [5]. Στο Σχήμα 2.1 φαίνεται η αρχιτεκτονική του συστήματος αυτού.



Σχήμα 2.1 Αρχιτεκτονική Κεντροποιημένου Συστήματος

2.1.2 Κατανεμημένα Συστήματα Εκδόσεων

Από την άλλη πλευρά τα κατανεμημένα συστήματα εκδόσεων δίνουν την ικανότητα στους χρήστες όχι μόνο να ανακτήσουν τα αρχεία από τον εξυπηρετητή αλλά να δημιουργήσουν ένα κλώνο ολόκληρου του αποθετηρίου στην τοπική τους μηχανή. Έτσι σε περίπτωση που ο εξυπηρετητής αντιμετωπίζει κάποιο πρόβλημα και χαθούν τα αρχεία που βρίσκονται σε αυτόν, τότε μπορούν να τα επαναφέρουν αντιγράφοντας όλο το αποθετήριο πίσω στον εξυπηρετητή, που διατηρεί ένας από τους χρήστες στην τοπική του μηχανή. Επιπλέον τα κατανεμημένα συστήματα επιτρέπουν στους χρήστες να έχουν πολλά απομακρυσμένα αποθετήρια ώστε να μπορούν να συνεργάζονται με διάφορες ομάδες ανάπτυξης στα ίδια έργα ανάπτυξης [5]. Στο Σχήμα 1.2 φαίνεται η αρχιτεκτονική του συστήματος αυτού.



Σχήμα 2.2 Αρχιτεκτονική Κατανεμημένου Συστήματος

2.1.3 GitHub

Το GitHub είναι μια κοινωνική πλατφόρμα κώδικα, η οποία επιτρέπει στους μηχανικούς λογισμικού να δουλεύουν αποδοτικά πάνω σε διάφορα έργα, να συνδέονται και να συνεργάζονται με άλλους μηχανικούς λογισμικού, καθώς και να μπορεί ο χρήστης να βλέπει τη δουλειά άλλων χρηστών. Σύμφωνα με στατιστικά στοιχεία του 2016 το GitHub φιλοξενεί περισσότερους από 14 εκατομμύρια χρήστες και περισσότερα από 35 εκατομμύρια αποθετήρια κώδικα, κατατάσσοντας το ως το μεγαλύτερο σύστημα φιλοξενίας πηγαίου κώδικα στον κόσμο [6].

Όσο αφορά τη λειτουργία του ανήκει στην κατηγορία των κατανεμημένων συστημάτων εκδόσεων και επιτρέπει τη διαχείριση εκδόσεων αρχείων κώδικα και τη δημιουργία αποθετηρίων. Δηλαδή ο κάθε χρήστης μπορεί να δημιουργήσει το δικό του μέρος αποθήκευσης κώδικα ενός συγκεκριμένου έργου και να ανεβάζει εκεί τα συστατικά λογισμικού και να ενημερώνει το αποθετήριο, όπως και να δημιουργεί διαφορετικές εκδόσεις του έργου. Επίσης προσφέρει την ικανότητα στο χρήστη να μπορεί να εντοπίζει αλλαγές

ανάμεσα σε διαφορετικές εκδόσεις ενός αρχείου αλλά παρέχει και διάφορες ευκολίες όσο αφορά τη συνεργασία χρηστών, όπως είναι η παρακολούθηση σφαλμάτων και η διαχείριση διεργασιών μέσα σε ομάδες ανάπτυξης λογισμικού.

Τέλος ο κάθε χρήστης διατηρεί το δικό του προφίλ το οποίο περιέχει όλα τα αποθετήρια που έχει δημιουργήσει, καθώς και όλες τις συνεισφορές που έχει κάνει. Αναφορικά με τη σύνδεση ανάμεσα στους χρήστες, υπάρχει η έννοια των ακολούθων (followers) όπου οι χρήστες μπορούν να ακολουθήσουν και ταυτόχρονα να ακολουθούνται από άλλους χρήστες.

2.1.4 Bitbucket

Όπως είναι το GitHub που αναφέρθηκε προηγουμένως, ένα παρόμοιο σύστημα είναι και το Bitbucket, το οποίο επίσης ανήκει στην κατηγορία των κατανεμημένων συστημάτων εκδόσεων.

Έχει σχεδόν τις ίδιες λειτουργίες με το GitHub όσο αφορά τη φιλοξενία κώδικα και τη σύνδεση μεταξύ χρηστών. Επίσης ο κάθε χρήστης διατηρεί το δικό του προφίλ όπου υπάρχουν τα συστατικά λογισμικού και τα αποθετήρια κώδικα που δημιουργήθηκαν από τον χρήστη. Τα αποθετήρια μπορούν να είναι είτε ανοικτά προς την κοινότητα, είτε προσωπικά ώστε να είναι ορατά μόνο από τον χρήστη.

2.2 Ιστοσελίδες Ερωταπαντήσεων

Μια ιστοσελίδα ερωταπαντήσεων είναι ένας ιστότοπος όπου ο χρήστης μπορεί να εγγραφεί και ακολούθως να αναρτήσει τις δικές του ερωτήσεις σχετικά με κάποιο θέμα. Ακολούθως οι υπόλοιποι χρήστες της ιστοσελίδας, εφόσον πιστεύουν ότι έχουν τη γνώση και μπορούν να απαντήσουν κάποια συγκεκριμένη ερώτηση, τότε γράφουν την απάντηση τους και την αναρτούν κάτω από την σχετική ερώτηση. Εκτός από αυτό οι χρήστες μπορούν να ψηφίζουν αρνητικά ή θετικά τόσο τις ερωτήσεις που αναρτώνται στην ιστοσελίδα, όσο και τις απαντήσεις και να σχολιάζουν τις αναρτήσεις. Όλοι οι χρήστες διατηρούν το δικό τους προφίλ όπως και κάποιο ιστορικό σύμφωνα με τις ερωτήσεις και απαντήσεις που έχουν αναρτήσει. Παράλληλα για κάθε χρήστη υπάρχει είτε η φήμη ανάλογα με την συνεισφορά του στην ιστοσελίδα, είτε οι ακόλουθοι, οι οποίοι είναι άλλοι χρήστες που τον ακολουθούν. Στην

διπλωματική αυτή πρόκειται να ασχοληθούμε με μια ιστοσελίδα ερωταπαντήσεων η οποία είναι το Stack Overflow.

2.2.1 Stack Overflow

Το Stack Overflow είναι η μεγαλύτερη διαδικτυακή κοινότητα στην οποία οι μηχανικοί λογισμικού μπορούν να εξελίσσουν, να μοιραστούν τις γνώσεις τους και να βελτιώσουν την σταδιοδρομία τους [7]. Αξίζει να σημειωθεί ότι ανήκει σε μια από τις μεγαλύτερες ιστοσελίδες ερωταπαντήσεων, το Stack Exchange και αφορά προγραμματιστικά θέματα. Φιλοξενεί περισσότερους από 4,000,000 χρήστες και περισσότερες από 10,000,000 ερωτήσεις σύμφωνα με στατιστικά στοιχεία του 2014 [3].

Αναφορικά με τον τρόπο λειτουργίας του Stack Overflow, λειτουργεί με τον τρόπο που περιγράψαμε πιο πάνω όσο αφορά τις σελίδες ερωταπαντήσεων, δηλαδή επιτρέπει στους χρήστες να αναρτήσουν ερωτήσεις, απαντήσεις, κάποια σχόλια και να βαθμολογήσουν θετικά ή αρνητικά μια ανάρτηση. Επίσης χρησιμοποιεί ετικέτες (tags), σχετικές με το θέμα τις ιστοσελίδας, ώστε να κατηγοριοποιούνται οι ερωτήσεις που αναρτώνται από τους χρήστες. Παράλληλα προσφέρει κάποιες μεθόδους βαθμολόγησης των χρηστών οι οποίες είναι τα εμβλήματα(badges) και η φήμη(reputation). Συγκεκριμένα δίνει τη δυνατότητα στους χρήστες να αποκτήσουν διάφορα εμβλήματα (badges) που αφορούν όλες τις ετικέτες της ιστοσελίδας. Τα εμβλήματα χωρίζονται σε τρεις κατηγορίες οι οποίες είναι χρυσά, αργυρά και χάλκινα με το πρώτο να είναι το υψηλότερο. Για να κερδίσει ένας χρήστης κάποιο έμβλημα υπάρχουν διάφοροι τρόποι που σχετίζονται με τη δραστηριότητά του στην ιστοσελίδα. Σχετικά με την φήμη του χρήστη, η ιστοσελίδα υπολογίζει ένα ακέραιο αριθμό λαμβάνοντας υπόψη ξανά τη δραστηριότητα που έχει στην ιστοσελίδα. Αυτός ο αριθμός όσο μεγαλύτερος είναι, τόσο μεγαλύτερη φήμη έχει ο χρήστης και τόσο πιο ενεργός είναι στο Stack Overflow.

2.3 Μέσα Κοινωνικής Δικτύωσης

Τα μέσα κοινωνικής δικτύωσης είναι διαδικτυακά εργαλεία που επιτρέπουν στους ανθρώπους να δημιουργούν, να μοιράζονται ή να ανταλλάζουν τόσο πληροφορίες, σχετικά με την καριέρα, τις ιδέες και τα ενδιαφέροντά τους, όσο και εικόνες, βίντεο μέσα σε ιδεατές κοινότητες και δίκτυα. Οι χρήστες αφού δημιουργούν τα προφίλ τους, παρέχοντας κάποιες πληροφορίες της επιθυμίας τους σχετικά με τον εαυτό τους, μπορούν να χρησιμοποιήσουν το μέσο κοινωνικής

δικτύωσης. Το δίκτυο τους παρέχει τη δυνατότητα σύνδεσης και επικοινωνίας με προφίλ άλλων χρηστών ή ομάδες ατόμων.

Επιπρόσθετα τα δίκτυα αυτά εμφανίζονται σε πολλές μορφές συμπεριλαμβανομένων των blogs, κοινωνικά δίκτυα επιχειρήσεων, κοινωνικά δίκτυα, κοινής χρήσης εικόνων και βίντεο, φόρουμ, εικονικούς κόσμους και άλλα. Το όφελος των χρηστών από τα κοινωνικά δίκτυα και ο λόγος που μοιράζονται πληροφορίες είναι να γίνονται δημοφιλείς, να ανοίγουν νέες ευκαιρίες εργοδότησης και να έχουν χρηματικό κέρδος [8].

2.3.1 Twitter

Το Twitter είναι μια δημοφιλής υπηρεσία microblogging. Με τον όρο microblogging εννοούμε ένα σύστημα στο οποίο οι χρήστες δημοσιεύουν πληροφορίες και αλληλεπιδρούν μέσω μικρού μεγέθους κειμένων. Επίσης χαρακτηρίζεται ως ένας σύντομος τρόπος επικοινωνίας, αφού χρησιμοποιώντας μικρά κείμενα μειώνεται ο χρόνος που χρειάζεται ο χρήστης για να κάνει μια δημοσίευση [9].

Στο Twitter οι χρήστες δημιουργούν το προφίλ τους και μπορούν να περιγράψουν και να μοιραστούν πληροφορίες σχετικές με τη ζωή τους. Αυτό επιτυγχάνεται χρησιμοποιώντας σύντομες αναρτήσεις μέχρι 140 χαρακτήρων, τα γνωστά tweets [10], όπως επίσης και εικόνες. Οι χρήστες μπορούν να ακολουθούν ή να ακολουθούνται από άλλους χρήστες. Η σημασία του να ακολουθείς ένα χρήστη στο Twitter είναι ότι θα λαμβάνεις όλα τα tweets που αναρτά ο συγκεκριμένος χρήστης.

Επιπλέον το Twitter διατηρεί ετικέτες οι οποίες δημιουργούνται από τους χρήστες και αποκαλούνται hashtags. Πιο συγκεκριμένα ο χρήστης εμπεριέχει στο κείμενο που θα αναρτήσει, το σύμβολο '#' ακολουθούμενο από λέξεις χωρίς κενά μεταξύ τους, το οποία αποτελούν το hashtag. Κάθε hashtag που χρησιμοποιείται έχει τη δική του σελίδα που περιέχει όλες τις αναρτήσεις χρηστών που εμπεριέχουν το συγκεκριμένο hashtag. Ακόμα οι χρήστες μπορούν να δηλώνουν ποιες αναρτήσεις τους αρέσουν(likes), καθώς και να τις δημοσιεύουν στο δικό τους προφίλ(retweets).

2.4. Λόγοι Επιλογής των Πηγών

Ανακεφαλαιώνοντας σκοπός της παρούσας διπλωματικής είναι η δημιουργία προφίλ μηχανικών λογισμικού από κοινωνικά δίκτυα, όπως προαναφέρεται. Για αυτό το λόγο επιλέχθηκαν οι τέσσερις πιο πάνω πηγές.

Τα συστήματα ελέγχου εκδόσεων που είναι το GitHub και το Bitbucket μπορούν να μας παρέχουν πληροφορίες σχετικά με την έως τώρα εργασία ενός μηχανικού λογισμικού την οποία έχει αναρτήσει στα δίκτυα αυτά, καθώς και χρήσιμα δεδομένα που αφορούν συγκεκριμένα πεδία του προγραμματισμού, όπως συγκεκριμένες γλώσσες προγραμματισμού που χρησιμοποιεί ο χρήστης, αριθμό των συνεισφορών που έχει κάνει στα αποθετήρια και δημοτικότητα. Πριν επιλεγούν τα δυο αυτά συστήματα ελέγχου εκδόσεων, έγινε μια αναζήτηση σύμφωνα με τα στοιχεία που εμφανίζονται στο προφίλ των χρηστών στο καθένα από αυτά όπως φαίνεται στον Πίνακα 2.3.

VCS	Last Activities	Repositories Contributed to/fork	Contributions Calendar	Followers-Starred	Badges	Commits by Project	Commits by Language	Most experienced in
GitHub	✓	✓	✓	✓	x	x	x	x
Bitbucket	✓	✓	✓	x	✓	✓	✓	x
OpenHub	✓	✓	x	✓	x	x	✓	✓
Codeplex	✓	✓	x	✓	x	x	x	x
SourceForge	✓	✓	x	x	x	x	x	x
Launchpad	✓	✓	x	x	✓	x	x	✓

Πίνακας 2.3 Στατιστικά προφίλ σε διάφορα VCS

Επιλέχθηκε το Stack Overflow αφού είναι πιθανόν η πιο δημοφιλής ιστοσελίδα ερωταπαντήσεων που αφορά των προγραμματισμό, η οποία περιέχει μεγάλους όγκους δεδομένων σχετικά με τη γνώση που έχουν οι χρήστες πάνω σε συγκεκριμένα θέματα. Ακόμα χρήσιμη θεωρείται η άποψη των υπολοίπων χρηστών για ένα συγκεκριμένο μηχανικό λογισμικού, η οποία εκφράζεται μέσω των θετικών και αρνητικών ψήφων και η φήμη που χτίζει ο κάθε χρήστης ανάλογα με τα επιτεύγματά του. Γενικά αυτή η πηγή μπορεί να παρέχει στοιχεία σχετικά με την εμπειρογνωμοσύνη ενός μηχανικού λογισμικού.

Τέλος σύμφωνα με τον Nielsen, οι χρήστες του διαδικτύου συνεχίζουν να ξοδεύουν περισσότερο χρόνο στα μέσα κοινωνικής δικτύωσης παρά σε άλλου είδους ιστοσελίδες. Παράλληλα, ο χρόνος που ξοδεύεται στα μέσα κοινωνικής δικτύωσης στις Η.Π.Α μέσω

ηλεκτρονικών υπολογιστών και κινητών συσκευών αυξήθηκε κατά 99 τοις εκατό σε 121 δισεκατομμύρια λεπτά τον Ιούλιο του 2012 εν συγκρίσει με τον Ιούλιο 2011 που ήταν 66 δισεκατομμύρια λεπτά [8]. Κατά συνέπεια είναι ανάγκη να μελετηθεί επιπρόσθετα και ένα κοινωνικό δίκτυο γενικής μορφής, στη δεδομένη περίπτωση το Twitter, που δεν αφορά κάποιο συγκεκριμένο θέμα όπως είναι ο προγραμματισμός, αφού οι χρήστες μπορούν να μοιραστούν το οτιδήποτε αφορά τη ζωή τους μέσα από τα tweets, έτσι μπορούν να μοιραστούν και θέματα που αφορούν τη δραστηριότητά τους στις τεχνολογίες λογισμικού.

2.5 Ετικέτες(Tags) Stack Overflow

Για την αξιοποίηση των πιο πάνω πηγών έχουν χρησιμοποιηθεί κάποιες ετικέτες οι οποίες βρίσκονται ανάμεσα στις εκατό κορυφαίες του Stack Overflow. Αυτές οι ετικέτες αφορούν γλώσσες προγραμματισμού και τεχνολογίες της πληροφορικής και θα συμβάλουν στη κατηγοριοποίηση των δεδομένων που θα συλλεγούν. Οι ετικέτες αυτές βρίσκονται στο παράρτημα Α.

Κεφάλαιο 3

Εξόρυξη Δεδομένων

3.1 Προγραμματιστικά Πλαίσια (API)	13
3.2 GitHub API	15
3.3 Bitbucket API	18
3.4 Stack Overflow API	19
3.5 Twitter API	20

3.1 Προγραμματιστικά Πλαίσια (API)

Για τη συλλογή των δεδομένων από τις προαναφερθείσες πηγές χρησιμοποιήθηκαν τα APIs της κάθε πηγής. Το προγραμματιστικό πλαίσιο είναι μια διαδικτυακή υπηρεσία που αφορά ένα σύστημα, η οποία επιτρέπει την εξόρυξη κάποιων δεδομένων που βρίσκονται στο σύστημα αυτό. Βασίζεται πάνω στην αρχιτεκτονική REST (Representational Estate Transfer) η οποία επικοινωνεί διαμέσου του πρωτοκόλλου HTTP (Hypertext Transfer Protocol) και χρησιμοποιεί το λεξικό του το οποίο ορίζεται ως εξής:

- Μέθοδοι (GET, POST, PUT, DELETE)
- HTTP URI syntax (paths, parameters, κτλ.)
- Τύποι Δεδομένων(xml, json, html, κτλ.)
- HTTP Response codes

Πιο κάτω πρόκειται να εξηγήσουμε συνοπτικά το λεξικό του HTTP πρωτοκόλλου.

Μέθοδος GET:

Η μέθοδος αυτή προνοεί την ανάκτηση οποιασδήποτε πληροφορίας η οποία ταυτοποιείται από ένα συγκεκριμένο αίτημα μέσω URI. Σε περίπτωση που το URI αναφέρεται σε μια διεργασία η οποία παράγει κάποιες συγκεκριμένες πληροφορίες, αυτές θα συμπεριληφθούν στην απάντηση που θα επιστραφεί σ' αυτόν που έστειλε το αίτημα [11].

Μέθοδος POST:

Χρησιμοποιείται για να αιτηθεί σε κάποιο διακομιστή, να αποδεχτεί και να αποθηκεύσει τα δεδομένα που περιέχονται στο αίτημα και να τα προωθήσει σε μια συγκεκριμένη διεργασία που ταυτοποιείται από το URI, που περιέχεται στο αίτημα [12].

Μέθοδος PUT:

Είναι παρόμοια με την POST αφού επίσης αιτείται σε κάποιο διακομιστή να αποδεχτεί τα δεδομένα τα οποία περιέχονται στο αίτημα και να τα προωθήσει στη διεργασία που ταυτοποιείται από το URI, το οποίο περιέχεται στο αίτημα. Αν στη διεργασία που προορίζονται τα δεδομένα, υπάρχουν είδη κάποια δεδομένα, τότε αντικαθίστανται από τα νέα δεδομένα [11].

Μέθοδος DELETE:

Η μέθοδος DELETE αιτείται στο διακομιστή να διαγράψει ένα συγκεκριμένο πόρο, ο οποίος ταυτοποιείται επίσης από το URI που περιέχεται στο αίτημα.

HTTP URI (Uniform Resource Identifiers):

Είναι επίσης γνωστό με τα ονόματα: διευθύνσεις WWW, Universal Resource Identifiers και ως συνδυασμός των Uniform Resource Locators (URL) και Names (URN). Αποτελεί απλές μορφές κειμένου οι οποίες ταυτοποιούν ένα συγκεκριμένο πόρο μέσω ονόματος, τοποθεσίας ή άλλου χαρακτηριστικού.

Τύποι Δεδομένων (MEDIA TYPES):

Το πρωτόκολλο HTTP χρησιμοποιεί τύπους δεδομένων ως τύπο περιεχομένων αιτήματος και ως τύπο αποδοχής της απάντησης, με σκοπό οι τύποι δεδομένων να είναι ανοικτοί, επεκτάσιμοι και συγκεκριμένοι για κάποια συστήματα.

RESPONSE CODES:

Σε κάθε αίτημα που πραγματοποιείται χρησιμοποιώντας τις μεθόδους του HTTP, επιστρέφεται πίσω μια απάντηση. Αυτή η απάντηση περιέχει ένα τριψήφιο κωδικό ο οποίος αντικατοπτρίζει κάποια πληροφορία σχετικά με την κατάσταση αίτημα που στάλθηκε στο διακομιστή.

Επομένως ήταν η καλύτερη λύση για εξόρυξη πληροφοριών από τα συγκεκριμένα συστήματα και για αυτό επιλέχθηκαν τα APIs. Επιπρόσθετα όλα τα APIs που χρησιμοποιήθηκαν είναι υλοποιημένα στη γλώσσα PHP και ο τύπος δεδομένων χρησιμοποιείται για την επιστροφή τους είναι ο τύπος JSON.

Τα δεδομένα που θα συλλεχθούν αφορούν γλώσσες και τεχνολογίες οι οποίες βρίσκονται ανάμεσα στις εκατό πιο δημοφιλείς ετικέτες του Stack Overflow και χρησιμοποιούνται από τη ιστοσελίδα που δημιουργήθηκε στα πλαίσια της παρούσας διπλωματικής.

Επίσης αξίζει να αναφέρουμε ότι για την επικοινωνία μεταξύ του συστήματος και των APIs, για κάποια από αυτά χρειάστηκε να δημιουργήσουμε εφαρμογές. Με τη δημιουργία των εφαρμογών αυτών δημιουργούνται και κάποια κλειδιά πρόσβασης. Οι εφαρμογές αυτές χρησιμοποιούν αυτά τα κλειδιά για ταυτοποίηση από το API, από το οποίο θα συλλέξουμε δεδομένα και επιπρόσθετα προσφέρουν μεγαλύτερο αριθμό αιτημάτων προς τα APIs. Θα αναφερθούμε πιο κάτω στις εφαρμογές που έχουμε δημιουργήσει.

3.2 GitHub API

Τα δεδομένα που αφορούν το GitHub συλλέγονται μέσω του GitHub API. Αρχικά έχουμε δημιουργήσει μια εφαρμογή στο GitHub με το όνομα Expertise Analyzer λόγω του ότι χρειάζεται να γίνει ταυτοποίηση της εφαρμογής, η οποία θα λαμβάνει τα δεδομένα και λόγω του ότι τα αιτήματα στο API είναι περιορισμένα. Με την εφαρμογή μας παρέχεται αυξημένος αριθμός αιτημάτων ο οποίος ανέρχεται στα 5000 αιτήματα ανά ώρα.

Για τη χρήση του GitHub API χρησιμοποιήθηκε το HTTPFUL. Το HTTPFUL είναι μια απλή αναγνώσιμη βιβλιοθήκη της PHP η οποία προορίζεται για επικοινωνία με το HTTP. Παρέχει την ευκολία στον χρήστη να αλληλεπιδρά με διάφορα APIs χωρίς τη χρήση της μεθόδου της PHP curl set_opt και είναι ένα ιδανικό REST Client. Το Httpful υποστηρίζει [13]:

- Readable HTTP Method Support (GET, PUT, POST, DELETE, HEAD, and OPTIONS)
- Custom Headers
- Automatic "Smart" Parsing
- Automatic Payload Serialization
- Basic Authentication
- Client Side Certificate Authentication
- Request "Templates"

Αρχικά γίνεται η ταυτοποίηση της εφαρμογής Expertise Analyzer από το API. Ακολούθως χρησιμοποιώντας το όνομα χρήστη, ενός χρήστη, στέλνονται διάφορα αιτήματα στο API. Αυτά τα αιτήματα αφορούν τα εξής δεδομένα:

- Ακόλουθοι
- Ημερομηνία δημιουργίας λογαριασμού
- Αποθετήρια κώδικα
- Κύρια γλώσσα προγραμματισμού αποθετηρίων
- Commits αρχείων κώδικα ανά χρόνο

Για την αποστολή των αιτημάτων στο API χρησιμοποιούνται οι εξής μέθοδοι που παρέχονται από αυτό:

- **GET /users/:username** : Για να ελέγξουμε αν το όνομα χρήστη υπάρχει στο GitHub.
- **GET /users/:username/repos**: Για να πάρουμε τα αποθετήρια κώδικα του χρήστη, τα οποία του ανήκουν και αποθετήρια στα οποία έχει συνεισφέρει.
- **GET /repos/:owner/:repo/commits** : Για να πάρουμε τα commits που έχει κάνει ο χρήστης σε συγκεκριμένο αποθετήριο κώδικα.

Στην εικόνα 3.1 παρουσιάζεται ο κώδικας ο οποίος χρησιμοποιείται για τη συλλογή των commits του χρήστη ανά χρονολογία.

```
for($i=0;$i<count($repos_by_name);$i++){
    $repo_name=$repos_by_name[$i]['name'];
    $repo_owner=$repos_by_name[$i]['owner'];
    $repo_language=$repos_by_name[$i]['language'];

    do{
        //Uri to get commits for a repo that user contributed to.
        $uri="https://api.github.com/repos/".$repo_owner."/".$repo_name."/commits?page=".$page.$client_id_secret;
        $response = \Httpful\Request::get($uri)->expectsJson()->send();
        $commits = json_decode($response);

        //if no commits exist break. It returns an object with a message, instead it returns an array of objects.
        if(gettype($commits)=="object"){
            break;
        }

        //For each commit in page.
        for($j=0;$j<count($commits);$j++){
            //Some author fields are returned NULL from Github API.
            if($commits[$j]->{"author"}==null){
                continue;
            }

            if(strcmp($commits[$j]->{"author"}->{"login"},$user_id)==0){
                //get the year of commit
                $year = $commits[$j]->{"commit"}->{"author"}->{"date"};
                $year= split('-', $year);
                $year=strval($year[0]);
                $commits_counter[$year][$repo_language]++;
            }
        }
        $page++;
    }while(count($commits)>0);
}
```

Εικόνα 3.0.1 Συλλογή Commits από GitHub

Άρα χρησιμοποιώντας τα πιο πάνω δεδομένα εξάγονται τα τελικά δεδομένα από το GitHub, τα οποία θα χρησιμοποιηθούν από την ιστοσελίδα για τη δημιουργία του προφίλ.

Αυτά τα δεδομένα είναι:

- Ακόλουθοι
- Διάρκεια λογαριασμού: Μετριέται σε μέρες
- Διάρκεια δραστηριότητας: Είναι η διαφορά μεταξύ της ημερομηνίας δημιουργίας λογαριασμού και της ημερομηνίας δημιουργίας του πρώτου αποθετηρίου. Μετριέται σε μέρες
- Αποθετήρια κώδικα ανά γλώσσα
- Αριθμός αποθετηρίων κώδικα
- Commits αρχείων κώδικα ανά χρόνο και ανά γλώσσα: Η γλώσσα του αρχείου είναι αυτή του αποθετηρίου στο οποίο έχει παραδοθεί το αρχείο.

Στο παράρτημα Α υπάρχει ο πλήρης κώδικας για συλλογή δεδομένων χρησιμοποιώντας το GitHub API.

3.3 Bitbucket API

Στη συνέχεια χρησιμοποιήθηκε το Bitbucket API για εξόρυξη δεδομένων από το Bitbucket. Είναι επίσης βασισμένο στην αρχιτεκτονική REST όπως και το GitHub API. Στο σημείο αυτό δεν ήταν απαραίτητη η δημιουργία εφαρμογής στο Bitbucket, αφού το όριο των αιτημάτων είναι το ίδιο είτε γίνει ταυτοποίηση της εφαρμογής είτε όχι. Να αναφέρουμε ότι η ταυτοποίηση ενός χρήστη γίνεται με το όνομα χρήστη που διατηρεί στο Bitbucket.

Για τη χρήση του Bitbucket API δεν χρησιμοποιήθηκε κάποια συγκεκριμένη βιβλιοθήκη παρά μόνο οι συναρτήσεις της **PHP file_get_contents** για το κάλεσμα των κατάλληλων μεθόδων του API και η **json_decode** για αποκωδικοποίηση της απάντησης σε μορφή **JSON** από το API.

Για την αποστολή των αιτημάτων χρησιμοποιούνται οι εξής μέθοδοι που είναι διαθέσιμες από το API:

- **GET https://api.bitbucket.org/2.0/users/{username_or_uuid}** : Για να ελέγξουμε αν ο χρήστης υπάρχει στο σύστημα.

- **GET https://api.bitbucket.org/2.0/repositories/{username_or_uuid}** : Για να πάρουμε τα αποθετήρια κώδικα του χρήστη ανά γλώσσα, τον αριθμό των αποθετηρίων και τα commits που έχει κάνει ανά γλώσσα.
- **GET https://api.bitbucket.org/2.0/users/{username_or_uuid}/followers** : Για να πάρουμε τους ακόλουθους του χρήστη.

Έτσι τα πιο πάνω δεδομένα που συλλέγονται από το API, χωρίς περαιτέρω επεξεργασία είναι και τα τελικά που θα χρησιμοποιηθούν από την ιστοσελίδα για τη δημιουργία του προφίλ του χρήστη. Συγκεντρωτικά είναι τα εξής:

- Ακόλουθοι
- Αποθετήρια κώδικα ανά γλώσσα
- Αριθμός αποθετηρίων κώδικα
- Commits αρχείων κώδικα ανά χρόνο και ανά γλώσσα: Η γλώσσα του αρχείου είναι αυτή του αποθετηρίου στο οποίο έχει παραδοθεί το αρχείο.

Στο παράρτημα Α υπάρχει ο πλήρης κώδικας για συλλογή δεδομένων χρησιμοποιώντας το Bitbucket API.

3.4 Stack Overflow API

Μετά από αυτό ακολουθεί το API του Stack Overflow. Το API αυτό υπόκειται στο Stack Exchange API από το οποίο μπορεί να επιλεγεί συγκεκριμένη σελίδα για συλλογή στοιχείων, στη περίπτωση αυτή το Stack Overflow. Επίσης έχει δημιουργηθεί εφαρμογή με το όνομα Expertise Analyzer για τη χρήση του API και μαζί δημιουργείτε και ένα κλειδί ταυτοποίησης. Έτσι παρέχεται μεγαλύτερος αριθμός αιτημάτων, αφού γίνει ταυτοποίηση.

Για τη χρήση του API χρησιμοποιήθηκε η Stack.PHP η οποία είναι μια βιβλιοθήκη της PHP για πρόσβαση στο API του Stack Exchange. Μερικές από τις επιλογές που προσφέρει είναι οι πιο κάτω [14]:

- Παρέχει πλήρη υλοποίηση του Stack Exchange API.
- Παρέχει ευελιξία όσο αφορά το caching: Χρησιμοποιεί κάποιες τεχνικές όσο αφορά το caching ώστε να μειώνει τον αριθμό των αιτημάτων που αποστέλλονται στο API.
- Παρέχει ευκολία στη προσπέλαση των ανταποκρίσεων από το API, οι οποίες είναι σελιδοποιημένες.

Γενικά παρέχει όλες τις επιλογές που περιέχει και το API.

Η ταυτοποίηση των χρηστών γίνεται μέσω του Stack Overflow ID το οποίο καθιστά ένα χρήστη μοναδικό. Για την αποστολή των αιτημάτων χρησιμοποιούνται οι εξής μέθοδοι του Stack.PHP:

- **API::\$key** : Ταυτοποίηση της εφαρμογής Expertise Analyzer με βάση το κλειδί του.
- **API::Site('stackoverflow')** : Επιλογή σελίδας του Stack Exchange, στη περίπτωση αυτή είναι το Stack Overflow.
- **Users()->SortByCreation()->Exec()** : Επιστρέφει τους χρήστες ταξινομημένους με την ημερομηνία δημιουργίας του λογαριασμού. Αυτό γίνεται για έλεγχο του ID. Σε περίπτωση που χρειαστεί να αναζητηθεί ένας χρήστης με μεγαλύτερο ID από τον πιο νέο χρήστη του Stack Overflow, τότε σημαίνει ότι αυτός ο χρήστης δεν υπάρχει.
- **Users(\$user_id)->TopAnswerTags()->Exec()** : Επιστρέφει πληροφορίες από τις οποίες συλλέγονται ο αριθμός απαντήσεων και αριθμός θετικών ψήφων, όσο αφορά τις απαντήσεις του χρήστη ανά ετικέτα, ξεκινώντας από την κορυφαία του ετικέτα(αυτή που έχει τις περισσότερες απαντήσεις) και συνεχίζοντας κατά φθίνουσα σειρά.
- **Users(\$user_id)->Exec()** : Ανάκτηση πληροφοριών ενός χρήστη με βάση το ID του στο Stack Overflow. Από αυτές τις πληροφορίες συλλέγονται τα χρυσά, αργυρά και χάλκινα εμβλήματα που έχει ο χρήστης, όπως και η φήμη του στο Stack Overflow.
- **Users(\$user_id)->Badges()->Exec()** : Ανάκτηση πληροφοριών ενός χρήστη με βάση το ID του, όσο αφορά τα εμβλήματα του στις γλώσσες και τεχνολογίες που επιλέγηκαν για την υλοποίηση της ιστοσελίδας της διπλωματικής αυτής.

Δεδομένα που συλλέγονται από το Stack Overflow:

- Φήμη
- Χρυσά εμβλήματα ανά ετικέτα(που είναι ανάμεσα στις ετικέτες οι οποίες χρησιμοποιούνται από την ιστοσελίδα της διπλωματικής)
- Αργυρά εμβλήματα ανά ετικέτα(που είναι ανάμεσα στις ετικέτες οι οποίες χρησιμοποιούνται από την ιστοσελίδα της διπλωματικής)
- Χάλκινα εμβλήματα ανά ετικέτα(που είναι ανάμεσα στις ετικέτες οι οποίες χρησιμοποιούνται από την ιστοσελίδα της διπλωματικής)
- Συνολικός αριθμός εμβλημάτων

Στην εικόνα 3.2 παρουσιάζεται ο κώδικας για συλλογή της φήμης και των εμβλημάτων.

```

//Query to get a user object.
$response = $stackoverflow->Users($user_id)->Exec();
$user=$response->Fetch(FALSE);
//Get user's badges and reputation.
$bronze_badges = $user["badge_counts"]["bronze"];
$silver_badges = $user["badge_counts"]["silver"];
$gold_badges = $user["badge_counts"]["gold"];

$reputation = $user["reputation"];

$response=$stackoverflow->Users($user_id)->Badges()->Exec();
while($array=$response->Fetch()){
    $badge=strtoupper($array["name"]);
    $rank=$array["rank"];

    if(in_array($badge,$languages_keys))
        $badges[$badge].=$rank;
}

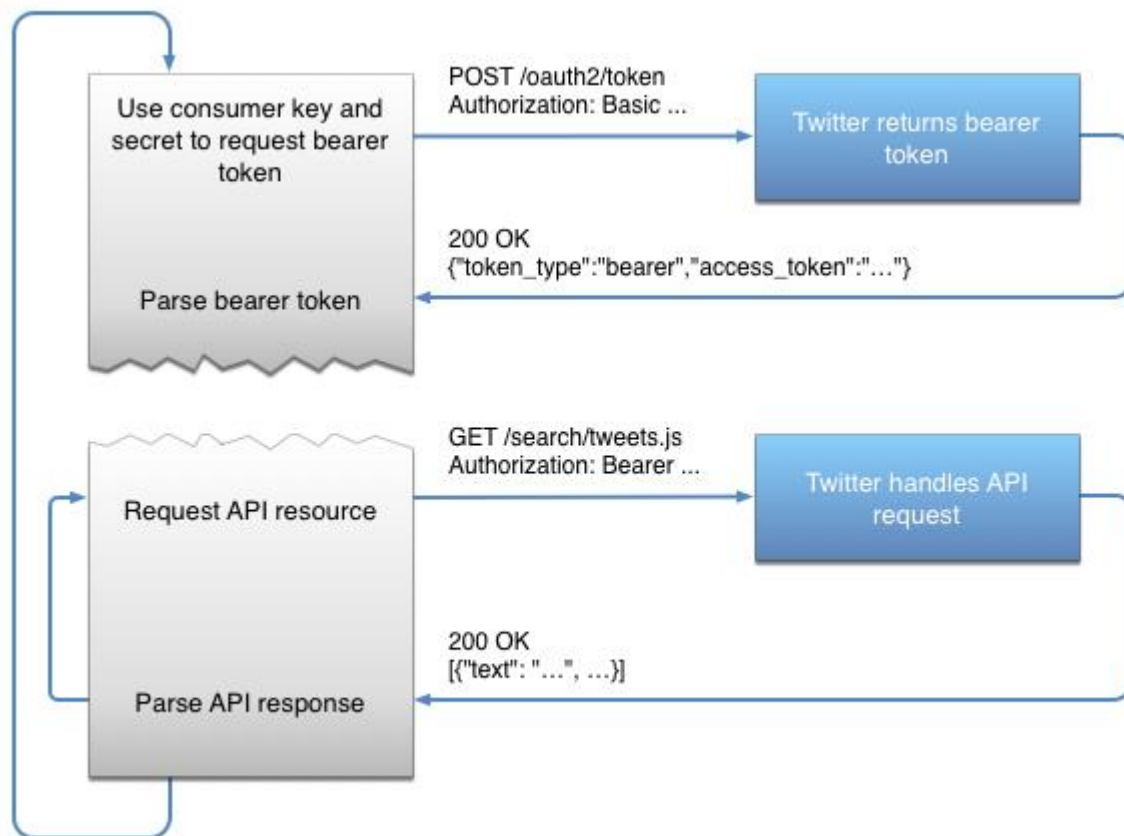
```

Εικόνα 3.0.2 Συλλογή φήμης και εμβλημάτων από Stack Overflow

3.5 Twitter API

Τέλος για τη συλλογή δεδομένων από το Twitter, χρησιμοποιείται το Twitter REST API , το οποίο είναι και αυτό τύπου REST. Το Twitter API παρέχει πρόσβαση για διάβασμα και γράψιμο όσο αφορά δεδομένα του Twitter. Όπως για παράδειγμα αναρτήσεις tweet και ανάγνωση προφίλ ενός χρήστη [15].

Πρέπει να αναφέρουμε ότι για ταυτοποίηση στο Twitter API έχουμε δημιουργήσει εφαρμογή με το όνομα «**Diplwmatiki**» μέσω της οποίας γίνονται οι κλήσεις. Στο Twitter είναι απαραίτητο να δημιουργηθεί εφαρμογή για τη χρησιμοποίηση του API. Μπορούμε να δούμε στην εικόνα 4.1 τον τρόπο με τον οποίο γίνεται η ταυτοποίηση [16]. Αρχικά στέλνεται αίτημα στο twitter το οποίο περιέχει το consumer key και secret, τα οποία δημιουργούνται μαζί με την εφαρμογή και αφού γίνει η ταυτοποίηση από το API επιστρέφεται το bearer access token το οποίο επιτρέπει τη πρόσβαση στο API για συλλογή δεδομένων.



Εικόνα 4.0.3 Ταυτοποίηση Εφαρμογής από το Twitter API

Για χρήση του Twitter API χρησιμοποιήθηκε η βιβλιοθήκη της PHP TwitterOAuth. Περιέχει όλες τις μεθόδους από το Twitter API και διευκολύνει το κάλεσμα τους μέσα από τον κώδικα της PHP. Από τη συγκεκριμένη βιβλιοθήκη χρησιμοποιούνται οι εξής μέθοδοι:

- **new TwitterOAuth(CONSUMER_KEY, CONSUMER_SECRET, ACCESS_TOKEN, ACCESS_TOKEN_SECRET)** : Οι παράμετροι δημιουργούνται αυτόματα από την εφαρμογή και χρησιμοποιούνται για ταυτοποίηση από το Twitter API.
- **get("statuses/user_timeline", array("screen_name" => \$screen_name, "count" => 200, "page"=>\$page))** : Επιστροφή μέχρι 3200 από τα πιο πρόσφατα tweets που βρίσκονται στο προφίλ ενός συγκεκριμένου χρήστη.
- **user->followers_count** : Συλλογή ακολούθων χρήστη.
- **\$status->text**: Συλλογή του κειμένου ενός tweet. Εξαιρούνται τα tweets τα οποία έχει κάνει retweet ο χρήστης από άλλα προφίλ.
- **\$status->retweet_count**: Αριθμός retweets ενός tweet.
- **\$status->favorite_count**: Αριθμός likes ενός tweet.

Συγκεντρωτικά όλα τα δεδομένα που συλλέγονται είναι τα εξής:

- Ακόλουθοι χρήστη.
- Tweets ανά γλώσσα προγραμματισμού ή τεχνολογία.
- Retweets ανά γλώσσα προγραμματισμού ή τεχνολογία.
- Likes ανά γλώσσα προγραμματισμού ή τεχνολογία.

Αξίζει να αναφέρουμε ότι για τη συλλογή των tweets ενός χρήστη χρησιμοποιείται το λεξικό της Computer Hope με προγραμματιστικούς όρους. Η Computer Hope είναι μια ιστοσελίδα δωρεάν τεχνικής υποστήριξης και πληροφόρησης [17]. Αναλυτικότερα το λεξικό χρησιμοποιείται για να μπορούν να εντοπίζονται tweets τα οποία αφορούν τις γλώσσες προγραμματισμού και τεχνολογίες για τις οποίες συλλέγονται δεδομένα στη παρούσα διπλωματική. Αποτελείται από 575 λέξεις και κάποιες από τις λέξεις που περιέχονται σε αυτό είναι το «while», «class», «JSON», «Hash» και πολλές άλλες. Για να γίνει πιο σαφές εντοπίζονται τα tweets και εφόσον περιέχουν μια γλώσσα ή τεχνολογία που μας ενδιαφέρει, γίνεται έλεγχος αν περιέχει τουλάχιστον και άλλη μια λέξη σχετική με τον προγραμματισμό, η οποία να περιέχεται στο πιο πάνω λεξικό. Αυτό γίνεται εξαιτίας του ότι μερικές από τις ετικέτες που μελετούμε έχουν και άλλες έννοιες εκτός από προγραμματιστικές. Επομένως θέλουμε να αποκλείσουμε το ενδεχόμενο να συλλέξουμε tweets που δεν σχετίζονται με το θέμα μας, σε όσο μεγαλύτερο βαθμό γίνεται, παρόλο που ίσως να χάνονται κάποια tweets που μας ενδιαφέρουν. Για αυτό το λόγο ακολουθείται η συγκεκριμένη προσέγγιση.

Στο παράρτημα Α υπάρχει ο πλήρης κώδικας για συλλογή δεδομένων χρησιμοποιώντας το Twitter API.

Κεφάλαιο 4

Στατιστικά – Βαθμολογίες Χρήστη

4.1 Στατιστικά από GitHub	23
4.2 Στατιστικά από Bitbucket	24
4.3 Στατιστικά από Stack Overflow	25
4.4 Στατιστικά από Twitter	25
4.5 Βαθμολογίες Χρήστη	26
4.6 Σύγκριση με Προηγούμενες Προσεγγίσεις	32

4.1 Στατιστικά από GitHub

Στην παρουσίαση του προφίλ στον χρήστη εμφανίζονται κάποια στατιστικά από το GitHub. Αυτά τα στατιστικά συλλέγονται από το API στο οποίο αναφερθήκαμε στο προηγούμενο κεφάλαιο. Κάποια παρουσιάζονται με τη χρήση γραφικών παραστάσεων και άλλα με τη μορφή πινάκων. Τα στατιστικά από το GitHub είναι τα εξής:

- **Αριθμός Αποθετηρίων (Repositories):** Είναι τα αποθετήρια που ανήκουν στον χρήστη και έχουν μια συγκεκριμένη γλώσσα προγραμματισμού. Επίσης στα αποθετήρια του χρήστη συγκαταλέγονται και αυτά στα οποία ο χρήστης έχει συνεισφέρει. Τα αποθετήρια τα οποία δεν χαρακτηρίζονται από κάποια γλώσσα από το σύστημα του GitHub ή αυτά που είναι άδεια δεν υπολογίζονται μέσα στα αποθετήρια του χρήστη.
- **Διάρκεια Εμπλοκής (Involvement's Duration):** Είναι η διάρκεια σε μέρες από την μέρα που ο χρήστης δημιούργησε το πρώτο αποθετήριο έως την ημέρα που δημιούργησε το προφίλ του στην ιστοσελίδα Expertise Analyzer (δηλ. που θα συμπίπτει με τη σημερινή).
- **Διάρκεια Λογαριασμού (Account's Duration):** Είναι η διάρκεια σε μέρες από την μέρα που δημιούργησε ο χρήστης το λογαριασμό του έως τη μέρα που δημιούργησε το προφίλ του στην ιστοσελίδα Expertise Analyzer (δηλ. που θα συμπίπτει με τη σημερινή).

- **Ακόλουθοι (Followers):** Είναι ο αριθμός των ακολούθων που έχει ο χρήστης στο λογαριασμό του στο Github.
- **Commits αρχείων:** Είναι ο αριθμός των commits που έχει κάνει ο χρήστης. Παρουσιάζονται παραστατικά σε γραφική παράσταση όπου αναγράφεται η χρονολογία του commit όπως και η γλώσσα προγραμματισμού.
- **Κορυφαίες επτά γλώσσες (Top 7 Languages):** Είναι οι κορυφαίες επτά γλώσσες που χρησιμοποίησε ο χρήστης. Όταν λέμε κορυφαίες εννοούμε αυτές στις οποίες ανήκουν τα περισσότερα αποθετήρια του χρήστη. Παρουσιάζονται σε γραφική παράσταση.
- **Ποσοστά Γλωσσών (Languages' Percentages):** Είναι οι γλώσσες προγραμματισμού που ανήκουν τα αποθετήρια του χρήστη και εμφανίζονται κατά ποσοστό σε γραφική παράσταση κυκλικής μορφής (pie).

4.2 Στατιστικά από Bitbucket

Ακολούθως έχουμε τα στατιστικά του χρήστη, από τις πληροφορίες που συλλέγονται από το Bitbucket API. Στα στατιστικά αυτά συμπεριλαμβάνονται τα:

- **Αποθετήρια:** Τα αποθετήρια τα οποία έχει δημιουργήσει ο χρήστης στο λογαριασμό του και ανήκουν σε μια συγκεκριμένη γλώσσα προγραμματισμού.
- **Ακόλουθοι:** Είναι ο αριθμός των ακολούθων οι οποίοι ακολουθούν τον χρήστη.
- **Commits:** Είναι τα commits σε αρχεία κώδικα τα οποία ο χρήστης παρέδωσε σε κάποιο αποθετήριο. Εμφανίζονται σε γραφική παράσταση ανά χρονολογία και γλώσσα προγραμματισμού.
- **Κορυφαίες επτά γλώσσες:** Είναι τα αποθετήρια του χρήστη ανά γλώσσα προγραμματισμού και εμφανίζονται σε γραφική παράσταση οι κορυφαίες επτά γλώσσες που έχει χρησιμοποιήσει.
- **Ποσοστά Γλωσσών:** Είναι οι γλώσσες προγραμματισμού που ανήκουν τα αποθετήρια του χρήστη και εμφανίζονται κατά ποσοστό σε γραφική παράσταση κυκλικής μορφής (pie).

4.3 Στατιστικά από Stack Overflow

Τα επόμενα στατιστικά που θα εξηγήσουμε αφορούν το Stack Overflow. Έχουν συλλεχθεί από το API του Stack Overflow. Αυτά τα στοιχεία σχετίζονται με τα δεδομένα που διατηρεί ο χρήστης στο σύστημα αυτό τα οποία είναι τα εξής:

- **Φήμη (Reputation):** Είναι ένας βαθμός ο οποίος όπως εξηγήσαμε και προηγουμένως αυξάνεται όσο πιο μεγάλη είναι η δραστηριότητα του χρήστη στο σύστημα.
- **Εμβλήματα (Badges):** Είναι ο αριθμός των εμβλημάτων που μπορεί να κερδίσει ο χρήστης ανάλογα με τη δραστηριότητα του σε ένα συγκεκριμένο θέμα στο σύστημα. Χωρίζονται στις κατηγορίες χρυσά, αργυρά και χάλκινα.
- **Απαντήσεις:** Είναι οι απαντήσεις που έχει δώσει ο χρήστης οι οποίες αφορούν τις συγκεκριμένες ετικέτες του Stack Overflow που μελετώνται στη διπλωματική αυτή. Παρουσιάζονται σε κυκλική γραφική παράσταση τα ποσοστά ανά ετικέτα, όπως και σε πίνακα ο αριθμός των απαντήσεων ανά ετικέτα.
- **Θετικές Ψήφοι (Upvotes):** Είναι οι θετικές ψήφοι που έδωσαν άλλοι χρήστες σε απαντήσεις που έχει δώσει ο χρήστης οι οποίες αφορούν τις συγκεκριμένες ετικέτες του Stack Overflow που μελετώνται στη διπλωματική αυτή. Παρουσιάζονται σε κυκλική γραφική παράσταση τα ποσοστά ανά ετικέτα, όπως και σε πίνακα ο αριθμός των θετικών ψήφων ανά ετικέτα.
- **Κορυφαία εμβλήματα:** Είναι τα κορυφαία εμβλήματα ανά ετικέτα. Όταν λέμε κορυφαία εννοούμε τα εμβλήματα με την υψηλότερη κατηγορία με φθίνουσα σειρά χρυσό, αργυρό, χάλκινο.

4.4 Στατιστικά από Twitter

Όπως και για τα δυο προηγούμενα συστήματα, έτσι και για το Twitter παρουσιάζονται κάποια στατιστικά που συλλέγηκαν από το API του συγκεκριμένου συστήματος. Ανάμεσα σε αυτά τα στατιστικά βρίσκονται τα παρακάτω:

- **Ακόλουθοι:** Είναι ο αριθμός των χρηστών οι οποίοι ακολουθούν το συγκεκριμένο χρήστη στο Twitter.
- **Tweets:** Είναι ο αριθμός των tweets που έχει κάνει ο χρήστης και αφορούν μια συγκεκριμένη γλώσσα προγραμματισμού.
- **Retweets:** Ο αριθμός των επαναλήψεων των tweets του χρήστη που αφορούν συγκεκριμένη γλώσσα προγραμματισμού.
- **Likes:** Ο αριθμός των likes από άλλους χρήστες στα tweets που αφορούν συγκεκριμένη γλώσσα προγραμματισμού.
- **Ποσοστά Tweets:** Είναι τα ποσοστά των tweets ανά γλώσσα προγραμματισμού και εμφανίζονται σε γραφική παράσταση κύκλου, όπως και σε πίνακα με τον αριθμό των tweets ανά γλώσσα.

- **Ποσοστά Retweets:** Είναι τα ποσοστά των retweets ανά γλώσσα προγραμματισμού και εμφανίζονται σε γραφική παράσταση κύκλου, όπως και σε πίνακα με των αριθμό των retweets ανά γλώσσα.
- **Ποσοστά Likes:** Είναι τα ποσοστά των likes ανά γλώσσα προγραμματισμού και εμφανίζονται σε γραφική παράσταση κύκλου, όπως και σε πίνακα με των αριθμό των likes ανά γλώσσα.

4.5 Βαθμολογίες Χρήστη

Πέραν από τα προηγούμενα στατιστικά, έχουμε δημιουργήσει και κάποιες συγκεντρωτικές βαθμολογίες του χρήστη. Οι βαθμολογίες αυτές υπολογίζονται με βάση τα στατιστικά που περιγράψαμε πιο πριν. Κάποιες είναι γενικές και κάποιες άλλες υπολογίζονται συγκριτικά από το σύστημα. Οι συγκεντρωτικές βαθμολογίες συνεισφέρουν στο να υπάρχει μια πιο συνολική εικόνα του χρήστη από τη δραστηριότητά του στα διάφορα κοινωνικά δίκτυα. Προσφέρουν με τον τρόπο αυτό πρόσθετη πληροφόρηση σε σχέση με τα μεμονωμένα δίκτυα.

Ας τις εξετάσουμε πιο κάτω:

- **Overall Popularity:** Είναι φήμη η οποία υπολογίζεται χρησιμοποιώντας κάποια στατιστικά από το χρήστη σε σχέση με τις μέγιστες τιμές αυτών των στατιστικών.

Τύπος:

ΠΗΓΕΣ	ΠΕΔΙΑ ΠΗΓΩΝ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΕΔΙΟ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΗΓΗ
Github	Followers/GitHub MAX Followers	100%	33%
Bitbucket	Followers/Bitbucket MAX Followers	100%	33%
Stack Overflow	Reputation/ Stack Overflow MAX Reputation	100%	33%

Πίνακας 4.1 Ποσοστά Υπολογισμού Overall Popularity

Αυτή η βαθμολογία έχει δημιουργηθεί για να διακρίνεται το πόσο δημοφιλής είναι ο χρήστης σε σχέση με τους πιο δημοφιλείς χρήστες του κάθε συστήματος. Για αυτό και για να υπολογιστεί χρησιμοποιούνται μετρικές που αφορούν μόνο τη δημοτικότητα όπως είναι η ακόλουθοι και η φήμη. Να επισημάνουμε ότι αν κάποιος λογαριασμός δεν έχει δοθεί, τότε δε λαμβάνεται υπόψη στην βαθμολογία. Επίσης το

Twitter δεν θα μπορούσαμε να το συμπεριλάβουμε σε αυτή τη βαθμολογία επειδή δεν μπορεί να γίνει εύρεση του μηχανικού λογισμικού με τους περισσότερους ακόλουθους.

- **Expertise Analyzer Popularity:** Είναι η φήμη η οποία υπολογίζεται χρησιμοποιώντας κάποια στατιστικά από το χρήστη σε σχέση με τις μέγιστες τιμές αυτών των στατιστικών από το σύστημα Expertise Analyzer.

Τύπος:

ΠΗΓΕΣ	ΠΕΔΙΑ ΠΗΓΩΝ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΕΔΙΟ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΗΓΗ
Github	Followers/System MAX Followers	100%	25%
Bitbucket	Followers/System MAX Followers	100%	25%
Stack Overflow	Reputation/System Overflow MAX Reputation	100%	25%
Twitter	Followers/System MAX Followers	100%	25%

Πίνακας 4.2 Ποσοστά Υπολογισμού Expertise Analyzer Popularity

Η βαθμολογία αυτή υπολογίζεται επίσης με τις ίδιες μετρικές, όπως και η προηγούμενη, με τη μόνη διαφορά να είναι ότι οι δημοφιλέστεροι χρήστες που λαμβάνονται υπόψη στη βαθμολογία επιλέγονται μόνο από τους χρήστες που περιέχονται στο σύστημα που δημιουργήσαμε. Επομένως σε αυτή τη περίπτωση χρησιμοποιείται και το Twitter στη βαθμολογία αφού είναι δυνατόν να γνωρίζουμε τον πιο δημοφιλή χρήστη. Να αναφέρουμε ότι αν κάποιος λογαριασμός δεν έχει δοθεί, τότε δεν υπολογίζεται στη βαθμολογία.

- **Total Repositories:** Είναι τα συνολικά αποθετήρια του χρήστη.

Τύπος:

Total Repositories =

GitHub repositories + Bitbucket repositories

Υπολογίζονται αθροίζοντας τα αποθετήρια του GitHub και τα αποθετήρια του Bitbucket. Αυτό το στατιστικό μας δίνει μια εικόνα για το πόσα έργα(projects) έχει

δημιουργήσει ή δημοσιεύσει ο χρήστης και το πόσο ενεργός είναι στα συστήματα αυτά.

- **Expertise:** Είναι η γενική εμπειρογνωμοσύνη του χρήστη σε διάφορους τομείς.

Τύπος:

ΠΗΓΕΣ	ΠΕΔΙΑ ΠΗΓΩΝ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΕΔΙΟ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΗΓΗ
Github	Followers/System MAX Followers	50%	33,32%
	Account Duration/System MAX Account Duration	50%	
Bitbucket	Followers/System MAX Followers	100%	16,66%
Stack Overflow	Gold Badges/ System MAX Gold Badges	33%	49.98%
	Silver Badges/ System MAX Silver Badges	33%	
	Bronze Badges/ System MAX Bronze Badges	33%	

Πίνακας 4.3 Ποσοστά Υπολογισμού Expertise

Στον υπολογισμό της βαθμολογίας αυτής περιλαμβάνονται μετρικές που σχετίζονται με μετρικές από τρία διαφορετικά συστήματα. Η εμπειρογνωμοσύνη υπολογίζεται με βάση τα δεδομένα για κάθε χρήστη και τις κορυφαίες τιμές χρηστών για κάθε μετρική, που υπάρχουν στο σύστημα που δημιουργήσαμε. Αναφορικά με τη περίπτωση που κάποιος λογαριασμός δεν έχει δοθεί από το χρήστη τότε δε λαμβάνεται υπόψη στη βαθμολογία. Στον Πίνακα 4.1 φαίνεται η κλίμακα βαθμολόγησης της εμπειρογνωμοσύνης ενός μηχανικού λογισμικού με βάση το σκορ που θα υπολογιστεί.

Expertise Scale	Score(%)
Beginner	0-24
Intermediate	25-74
Veteran	75-100

Πίνακας 4.4 Κλίμακα Βαθμολόγησης

Για να δώσουμε ένα παράδειγμα, ας υποθέσουμε ότι ένας χρήστης έχει τα εξής δεδομένα σε σχέση με τα δεδομένα των κορυφαίων χρηστών από το σύστημα(Δεδομένα Χρήστη/ Κορυφαία Τιμή από το Σύστημα - με έντονα γράμματα): GitHub Repositories=8/**10**, GitHub account duration=85/**100**, Bitbucket Repositories=7/**10**, Stack Overflow Gold Badges=50/**60**, Stack Overflow Silver Badges=45/**50**, Stack Overflow Bronze Badges=40/**50**. Το αποτέλεσμα από τον υπολογισμό των πράξεων είναι 0.81 ή 81%. Άρα αντιλαμβανόμαστε ξεκάθαρα ότι ο συγκεκριμένος χρήστης με βάση το σκορ του ανήκει στην κατηγορία Veteran.

- **Most experienced in:** Είναι το πεδίο στο οποίο ο χρήστης διαθέτει την περισσότερη εμπειρία.

Τύπος:

ΠΗΓΕΣ	ΠΕΔΙΑ ΠΗΓΩΝ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΕΔΙΟ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΗΓΗ
Github	Repositories	100%	15%
Bitbucket	Repositories	100%	15%
Stack Overflow	Answers	50%	40%
	Upvotes	50%	
Twitter	Tweets	40%	30%
	Retweets	30%	
	Likes	30%	

Πίνακας 4.5 Ποσοστά Υπολογισμού Most Experienced in

Στη βαθμολογία αυτή λαμβάνονται υπόψη δεδομένα από όλα τα κοινωνικά δίκτυα

που έχουμε αναφέρει για κάθε γλώσσα προγραμματισμού και τεχνολογία συγκεκριμένα. Αφού γίνουν όλοι οι υπολογισμοί, εντοπίζεται η γλώσσα ή τεχνολογία με την πιο μεγάλη τιμή, άρα είναι και αυτή στην οποία ο χρήστης είναι πιο έμπειρος.

- **Knowledge:** Είναι η γνώση του χρήστη όσο αφορά συγκεκριμένα πεδία.

Υπολογίζεται με βάση το Stack Overflow.

Τύπος:

ΠΗΓΕΣ	ΠΕΔΙΑ ΠΗΓΩΝ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΕΔΙΟ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΗΓΗ
Github	Repositories	100%	15%
Bitbucket	Repositories	100%	15%
Stack Overflow	Answers	50%	40%
	Upvotes	50%	
Twitter	Tweets	40%	30%
	Retweets	30%	
	Likes	30%	

Πίνακας 4.6 Ποσοστά Υπολογισμού Knowledge

Οι παράμετροι που χρησιμοποιούνται στον υπολογισμό της συγκεκριμένης βαθμολογίας είναι αποκλειστικά και μόνο από το Stack Overflow επειδή αφορά καθαρά τις γνώσεις που απέκτησε ο χρήστης. Επομένως μέσα από τις απαντήσεις που έχει δώσει στο Stack Overflow μπορεί να διαφανεί η γνώση που έχει μοιραστεί με άλλους χρήστες.

- **Social Activity:** Είναι η κοινωνική δραστηριότητα του χρήστη για κάθε συγκεκριμένη γλώσσα προγραμματισμού ή τεχνολογία.

Τύπος:

ΠΗΓΕΣ	ΠΕΔΙΑ ΠΗΓΩΝ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΕΔΙΟ	ΠΟΣΟΣΤΟ ΑΝΑ ΠΗΓΗ
Stack Overflow	Answers/System MAX Answers	50%	50%
	Upvotes/System MAX Upvotes	50%	
Twitter	Tweets/System MAX Tweets	33%	50%
	Retweets/System MAX Retweets	33%	
	Likes/System MAX Likes	33%	

Πίνακας 4.7 Ποσοστά Social Activity

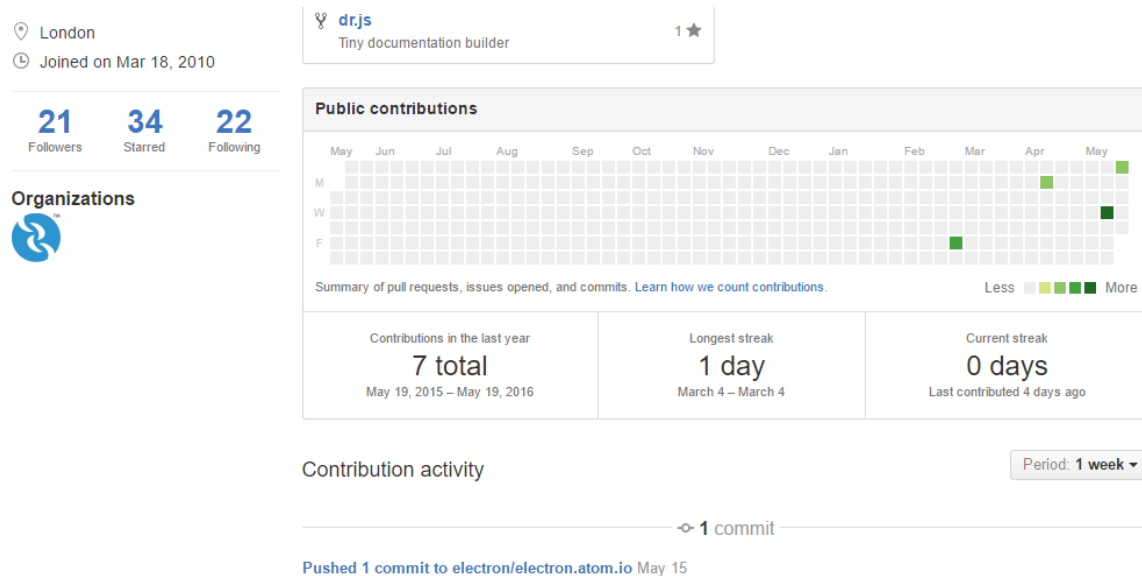
Τέλος η κοινωνική δραστηριότητα έχει ως στόχο να παρουσιάσει σε ποιο βαθμό είναι ενεργός ο χρήστης σε κάθε γλώσσα προγραμματισμού και τεχνολογία. Πιο συγκεκριμένα κατά πόσο δραστηριοποιείται στα δυο συγκεκριμένα συστήματα Stack Overflow και Twitter, περιλαμβάνοντας στη βαθμολογία τις δραστηριότητες που έκανε. Αν ένας λογαριασμός δεν έχει δοθεί από το χρήστη, τότε η τιμές που αφορούν το συγκεκριμένο λογαριασμό αντιστοιχούν σε μηδέν. Αυτό συμβαίνει επειδή δεν μπορεί να έχει παρόμοια ή ίδια βαθμολογία με ένα χρήστη ο οποίος χρησιμοποιεί περισσότερους λογαριασμούς.

4.6 Σύγκριση με Προηγούμενες Προσεγγίσεις

Διάφορα στατιστικά στοιχεία, όπως και κάποιες πληροφορίες, εμφανίζονται και στα προφίλ των χρηστών στα συστήματα GitHub, Bitbucket, Stack Overflow, Twitter, τα οποία χρησιμοποιήθηκαν στη διπλωματική αυτή αλλά και το Open HUB το οποίο δεν έχει χρησιμοποιηθεί.

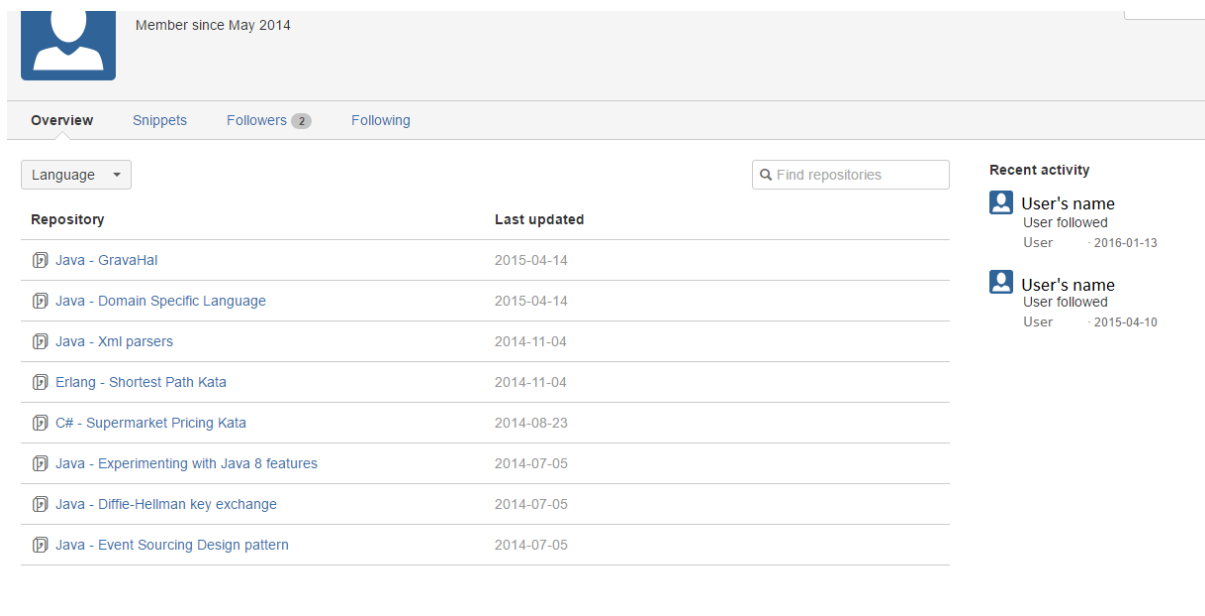
Το GitHub παρουσιάζει τα αποθετήρια του χρήστη, τα commits που έχει κάνει σε γραφική παράσταση και τους ακόλουθους. Εμφανίζει κάποιες γενικές πληροφορίες για το χρήστη όπως την ημερομηνία εγγραφής του στο σύστημα, μια μικρή παράγραφο για τον ίδιο και τους

οργανισμούς στους οποίους ανήκει. Όπως μπορούμε να δούμε και από την εικόνα 4.8 εκτός από τα πιο πάνω στοιχεία, εμφανίζει και τις τελευταίες δραστηριότητες που έχει πραγματοποιήσει ο χρήστης, όπως για παράδειγμα ένα commit το οποίο έχει κάνει πρόσφατα.



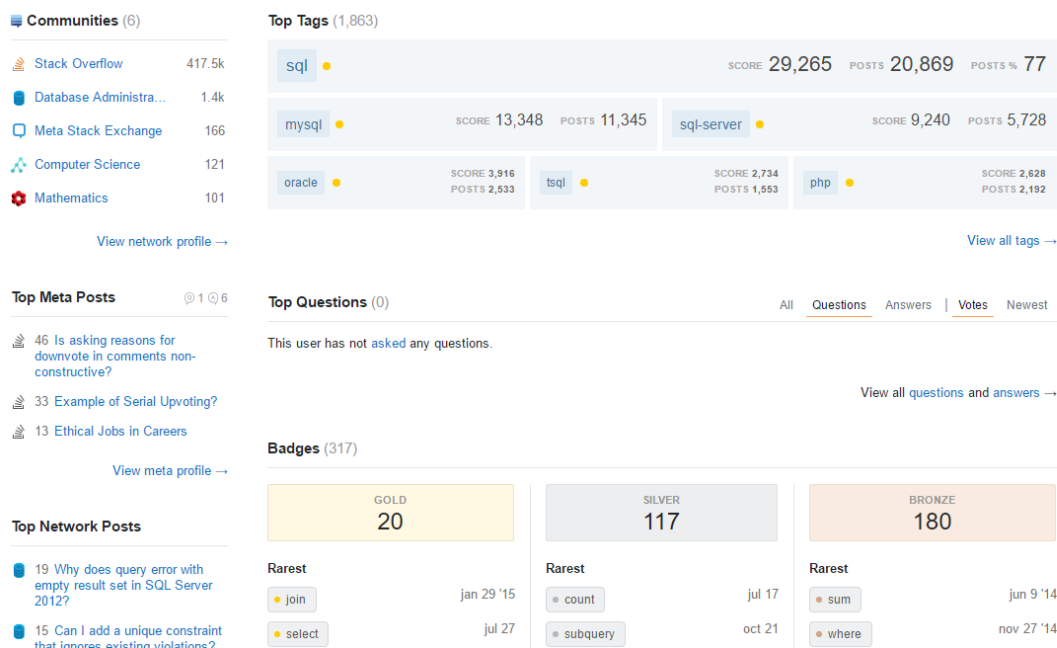
Εικόνα 4.8 Μέρος του προφίλ από το GitHub

Το επόμενο σύστημα είναι το Bitbucket το οποίο παρουσιάζει επίσης τα αποθετήρια του χρήστη, κάποιες γενικές πληροφορίες και την ημερομηνία εγγραφής του στο σύστημα. Μπορούμε να δούμε παραστατικά μέρος του προφίλ στην εικόνα 4.9, στην οποία φαίνονται οι ακόλουθοι του χρήστη αλλά και χρήστες τους οποίους ακολουθεί και στο δεξί άκρο τις εικόνες οι πρόσφατες δραστηριότητες του χρήστη.



Εικόνα 4.9 Μέρος του προφίλ από το Bitbucket

Συνεχίζοντας το Stack Overflow παρουσιάζει τη φήμη, όλες τις απαντήσεις, ερωτήσεις και εμβλήματα του χρήστη και ακόμα κάποιες γενικές πληροφορίες όπως ο συνολικός αριθμός απαντήσεων, ερωτήσεων και για πόσο χρονικό διάστημα είναι εγγεγραμμένος στο σύστημα. Επίσης εμφανίζονται οι κορυφαίες ετικέτες με τις οποίες έχει εμπλακεί ο χρήστης που κατά πάσα πιθανότητα είναι και αυτές στις οποίες έχει κερδίσει χρυσό έμβλημα. Μέρος του προφίλ φαίνεται στην εικόνα 4.10.

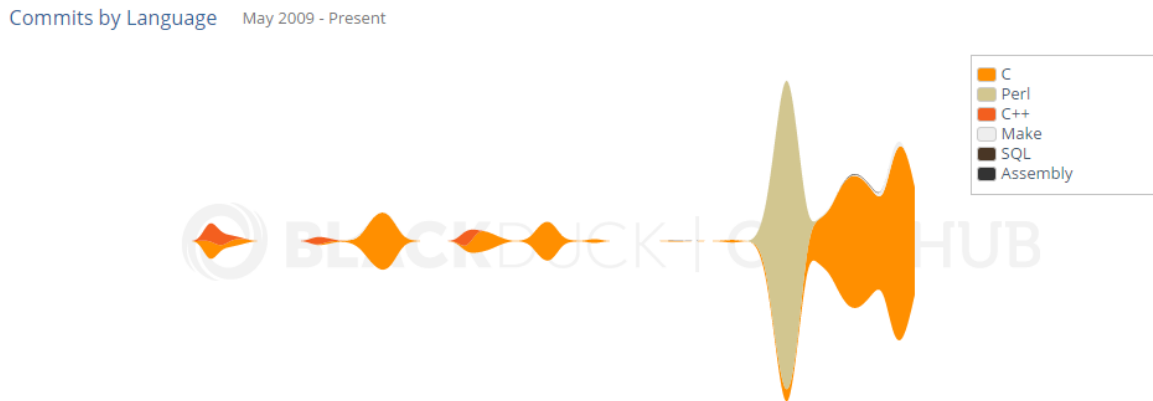


Εικόνα 4.10 Μέρος του προφίλ από το Stack Overflow

Όσο αφορά το Twitter, εμφανίζονται πιο λίγες πληροφορίες σε σχέση με τα προηγούμενα συστήματα που περιγράψαμε έως τώρα. Ανάμεσα σε αυτές είναι τα tweets του χρήστη και κάποιες γενικές πληροφορίες όπως η ημερομηνία εγγραφής του στο σύστημα και ίσως μια μικρή παράγραφος που περιγράφει τον ίδιο.

Ένα άλλο κοινωνικό δίκτυο για μηχανικούς λογισμικού το από το οποίο δεν συλλέγουμε οποιαδήποτε δεδομένα είναι το Open HUB. Όπως στα προηγούμενα και σε αυτό το σύστημα ο χρήστης έχει το προφίλ του, στο οποίο παρουσιάζονται κάποιες πληροφορίες. Ανάμεσα σε αυτές είναι τα αποθετήρια του χρήστη, τα εμβλήματα που έχει αποκτήσει, τα commits που έχει κάνει ανά γλώσσα και ανά έργο. Εκτός από αυτά εμφανίζονται και μερικά στατιστικά, όπως είναι η γλώσσα στην οποία διαθέτει τη περισσότερη εμπειρία, το πρώτο commit και το πιο πρόσφατο commit, τη διάρκεια εγγραφής στο σύστημα, σε πόσα έργα έχει συνεισφέρει και μια βαθμολογία που αφορά τη δημοτικότητα την οποία το σύστημα ονομάζει Kudo. Στην εικόνα

4.11 φαίνεται ο τρόπος παρουσίασης των commits ανά γλώσσα όπως εμφανίζονται στο Open HUB.



Εικόνα 4.11 Μέρος από το προφίλ του Open HUB

Ωστόσο πέραν από τα δεδομένα που συγκεντρώνει κάθε δίκτυο ξεχωριστά και εμφανίζει στο χρήστη, θα ήταν χρήσιμο να μπορεί να βλέπει και πληροφορίες για το προφίλ του οι οποίες προσφέρονται από περισσότερες από μια πηγές. Για αυτό το λόγο η ιστοσελίδα που υλοποιήθηκε στα πλαίσια της παρούσας διπλωματικής αποτελεί μια προσεγγιστική προσπάθεια συνδυασμού των στοιχείων από τους τέσσερις λογαριασμούς που προαναφέρθηκαν. Όστε να δίνει την ευκαιρία στο χρήστη να βλέπει τις πληροφορίες που αφορούν το προφίλ του, όχι διάσπαρτες στα διάφορα συστήματα αλλά συγκεντρωμένες σε ένα σύστημα. Επίσης θα είναι δυνατόν να βλέπει στατιστικά και βαθμολογίες, που αφορούν συγκεκριμένα πεδία, συνδυάζοντας περισσότερες από μια πηγές, όπως και κάποιες βαθμολογίες συγκριτικές με το δικό μας σύστημα. Ανάμεσα στις πληροφορίες αυτές υπάρχουν αρκετά ενδιαφέροντα στατιστικά τα οποία δεν παρουσιάζονται σε κανένα από τα τέσσερα συστήματα. Χρησιμοποιούνται όσο το δυνατό περισσότερο παραστατικές μέθοδοι παρουσίασης των στοιχείων, κάνοντας πιο ενδιαφέρον την εξερεύνηση του προφίλ και ανακαλύπτοντας πτυχές του προφίλ τις οποίες δεν είναι σε θέση να δει στα άλλα συστήματα. Επομένως ένας μηχανικός λογισμικού που θα επισκεφτεί το δικό μας σύστημα, θα έχει πιο ολοκληρωμένη εικόνα για το προφίλ του λόγω αυτής της ποικιλομορφίας συστημάτων που χρησιμοποιούμε σε αντίθεση με τα άλλα συστήματα.

Κεφάλαιο 5

Υλοποίηση και Λειτουργία της Ιστοσελίδας

5.1 Περιορισμοί	36
5.2 Περιπτώσεις Χρήσης	37
5.3 Αρχιτεκτονική	41
5.4 Βάση Δεδομένων	42
5.5 Διαπροσωπείες Συστήματος(Interfaces)	45
5.6 Λειτουργία Ιστοσελίδας	49
5.7 Εργαλεία που χρησιμοποιήθηκαν	51

5.1 Περιορισμοί

Όπως σε όλα τα συστήματα που αναπτύσσονται, έτσι και σε αυτό υπάρχουν κάποιοι περιορισμοί. Οι κύριοι περιορισμοί του συστήματος αυτού είναι ο αριθμός των αιτημάτων που παρέχονται από τα APIs που χρησιμοποιούμε. Στον Πίνακα 5.1 φαίνονται τα όρια όσο αφορά τα αιτήματα για κάθε API.

ΟΡΙΑ ΑΙΤΗΜΑΤΩΝ		
API	Με Ταυτοποίηση	Χωρίς Ταυτοποίηση
GitHub	5000/ώρα	60/ώρα
Bitbucket	60000/ώρα	60000/ώρα
Stack Overflow	10000/ημέρα	10000/ημέρα για κάθε IP
Twitter	300/15 λεπτά	0

Πίνακας 5.1 Όρια Αιτημάτων

Ο αριθμός των αιτημάτων που βλέπουμε στον Πίνακα 5.1 είναι για τις μεθόδους τις οποίες καλούμε μέσω των APIs. Άλλες μέθοδοι των APIs ίσως διαφέρουν στα όρια αιτημάτων.

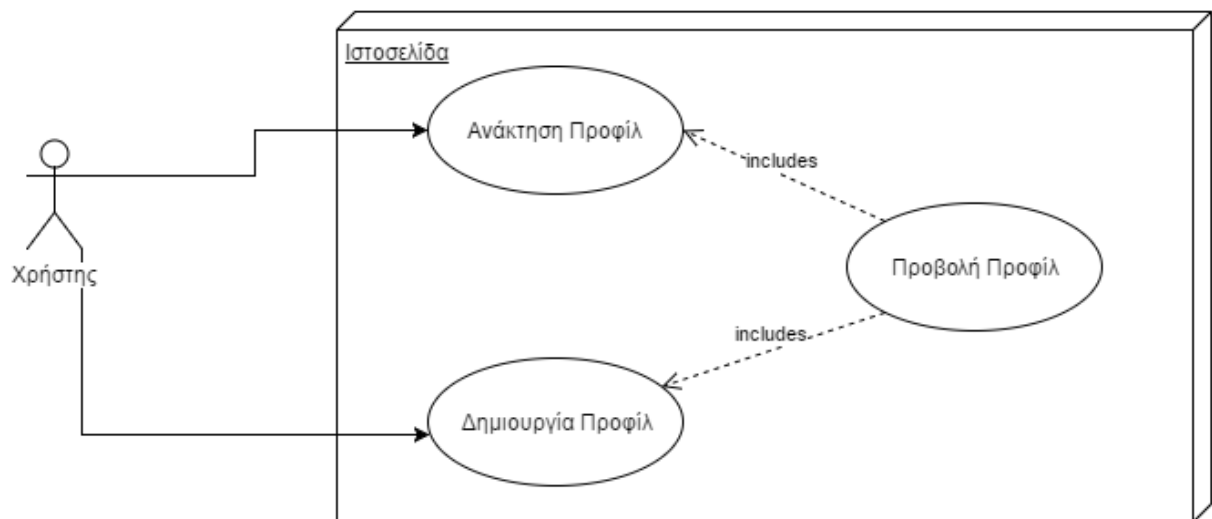
Ένας άλλος περιορισμός είναι η γλώσσα της ιστοσελίδας, η οποία είναι η Αγγλική, λόγω του ότι είναι και η διεθνές γλώσσα και μπορεί να αναγνωρίζεται από όλους τους χρήστες.

Επίσης το κάθε μοναδικό αναγνωριστικό (όνομα χρήστη ή id) ενός χρήστη σε ένα από τους λογαριασμούς GitHub, Bitbucket, Stack Overflow και Twitter μπορεί να αντιστοιχιστεί μόνο με ένα μοναδικό αναγνωριστικό για κάθε ένα από τους συγκεκριμένους λογαριασμούς. Πιο συγκεκριμένα, δεν θα είναι δυνατό για παράδειγμα να αντιστοιχούν δυο λογαριασμοί του GitHub σε ένα λογαριασμό του Twitter.

5.2 Περιπτώσεις Χρήσης

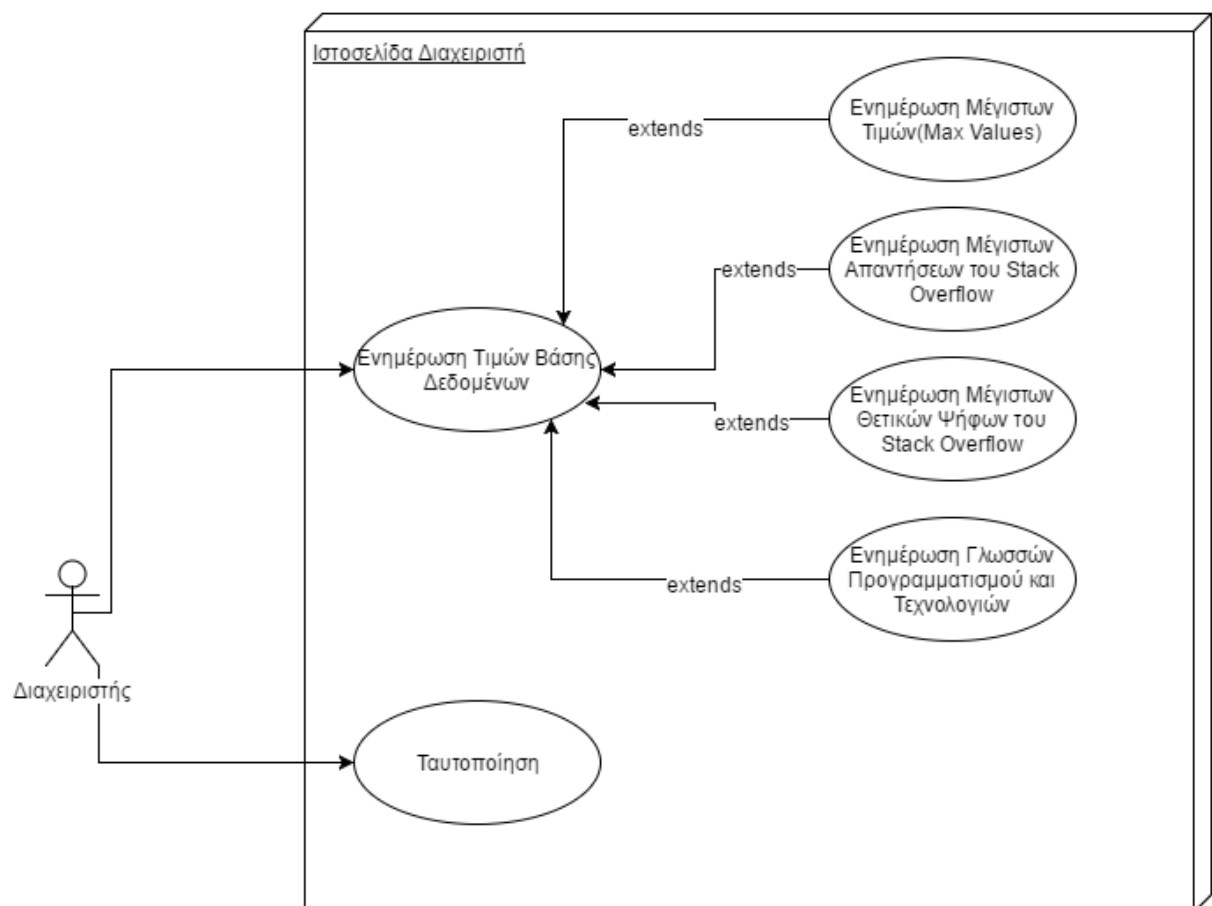
5.2.1 Χρήστες

Οι χρήστες του συστήματος χωρίζονται σε δυο κατηγορίες, τους απλούς χρήστες και το διαχειριστή. Πιο κάτω στο Σχήμα 5.1 μπορούμε να δούμε τις περιπτώσεις χρήσης του απλού χρήστη.



Σχήμα 5.2 Διάγραμμα Χρήστη

Στη συνέχεια βλέπουμε το Σχήμα 5.3 στο οποίο αναπαρίστανται οι περιπτώσεις χρήσης του διαχειριστή.



Σχήμα 5.3 Διάγραμμα Διαχειριστή

5.2.2 Περιγραφή Χρηστών

Στη συνέχεια θα περιγράψουμε τους χρήστες ως προς την αλληλεπίδρασή τους με το σύστημα. Η περιγραφή των χρηστών φαίνεται ξεκάθαρα στον Πίνακα 5.4

Χρήστης	Περιγραφή
Απλός Χρήστης	Ο οποιοσδήποτε χρήστης εισέλθει στο σύστημα θα μπορεί να κάνει μια βασική λειτουργία η οποία είναι να δει το προφίλ του. Για να το επιτύχει αυτό θα πρέπει να δώσει τα απαραίτητα στοιχεία και ακολούθως είτε να δημιουργήσει το προφίλ είτε να το ανακτήσει.
Διαχειριστής	Ο διαχειριστής του συστήματος αφού κάνει ταυτοποίηση στο σύστημα, θα έχει τη δυνατότητα να ενημερώσει κάποιες τιμές οι οποίες βρίσκονται στη Βάση Δεδομένων και χρησιμοποιούνται από το σύστημα. Οι ενημερώσεις που μπορεί να κάνει είναι αυτές που αφορούν τις μέγιστες τιμές, αυτές που αφορούν τις μέγιστες απαντήσεις και τις μέγιστες

	θετικές ψήφους του Stack Overflow και αυτές που αφορούν τις γλώσσες και τεχνολογίες που χρησιμοποιεί το σύστημα.
--	--

Πίνακας 5.4 Περιγραφή Χρηστών

5.2.3 Περιγραφή Περιπτώσεων Χρήσης

- **Προβολή Προφίλ**

Περιγραφή: Ο χρήστης αφού επισκεφθεί την ιστοσελίδα θα πρέπει να εισάγει τα στοιχεία(GitHub username, Bitbucket username, Stack Overflow ID, Twitter username) του ούτως ώστε να μπορεί να δει το προφίλ του. Αφού συμπληρώσει τα στοιχεία που απαιτούνται τότε έχει δυο επιλογές. Η μια είναι να δημιουργήσει το προφίλ του από την αρχή και η άλλη είναι να ανακτήσει το προφίλ του. Για να ανακτήσει το προφίλ του θα πρέπει να το έχει δημιουργήσει έστω μια φορά στο παρελθόν.

Χρήστες: Απλός Χρήστης

- **Ταυτοποίηση**

Περιγραφή: Ο χρήστης αφού εισέλθει στο σύστημα θα χρειαστεί να κάνει ταυτοποίηση χρησιμοποιώντας το ψευδώνυμο και τον κωδικό του ώστε να μπορεί να προχωρήσει με την ενημέρωση των διαφόρων τιμών στους πίνακες που θα έχει μπροστά του.

Χρήστες: Διαχειριστής

- **Ενημέρωση Μέγιστων Τιμών**

Περιγραφή: Ο χρήστης αφού κάνει σύνδεση στο σύστημα χρησιμοποιώντας το ψευδώνυμο και τον κωδικό του θα μπορεί να ενημερώνει τις μέγιστες τιμές στον πίνακα που θα έχει μπροστά του.

Χρήστες: Διαχειριστής

- **Ενημέρωση Μέγιστων Απαντήσεων Stack Overflow**

Περιγραφή: Ο χρήστης αφού κάνει σύνδεση στο σύστημα χρησιμοποιώντας το ψευδώνυμο και τον κωδικό του θα μπορεί να ενημερώνει τις μέγιστες απαντήσεις που αφορούν το Stack Overflow, στον πίνακα που θα του εμφανιστεί.

Χρήστες: Διαχειριστής

- **Ενημέρωση Μέγιστων Θετικών Ψήφων Stack Overflow**

Περιγραφή: Ο χρήστης αφού κάνει σύνδεση στο σύστημα χρησιμοποιώντας το ψευδώνυμο και τον κωδικό του θα μπορεί να ενημερώνει τις μέγιστες θετικές ψήφους των απαντήσεων που αφορούν το Stack Overflow, στον πίνακα που θα του εμφανιστεί.

Χρήστες: Διαχειριστής

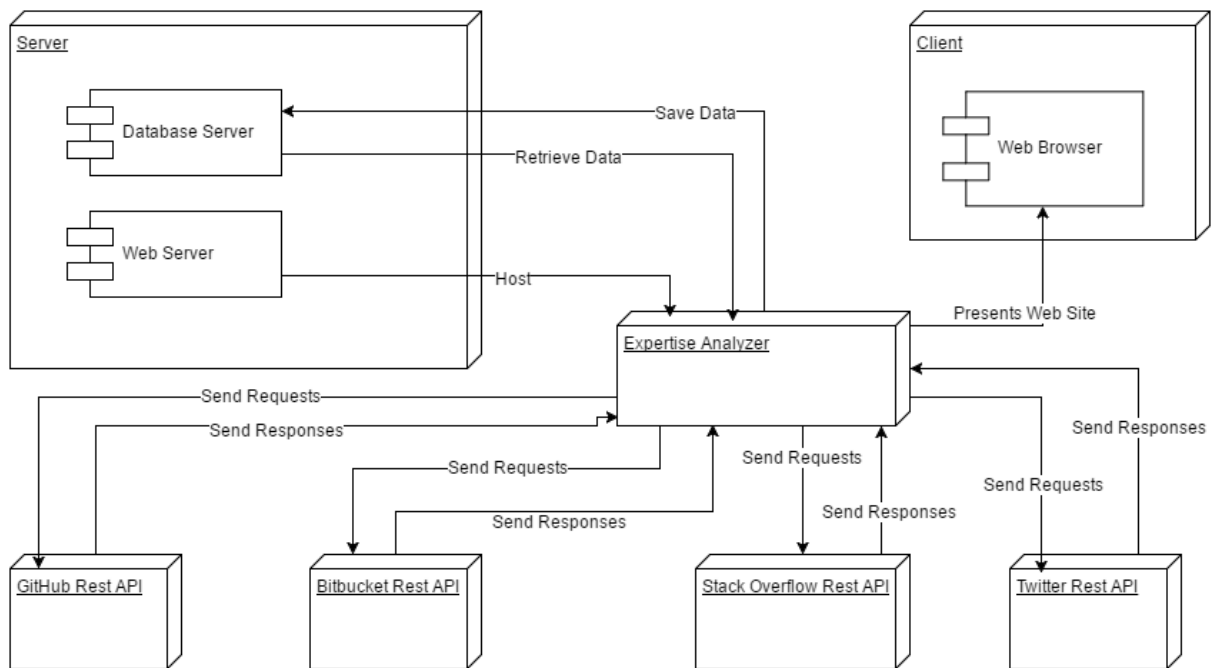
- **Ενημέρωση Γλωσσών Προγραμματισμού και Τεχνολογιών**

Περιγραφή: Ο χρήστης αφού κάνει σύνδεση στο σύστημα χρησιμοποιώντας το ψευδώνυμο και τον κωδικό του θα μπορεί να ενημερώνει στο σχετικό πίνακα τις γλώσσες προγραμματισμού και τεχνολογίες τις οποίες χρησιμοποιεί το σύστημα.

Χρήστες: Διαχειριστής

5.3 Αρχιτεκτονική

Το σύστημα σχεδιάστηκε με βάση την αρχιτεκτονική πελάτη-εξυπηρετητή(client-server). Δηλαδή ο πελάτης χρησιμοποιώντας την ιστοσελίδα στέλνει αιτήματα στον εξυπηρετητή και ο εξυπηρετητής ανταποκρίνεται πίσω στον πελάτη εμφανίζοντας του το κατάλληλο αποτέλεσμα. Στη περίπτωση του συστήματός μας υπάρχει ο εξυπηρετητής ο οποίος αποτελείται από τη βάση δεδομένων(Database) και τον διαδικτυακό εξυπηρετητή(Web Server), η ιστοσελίδα, τα διάφορα APIs και ο πελάτης. Η αρχιτεκτονική του συστήματος φαίνεται παραστατικά στο Σχήμα 5.5.

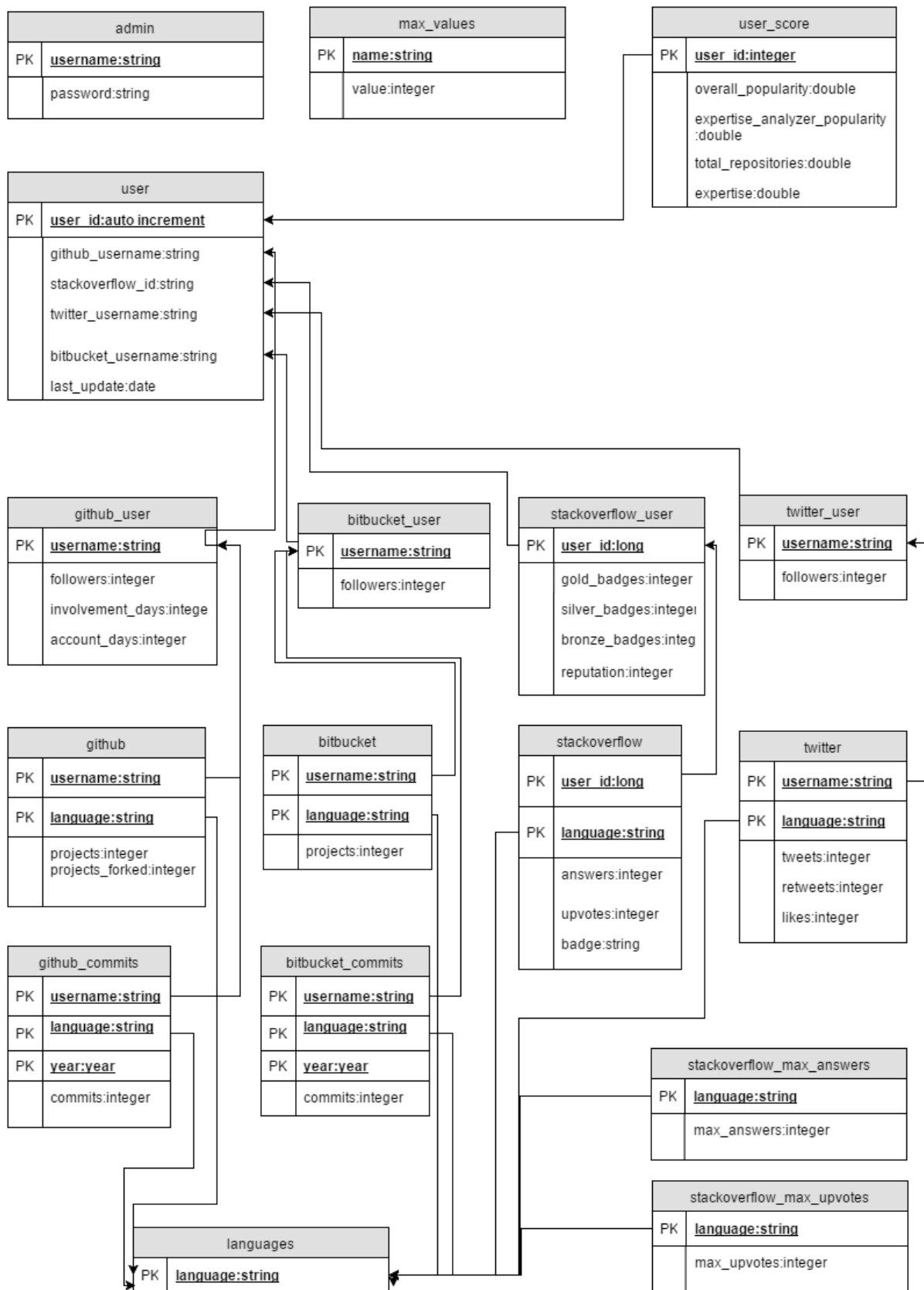


Σχήμα 5.5 Αρχιτεκτονική Συστήματος

Η ιστοσελίδα φιλοξενείται στο διαδικτυακό εξυπηρετητή. Ο χρήστης ή πελάτης, ανοίγοντας τον φυλλομετρητή στη τοπική του μηχανή μπορεί να έχει πρόσβαση στην ιστοσελίδα μας. Ακολούθως αφού δώσει τα απαραίτητα στοιχεία στέλνει αίτημα στην ιστοσελίδα και λαμβάνει τα αποτελέσματα. Η ιστοσελίδα επικοινωνεί με τα διάφορα APIs που χρησιμοποιούμε και επιπρόσθετα αποθηκεύει ή ανακτά δεδομένα από τη βάση δεδομένων και ανταποκρίνεται κατάλληλα στον πελάτη.

5.4 Βάση Δεδομένων

Σε αυτό το σημείο πρόκειται να περιγράψουμε τη βάση δεδομένων. Όπως βλέπουμε και στο Σχήμα 5.3 η βάση δεδομένων περιέχει τους πίνακες admin, max_values, user_score, user, github_user, bitbucket_user, stackoverflow_user, twitter_user, github, bitbucket, stackoverflow, twitter, github_commits, bitbucket_commits, languages stackoverflow_max_answers και stackoverflow_max_upvotes.



Σχήμα 5.6 Σχεσιακό Μοντέλο

Στη συνέχεια θα περιγράψουμε τον κάθε πίνακα και τον σκοπό που δημιουργήθηκε. Όπως μπορούμε να δούμε στον πίνακα 5.6 παρέχεται η περιγραφή του κάθε πίνακα ξεχωριστά.

Όνομα Πίνακα	Περιγραφή
admin	Περιέχει τα στοιχεία τα οποία χρειάζεται ο διαχειριστής για να εισέλθει στο σύστημα και να ενημερώσει κάποιες τιμές στη Βάση Δεδομένων.
max_values	Είναι οι μέγιστες τιμές για κάποιες παραμέτρους οι οποίες χρησιμοποιούνται για τον υπολογισμό κάποιων σκορ. Για παράδειγμα η τιμή για τον χρήστη που έχει τα πιο πολλά αποθετήρια στο GitHub.
user_score	Αποθηκεύονται τα σκορ που υπολογίζονται για τους χρήστες
user	Είναι ένα πίνακας στον οποίο ταυτίζονται τα μοναδικά διακριτικά που διατηρούν οι χρήστες στα τέσσερα συστήματα με επιπλέον ένα μοναδικό κλειδί που αυξάνεται αυτόματα.
github_user	Περιέχονται στοιχεία του GitHub σχετικά με τον χρήστη.
bitbucket_user	Περιέχονται στοιχεία του Bitbucket σχετικά με τον χρήστη.
stackoverflow_user	Περιέχονται στοιχεία του Stack Overflow σχετικά με τον χρήστη.
twitter_user	Περιέχονται στοιχεία του Twitter σχετικά με τον χρήστη.
github	Περιέχονται δεδομένα σχετικά με τα αποθετήρια του χρήστη και τη γλώσσα προγραμματισμού στην οποία ανήκουν.
bitbucket	Περιέχονται δεδομένα σχετικά με τα αποθετήρια του χρήστη και τη γλώσσα προγραμματισμού στην οποία ανήκουν.
stackoverflow	Περιέχονται δεδομένα σχετικά με τις απαντήσεις που έχει δώσει ο χρήστης, τις θετικές ψήφους που έχει πάρει και τα εμβλήματα που έχει κερδίσει, για κάθε γλώσσα προγραμματισμού ή τεχνολογία.

twitter	Περιέχονται όλα τα tweets, retweets και likes, σχετικά με το προφίλ του χρήστη στο Twitter, για κάθε γλώσσα προγραμματισμού ή τεχνολογία.
github_commits	Είναι όλα τα commits αρχείων κώδικα και μη η οποίες έχουν πραγματοποιηθεί στο Github και ανήκουν σε μια συγκεκριμένη γλώσσα προγραμματισμού
bitbucket_commits	Είναι όλα τα commits αρχείων κώδικα και μη η οποίες έχουν πραγματοποιηθεί στο Bitbucket και ανήκουν σε μια συγκεκριμένη γλώσσα προγραμματισμού
stackoverflow_max_answers	Είναι ο αριθμός των απαντήσεων του χρήστη που έχει δώσει τις περισσότερες απαντήσεις ,σε κάθε γλώσσα προγραμματισμού ή τεχνολογία.
stackoverflow_max_upvotes	Είναι ο αριθμός των θετικών ψήφων του χρήστη που έχει πάρει τις περισσότερες θετικές ψήφους στις απαντήσεις του, σε κάθε γλώσσα προγραμματισμού ή τεχνολογία.
languages	Είναι όλες οι γλώσσες και τεχνολογίες οι οποίες βρίσκονται ανάμεσα στις κορυφαίες εκατό ετικέτες του Stack Overflow

Πίνακας 5.7 Περιγραφή Πινάκων

5.5 Διαπροσωπείες Συστήματος(Interfaces)

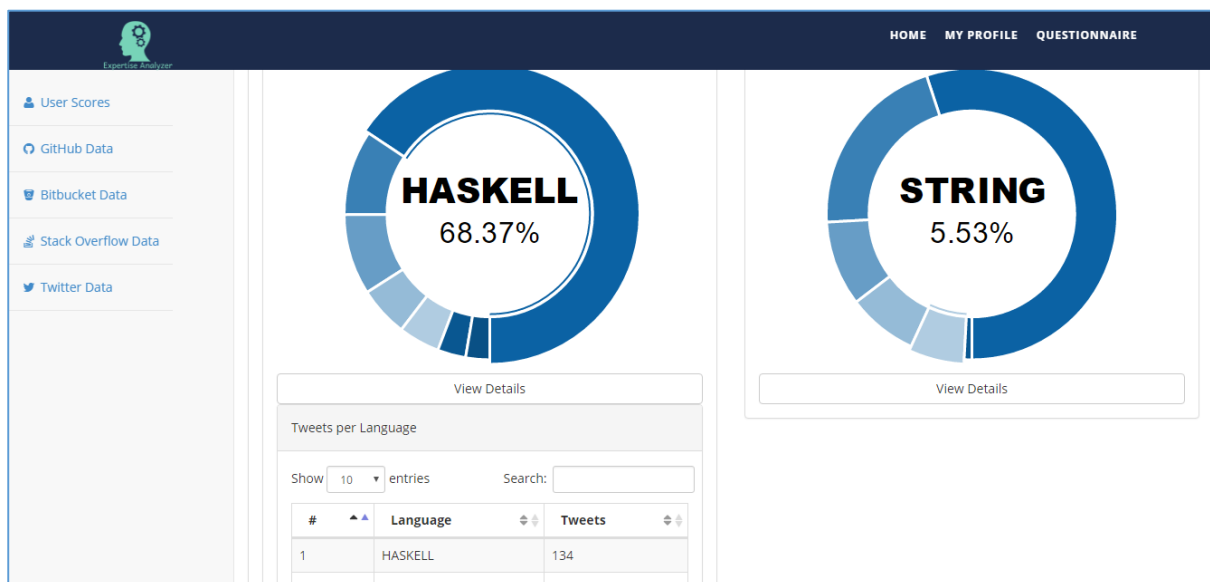
5.5.1 Διαπροσωπεία Χρήστη

Η διαπροσωπεία του χρήστη δημιουργήθηκε ώστε ο χρήστης να μπορεί να αλληλεπιδρά με το σύστημα. Χωρίζεται σε δυο βασικές σελίδες. Η πρώτη είναι η αρχική η σελίδα η οποία εμφανίζεται στον χρήστη μόλις επισκεφθεί την σελίδα. Περιλαμβάνει τα πιο κάτω:

- Γενικές πληροφορίες για τη σελίδα
- Πεδία τα οποία πρέπει να συμπληρώσει ο χρήστης
- Πληροφορίες για τα πεδία τα οποία χρειάζεται να συμπληρώσει ο χρήστης
- Επεξήγηση και κάποιες διευκρινίσεις όσο αφορά τη δημιουργία προφίλ του χρήστη
- Στοιχεία επικοινωνίας

Στην εικόνα 5.8 και 5.9 εμφανίζονται κάποια μέρη από τη διαπροσωπεία του χρήστη.

Εικόνα 5.8 Διαπροσωπεία Χρήστη Α



Εικόνα 5.9 Διαπροσωπεία Χρήστη Β

Η δεύτερη σελίδα είναι το προφίλ του χρήστη το οποίο μπορεί να δει αφού συμπληρώσει τα στοιχεία του στην αρχική σελίδα. Έχει υλοποιηθεί με τρόπο ώστε να τα δεδομένα που παρουσιάζονται να είναι εύκολα κατανοητά και ο χρήστης να κατευθύνεται χωρίς δυσκολία στο σημείο που θέλει.

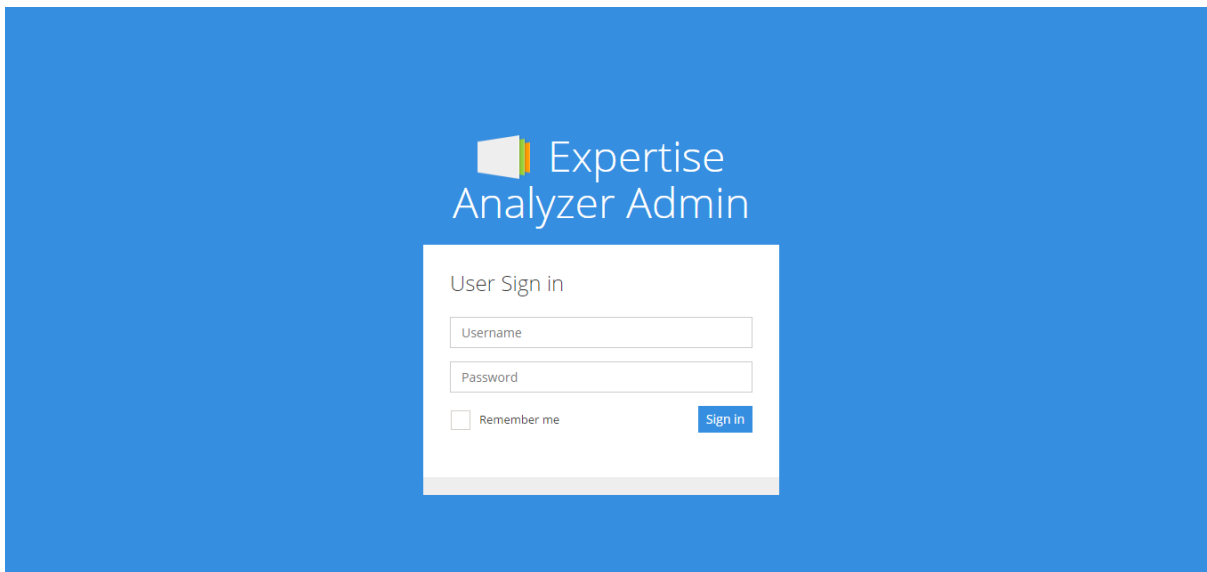
Η διαπροσωπεία του προφίλ περιλαμβάνει σχετικά με το χρήστη τα ακόλουθα:

- Βαθμολογίες
- Στατιστικά
- Λογαριασμούς
- Γραφικές παραστάσεις
- Πίνακες

5.5.2 Διαπροσωπεία Διαχειριστή

Εκτός από τη διαπροσωπεία χρήστη, έχει υλοποιηθεί και η διαπροσωπεία του διαχειριστή. Η διαπροσωπεία του διαχειριστή αποτελείται από τρία είδη διαφορετικών σελίδων. Το πρώτο είδος σελίδας είναι η σελίδα εισδοχής του χρήστη στο σύστημα η οποία περιλαμβάνει:

- Πεδίο εισαγωγής ονόματος χρήστη
- Πεδίο εισαγωγής κωδικού



Εικόνα 5.0.10 Διαπροσωπεία Διαχειριστή Α

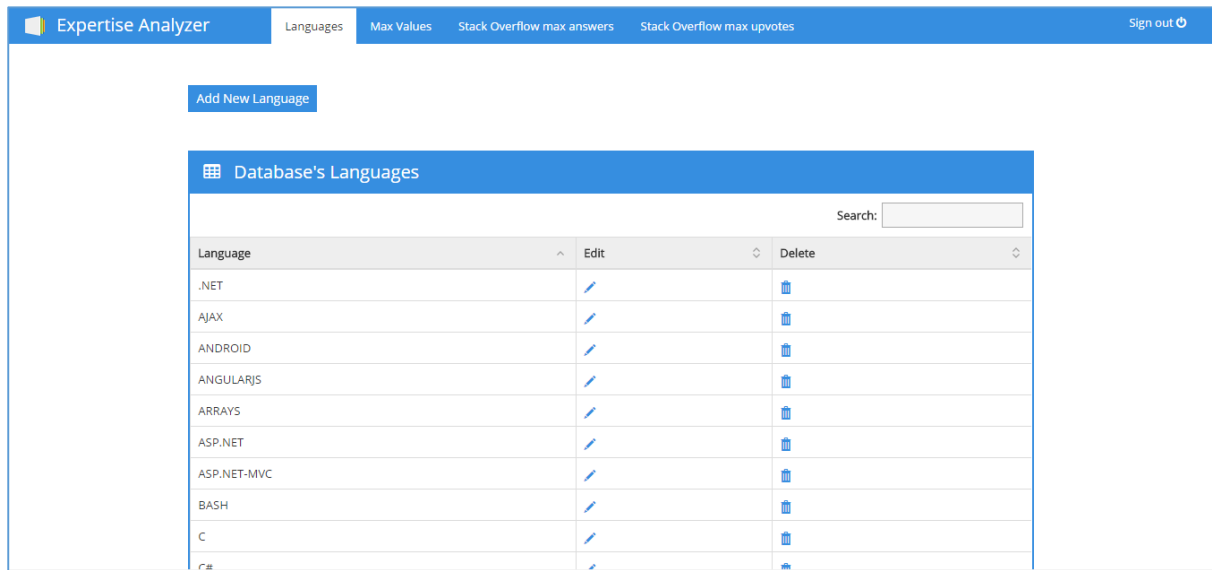
Στην εικόνα 5.10 μπορούμε να διακρίνουμε τη σελίδα εισδοχής του διαχειριστή.

Το δεύτερο είδος σελίδας είναι τέσσερις σελίδες όμοιες, με μόνη διαφορά τους πίνακες τους οποίους ο διαχειριστής μπορεί να ενημερώσει. Πιο συγκεκριμένα, είναι οι εξής πίνακες:

- Πίνακας μεγίστων τιμών
- Πίνακας με τις γλώσσες προγραμματισμού και τεχνολογίες
- Πίνακας με τις μέγιστες απαντήσεις του Stack Overflow

- Πίνακας με τις μέγιστες θετικές ψήφους του Stack Overflow

Μπορούμε να τις δούμε και παραστατικά στην εικόνα 5.11 και συγκεκριμένα τη σελίδα με τον πίνακα που περιέχει τις γλώσσες προγραμματισμού και τεχνολογίες.



Εικόνα 5.11 Διαπροσωπεία Διαχειριστή Β

Το τελευταίο είδος σελίδας είναι η σελίδα η οποία εμφανίζεται όταν ο χρήστης επιλέξει να κάνει μια προσθήκη ή τροποποίηση ενός συγκεκριμένου στοιχείου. Δηλαδή όταν πατήσει στο κουμπί Add ή στο μολύβι στην στήλη Edit όπως φαίνονται στην εικόνα 5.11. Στην σελίδα αυτή περιέχονται όλα τα πεδία που πρέπει να συμπληρώσει για την προσθήκη ή σε περίπτωση τροποποίησης, εμφανίζονται όλες οι πληροφορίες σχετικά με το στοιχείο επιλογής του χρήστη και δίπλα τα πεδία με τις υπάρχουσες τιμές, τις οποίες μπορεί να τροποποιήσει. Η σελίδα προσθήκης και επεξεργασίας είναι ορατή στη εικόνα 5.12.

The image shows a web application interface for 'Expertise Analyzer'. At the top, there is a blue navigation bar with the application name and several menu items: 'Languages', 'Max Values', 'Stack Overflow max answers', 'Stack Overflow max upvotes', and a 'Sign out' link with a user icon. The main content area is white and contains a modal form titled 'Edit Max Answers'. This form has two input fields: 'Language Name' with the value '.NET' and 'Max Value' with the value '5233'. Below these fields are two buttons: 'Submit' (in blue) and 'Cancel' (in grey).

Εικόνα 5.12 Επεξεργασία Μέγιστης Τιμής Απαντήσεων

5.6 Λειτουργία Ιστοσελίδας

5.6.1 Λειτουργία Ιστοσελίδας Χρήστη

Από το σημείο αυτό θα συνεχίσουμε με τη περιγραφή του τρόπου με τον οποίο λειτουργεί η ιστοσελίδα που χρησιμοποιεί ο χρήστης. Αρχικά ο χρήστης επισκέπτεται την ιστοσελίδα και αυτόματα βρίσκεται στο αρχείο `index.html` της ιστοσελίδας όπου βλέπει την αρχική σελίδα του συστήματος. Ακολούθως αφού συμπληρώσει τους λογαριασμούς του έχει δυο επιλογές, είτε να δημιουργήσει το προφίλ του είτε να το ανακτήσει. Αφού επιλέξει, η ροή μεταφέρεται σε ένα άλλο αρχείο, το `create-profile.php` μαζί με τους λογαριασμούς που συμπλήρωσε ο χρήστης.

Να αναφέρουμε ότι το αρχείο `create-profile.php` είναι ο κορμός του συστήματος από τον οποίο καλούνται όλες οι λειτουργίες που χρειάζεται να εκτελεστούν ώστε να δημιουργηθεί ή να ανακτηθεί το προφίλ του χρήστη. Επομένως γίνεται έλεγχος για να εξακριβωθεί ποιους λογαριασμούς συμπλήρωσε ο χρήστης ώστε στη συνέχεια να καλεστούν τα σχετικά αρχεία τα οποία είναι υπεύθυνα για την εξόρυξη δεδομένων από τα APIs που χρησιμοποιούμε. Για τον έλεγχο των λογαριασμών του χρήστη αρχικά ελέγχεται ποια πεδία έχει συμπληρώσει ο χρήστης.

Στη συνέχεια αποστέλλονται αιτήματα στα APIs για ανάκτηση του λογαριασμού του χρήστη, όμως μόνο για τους λογαριασμούς που έχει συμπληρώσει. Λαμβάνοντας πίσω τις απαντήσεις

από τα API γίνεται έλεγχος για να εξακριβωθεί αν το προφίλ του είναι έγκυρο, δηλαδή αν υπάρχει στο σύστημα. Αφού εντοπιστούν οι έγκυροι λογαριασμοί τότε εάν ο χρήστης επέλεξε να δημιουργήσει το προφίλ του συνεχίζεται κανονικά η αποστολή των υπολοίπων αιτημάτων στα APIs και συνεχίζεται η εξόρυξη δεδομένων. Σε περίπτωση που ο χρήστης επέλεξε να ανακτήσει το προφίλ του καλούνται άλλα αρχεία τα οποία είναι υπεύθυνα ώστε να ανακτήσουν τα δεδομένα του χρήστη από τη βάση δεδομένων. Η διαδικασία εξόρυξης δεδομένων έχει περιγραφεί στο κεφάλαιο 3 αναλυτικότερα.

Αφού συλλεγούν τα δεδομένα δημιουργούνται για κάθε λογαριασμό τα στατιστικά, οι κατάλληλες γραφικές παραστάσεις και οι πίνακες που μελετήσαμε στο κεφάλαιο 4, τα οποία αποθηκεύονται στην καθολική μεταβλητή `$_SESSION`. Η μεταβλητή `$_SESSION` είναι ορατή για όλη την περίοδο επίσκεψης του χρήστη στο σύστημα. Να αναφέρουμε ότι οι γραφικές παραστάσεις δημιουργούνται στη γλώσσα Javascript με τη βοήθεια της βιβλιοθήκης Morris.js [18] χρησιμοποιώντας τα απαραίτητα δεδομένα για κάθε γραφική. Στη συνέχεια γίνεται ο υπολογισμός των βαθμολογιών του χρήστη οι οποίες επίσης αποθηκεύονται στη καθολική μεταβλητή `$_SESSION`. Εφόσον ο χρήστης δημιουργεί το προφίλ του για πρώτη φορά αποθηκεύονται όλα τα απαραίτητα δεδομένα στη βάση δεδομένων καλώντας τα αρχεία που είναι υπεύθυνα για τη συγκεκριμένη εργασία. Σε αντίθεση αν ο χρήστης έχει δημιουργήσει ξανά το προφίλ του στο παρελθόν, ενημερώνονται τα υπάρχοντα δεδομένα στη βάση.

Όταν θα έχουν ολοκληρωθεί όλες οι πιο πάνω διαδικασίες, η ροή εκτέλεσης μεταβαίνει στο αρχείο `profile.php` το οποίο χρησιμοποιεί τη καθολική μεταβλητή `$_SESSION` ώστε να πάρει τα δεδομένα που αποθηκεύτηκαν εκεί από το αρχείο `create-profile.php` και να τα παρουσιάσει στον χρήστη.

5.6.2 Λειτουργία Ιστοσελίδας Διαχειριστή

Στη συνέχεια θα περιγράψουμε τον τρόπο με τον οποίο λειτουργεί η ιστοσελίδα του διαχειριστή. Αρχικά ο διαχειριστής εισέρχεται στην αρχική σελίδα στην οποία πρέπει να συμπληρώσει τα στοιχεία του (όνομα χρήστη, κωδικό) ώστε να γίνει η ταυτοποίηση του. Η ταυτοποίηση γίνεται μέσα από τη βάση δεδομένων στην οποία είναι αποθηκευμένα όλα τα ονόματα χρηστών και οι αντίστοιχοι κωδικοί των διαχειριστών.

Αφού ολοκληρωθεί η ταυτοποίηση μεταβαίνει στην σελίδα όπου εμφανίζονται οι διάφοροι πίνακες τους οποίους μπορεί να ενημερώσει. Μπορεί να επιλέξει ο ίδιος ποιο πίνακα θα ενημερώσει από το navigation bar στο οποίο εμφανίζονται όλοι οι πίνακες ονομαστικά. Για να

προσθέσει δεδομένα ο χρήστης κάνει κλικ στο κουμπί Add και για να τροποποιήσει υπάρχων δεδομένα κάνει κλικ στο εικονίδιο με το μολύβι στην στήλη Edit του πίνακα και τη γραμμή της επιθυμίας του. Επομένως εμφανίζεται ένα νέο παράθυρο όπου μπορεί να συμπληρώσει τα στοιχεία του δεδομένου που θέλει να προσθέσει ή εμφανίζονται όλα τα στοιχεία του δεδομένου που επέλεξε να αλλάξει και δίπλα μπορεί να συμπληρώσει τα νέα στοιχεία. Παράλληλα εφόσον επιλέξει να προσθέσει ή να αλλάξει δεδομένα, τότε καλείται το αρχείο `add_edit_<table_name>.php`, όπου το `table_name` είναι το όνομα του πίνακα στον οποίο θα γίνουν οι αλλαγές. Επομένως το `add_edit` ελέγχει τι ενέργεια θέλει να κάνει ο χρήστης και καλεί το αρχείο `insert_update<table_name>.php` (για το `table_name` ισχύει ότι και για το προηγούμενο αρχείο) το οποίο συνδέεται με τη βάση δεδομένων και εκτελεί τις κατάλληλες ενέργειες προσθήκης ή ενημέρωσης δεδομένων.

Επίσης ο διαχειριστής όπως έχει την επιλογή προσθήκης και τροποποίησης δεδομένων, έχει και την επιλογή διαγραφής. Αυτό είναι εφικτό πατώντας το κουμπί που βρίσκεται στην στήλη Delete και τη γραμμή επιλογής του χρήστη ανάλογα με το στοιχείο που θέλει να διαγράψει. Για τη διαγραφή καλείται επίσης το αρχείο `insert_update<table_name>.php` μαζί με την ενέργεια του χρήστη, η οποία είναι η διαγραφή.

5.7 Εργαλεία που χρησιμοποιήθηκαν

Για την υλοποίηση και λειτουργία του συστήματος χρησιμοποιήθηκαν κάποια εργαλεία όπως γλώσσες προγραμματισμού και άλλα. Ανάμεσα σε αυτά είναι:

- Γλώσσα HTML
- Γλώσσα Javascript
- Τεχνολογία JQuery
- Γλώσσα CSS
- Γλώσσα PHP
- Βιβλιοθήκη Morris JS
- Πακέτο Bootstrap
- Εργαλείο Notepad++
- Βάση Δεδομένων MySQL
- Εργαλείο XAMPP

Κεφάλαιο 6

Αποτελέσματα Χρήσης

6.1 Ερωτηματολόγιο	52
6.2 Αποτελέσματα	53

6.1 Ερωτηματολόγιο

Μετά την υλοποίηση του συστήματος, χρειάστηκε να γίνουν κάποιες δοκιμές ώστε να υπάρχει η άποψη των χρηστών για το σύστημα. Έτσι δημιουργήσαμε ερωτηματολόγιο με το οποίο θέλαμε να γίνει αξιολόγηση του συστήματος πάνω σε κάποια συγκεκριμένα πεδία. Αυτά τα πεδία είναι η κατανόηση του συστήματος, η ευχρηστία, η χρησιμότητα, η επέκταση και η σύσταση. Για να γίνει πιο ξεκάθαρο, θα εξηγήσουμε εν συντομία τα πιο πάνω πεδία. Η κατανόηση του συστήματος αφορά ερωτήσεις βασισμένες στο πόσο ξεκάθαρο είναι το περιεχόμενο, δηλαδή τα στοιχεία που ζητά από τον χρήστη, τα στοιχεία που παρουσιάζει και ποιος είναι ο σκοπός του. Επίσης η ευχρηστία σχετίζεται με την ευκολία χρήσης του συστήματος από το χρήστη και πιο συγκεκριμένα κατά πόσο είναι εύκολο ο χρήστης να ολοκληρώσει μια ενέργεια από την αρχή μέχρι το τέλος με επιτυχία. Συνεχίζοντας η χρησιμότητα αφορά το πόσο χρήσιμο είναι το σύστημα ώστε ο χρήστης να μπορεί να εξάγει συμπεράσματα για τις ικανότητές του και να του δώσει κίνητρο να τις βελτιώσει. Τέλος η επέκταση έχει να κάνει με νέα δεδομένα που θα μπορούσαν να προστεθούν στο σύστημα και η σύσταση αφορά κατά πόσο ο χρήστης θα προτείνει το σύστημα σε άλλα άτομα.

Οι ερωτήσεις του ερωτηματολογίου είναι οι εξής:

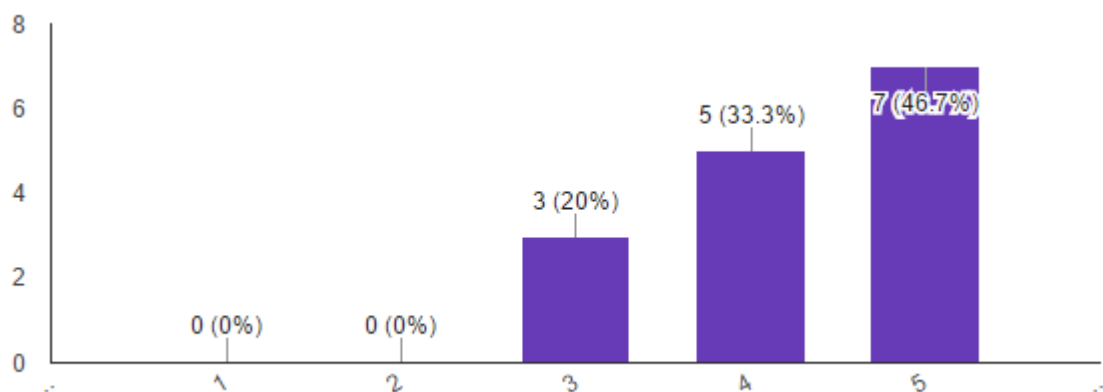
1. Οι πληροφορίες που ζητούνται από το σύστημα για δημιουργία του προφίλ είναι ξεκάθαρες;
2. Πόσο ικανοποιημένοι είστε με την ευκολία χρήσης της ιστοσελίδας;
3. Τα αποτελέσματα που εμφανίζονται στο προφίλ είναι κατανοητά;
4. Ποιο αποτέλεσμα του προφίλ, κατά τη γνώμη σας, δεν είναι κατανοητό;
5. Ποιο αποτέλεσμα του προφίλ, κατά τη γνώμη σας, δεν είναι χρήσιμο;

6. Υπάρχει κάποιο αποτέλεσμα που δεν σχετίζεται με την ιστοσελίδα;
7. Πόσο χρήσιμα είναι τα αποτελέσματα του προφίλ ώστε να σας δώσουν κίνητρο για βελτίωση των ικανοτήτων σας;
8. Τι άλλα αποτελέσματα θα θέλατε να δείτε στο προφίλ;
9. Γενικά βρίσκεται χρήσιμη την ιστοσελίδα;
10. Θα συστήνατε την ιστοσελίδα σε άλλους;
11. Θα θέλατε να συμπεριληφθούν στο προφίλ πληροφορίες και από άλλα κοινωνικά δίκτυα;

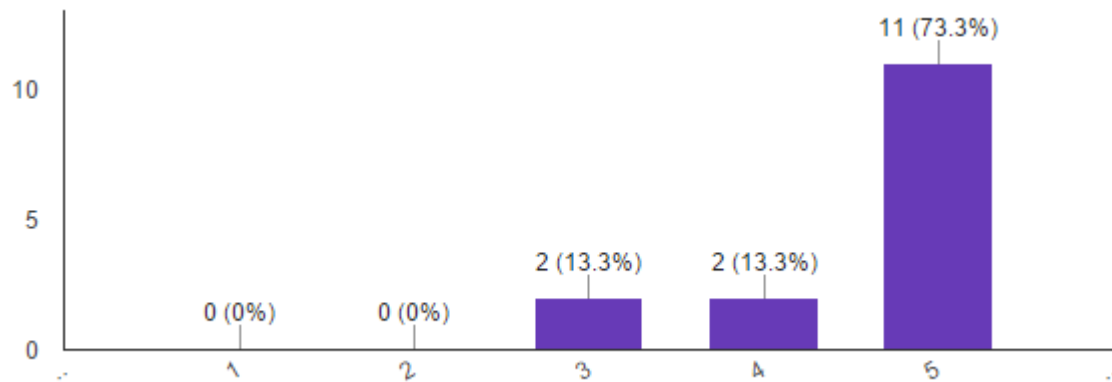
6.2 Αποτελέσματα

Τα ερωτηματολόγια δόθηκαν σε 15 άτομα σχετικά με την επιστήμη της πληροφορικής. Στις ερωτήσεις που οι χρήστες καλούνται να βαθμολογήσουν να αναφέρουμε ότι οι επιλογές είναι από το 1 μέχρι το 5, με το πρώτο να αντιστοιχεί στο «Διαφωνώ Απόλυτα» και το δεύτερο στο «Συμφωνώ Απόλυτα». Πιο κάτω μπορούμε να δούμε τα αποτελέσματα από τα ερωτηματολόγια.

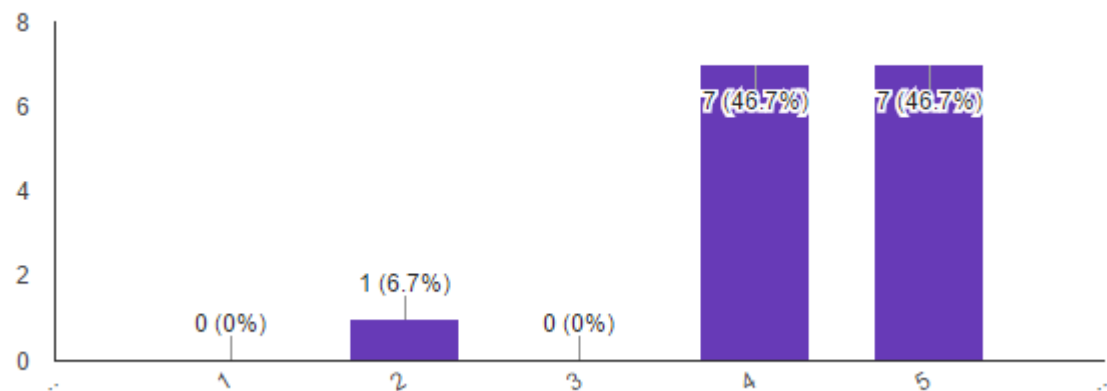
1. Οι πληροφορίες που ζητούνται από το σύστημα για δημιουργία του προφίλ είναι ξεκάθαρες;



2. Πόσο ικανοποιημένοι είστε με την ευκολία χρήσης της ιστοσελίδας;



3. Τα αποτελέσματα που εμφανίζονται στο προφίλ είναι κατανοητά;

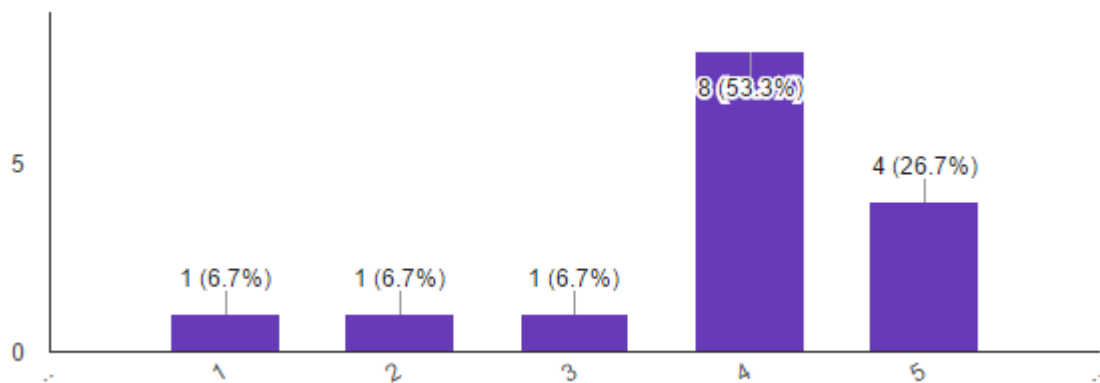


4. Ποιο αποτέλεσμα του προφίλ, κατά τη γνώμη σας, δεν είναι κατανοητό;

Εδώ τα αποτελέσματα τα οποία βαθμολογήθηκαν ως μη κατανοητά είναι:

- Overall Popularity: με 4 ψήφους από τους 15
- Social Activity: με 3 ψήφους από τους 15
- Expertise Analyzer Popularity: με 2 ψήφους από τους 15
- Expertise: με 2 ψήφους από τους 15
- Most experienced in: με 2 ψήφους από τους 15
- Total Languages' Percentage: με 1 ψήφο από τους 15
- Knowledge: με 1 ψήφο από τους 15
- Other: με 1 ψήφο από τους 15

5. Ποιο αποτέλεσμα του προφίλ, κατά τη γνώμη σας, δεν είναι χρήσιμο; Τα αποτελέσματα τα οποία βαθμολογήθηκαν ως μη χρήσιμα είναι τα πιο κάτω:
- Overall Popularity: με 1 ψήφο από τους 15
 - Total commits: με 1 ψήφο από τους 15
 - Total answers: με 1 ψήφο από τους 15
 - Total upvotes: με 1 ψήφο από τους 15
 - Joined Expertise Analyzer: με 1 ψήφο από τους 15
 - Social Activity: με 1 ψήφο από τους 15
 - Other: με 2 ψήφους από τους 15
6. Υπάρχει κάποιο αποτέλεσμα που δεν σχετίζεται με την ιστοσελίδα; Η μοναδική απάντηση που είχαμε στην ερώτηση αυτή, είναι για κάποιο χρήστη στον οποίο δεν εμφανίζονται τα commits του στο σύστημα αλλά στο GitHub ναί. Για αυτό να τονίσουμε ότι προηγουμένως έχουμε κάνει αναφορά στο ότι κάποια commits του GitHub δεν μπορούμε να τα συλλέξουμε διότι το GitHub δεν περιλαμβάνει πάντα την πληροφορία σχετικά με το ποιος είναι ο χρήστης ο οποίος δημιούργησε τα αρχεία που έγιναν commit.
7. Πόσο χρήσιμα είναι τα αποτελέσματα του προφίλ ώστε να σας δώσουν κίνητρο για βελτίωση των ικανοτήτων σας;

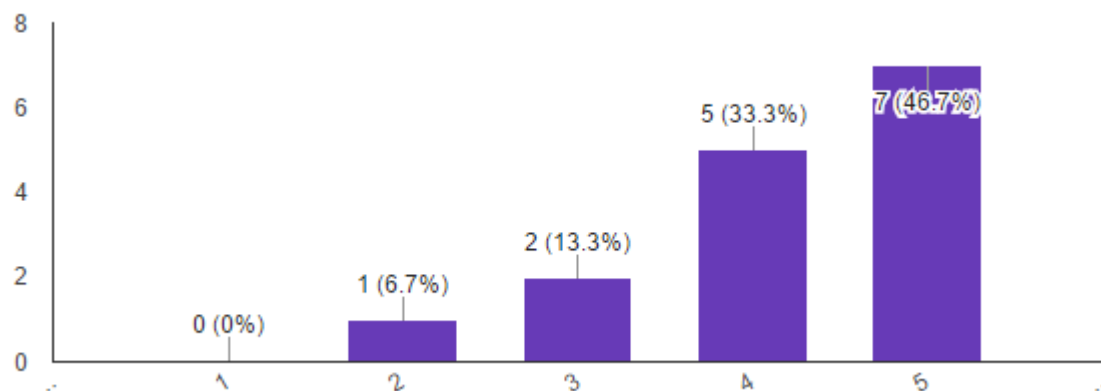


8. Τι άλλα αποτελέσματα θα θέλατε να δείτε στο προφίλ;

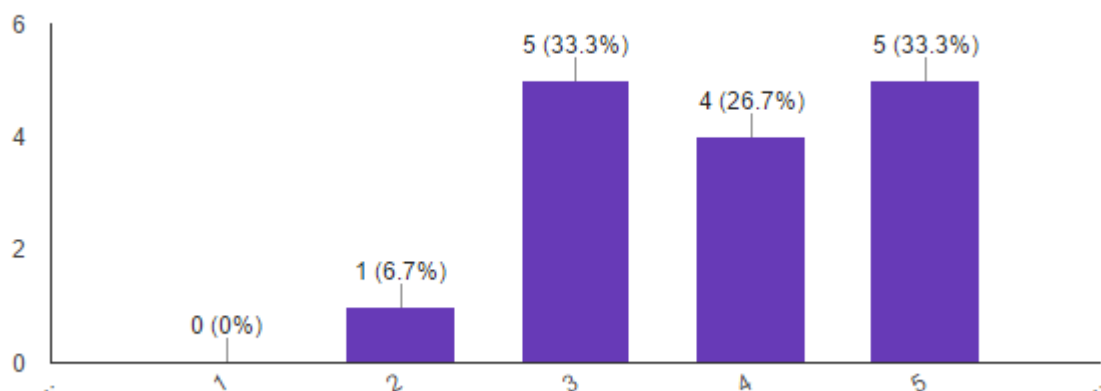
Σχετικά με αυτήν την ερώτηση είχαμε τις εξής απαντήσεις:

- Σύσταση χρηστών και σελίδων για να μπορούν να τους ακολουθήσουν οι χρήστες.
- Ανάλυση κώδικα των χρηστών και εντοπισμός κοινών λαθών που τυχόν να έχουν στον κώδικα τους.
- Να υπάρχει κάποιους είδους δίκτυο ώστε να μπορούν να συσχετίζονται οι χρήστες οι οποίες έχουν συνεργαστεί.
- Κάποια μέθοδος δημιουργίας του CV του χρήστη ή το μέρος που αφορά τεχνικά χαρακτηριστικά.
- Να γίνετε μια σύγκριση της προόδου του χρήστη σε σχέση με την προηγούμενη φορά που επισκέφθηκε την σελίδα.

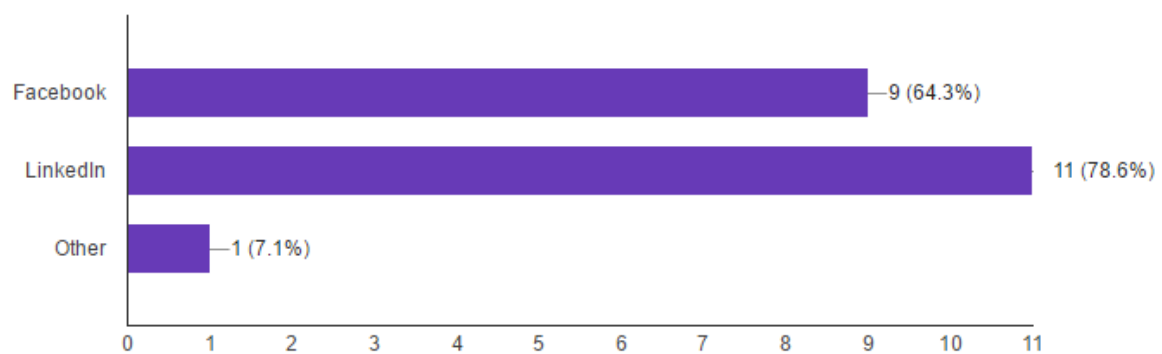
9. Γενικά βρίσκεται χρήσιμη την ιστοσελίδα;



10. Θα συστήνατε την ιστοσελίδα σε άλλους;



11. Θα θέλατε να συμπεριληφθούν στο προφίλ πληροφορίες και από άλλα κοινωνικά δίκτυα;



Κεφάλαιο 7

Συμπεράσματα

7.1 Συμπεράσματα	58
7.1 Μελλοντική Εργασία	60

7.1 Συμπεράσματα

Ανακεφαλαιώνοντας μελετήσαμε και εξηγήσαμε το σύστημα Expertise Analyzer που δημιουργήθηκε για τους σκοπούς της παρούσας διπλωματικής εργασίας. Αποτελεί ένα σύστημα το οποίο προσφέρεται για δημόσια χρήση από τους μηχανικούς λογισμικού, το οποίο πραγματοποιεί συλλογή δεδομένων από τις τέσσερις πηγές και παρουσιάζει στο χρήστη τις γνώσεις, τις ικανότητες, τη δημοτικότητα και τα υπόλοιπα ενδιαφέροντα στοιχεία που αναφέρθηκαν προηγουμένως. Βοηθά το χρήστη να γνωρίσει και να κατανοήσει τη δημόσια εργασία και δραστηριότητα που έχει πραγματοποιήσει και να μάθει σε πιο σημείο βρίσκεται σε σχέση με άλλους χρήστες. Αξίζει να αναφέρουμε ότι την παρούσα στιγμή είναι γνωστό ότι δεν υπάρχει παρόμοιο εργαλείο το οποίο να συνδυάζει πηγές από διαφορετικούς λογαριασμούς ενός μηχανικού λογισμικού. Αναμφίβολα το σύστημα αποτελεί μια πολύ ενδιαφέρουσα μέθοδο παρουσίασης του προφίλ μηχανικών λογισμικού. Επομένως η σημαντικότητα του συστήματος είναι εμφανής.

Επίσης το σύστημα βρίσκεται στην πρώτη έκδοσή του. Σε μεταγενέστερο στάδιο θα ήταν εφικτό να γίνει επέκταση του συστήματος ώστε να γίνει πιο πλούσιο σε ότι αφορά την πληροφορία των προφίλ και παράλληλα πιο πλήρης, συλλέγοντας επιπλέον δεδομένα.

Προχωρώντας θα μελετήσουμε τα αποτελέσματα των χρηστών οι οποίοι χρησιμοποίησαν το σύστημα και απάντησαν το σχετικό ερωτηματολόγιο. Θα κάνουμε μια ανάλυση των απαντήσεων θεωρώντας ότι οι βαθμολογίες 4 και 5 στις ερωτήσεις αποτελούν θετικές απαντήσεις, η βαθμολογία 3 ουδέτερες και οι 1 και 2 αρνητικές. Είχαμε αναφέρει ότι τα

ερωτηματολόγια στοχεύουν συγκεκριμένα πεδία τα οποία είναι η κατανόηση του συστήματος, η ευχρηστία, η χρησιμότητα, η επέκταση και η σύσταση.

Από πλευράς κατανόησης έχουν γίνει τρεις ερωτήσεις όπου στην πρώτη έχουμε λάβει θετικές απαντήσεις 12/15 και 3/15 ουδέτερες, στη δεύτερη 14/15 θετικές και 1/15 αρνητικές και στην τρίτη ερώτηση έχουμε 7/15 χρήστες οι οποίοι έδωσαν κατά μέσο όρο 2,28 αρνητικές ψήφους ο καθένας σε αποτελέσματα που δεν έχουν κατανοήσει, όσο αφορά το προφίλ του χρήστη. Δηλαδή από τις 12 αρνητικές απαντήσεις που θα μπορούσε να δώσει ο κάθε χρήστης και συνολικά 180 αρνητικές απαντήσεις και για τους 15, έχουμε λάβει 16/180 αρνητικές. Επομένως αναλύοντας αυτά τα αποτελέσματα συμπεραίνουμε ότι το σύστημα έχει ένα αρκετά υψηλό επίπεδο όσο αφορά την κατανόηση.

Συνεχίζοντας με την ευχρηστία, έχουν γίνει μια ερώτηση η οποία είχε ως αποτέλεσμα 13/15 θετικές ψήφους και 2/15 ουδέτερες. Άρα σχετικά με το πεδίο της ευχρηστίας μπορούμε να αποφανθούμε ότι το σύστημα είναι αρκετά εύχρηστο ώστε οι χρήστες να μπορούν να πλοηγηθούν σε αυτό.

Η επόμενη παράμετρος που εξετάσαμε είναι η χρησιμότητα του συστήματος, για την οποία έχουν απαντηθεί τρεις ερωτήσεις από τους χρήστες. Η πρώτη αφορά τα αποτελέσματα του προφίλ τα οποία δεν θεωρούν ότι είναι χρήσιμα. Συνολικά 6/15 χρήστες έδωσαν κατά μέσο όρο 1,33 αρνητικές ψήφους ο καθένας σε αποτελέσματα που δεν θεωρούν χρήσιμα. Δηλαδή από τις 12 αρνητικές απαντήσεις που θα μπορούσε να δώσει ο κάθε χρήστης και συνολικά 180 και για τους 15 χρήστες, έχουμε λάβει 8/180 αρνητικές απαντήσεις. Στη δεύτερη ερώτηση έχουν δοθεί 12/15 θετικές, 2/15 ουδέτερες και 1/15 αρνητικές ψήφοι. Στην Τρίτη και τελευταία ερώτηση έχουν δοθεί 12/15 θετικές, 1/15 ουδέτερη και 2/15 αρνητικές ψήφοι. Άρα είμαστε σε θέση να συμπεράνουμε ότι οι πλήστοι χρήστες βρίσκουν το σύστημα χρήσιμο.

Όσο αφορά τη σύσταση, έχει προσφερθεί μια ερώτηση για απάντηση από τους χρήστες. Η ερώτηση αυτή έχει λάβει 9/15 θετικές απαντήσεις, 5/15 ουδέτερες και 1/15 αρνητική. Επομένως σύμφωνα με τα αποτελέσματα το 60% των χρηστών σίγουρα θα πρότεινε το σύστημα και σε άλλους χρήστες, το 33% είτε θα το πρότεινε είτε όχι και ένα πολύ μικρό ποσοστό κάτω 1% κατά πάσα πιθανότητα δεν θα πρότεινε το σύστημα.

Την επέκταση θα την αναλύσουμε στο επόμενο υποκεφάλαιο μελλοντική εργασία που είναι πιο σχετικό με το συγκεκριμένο πεδίο.

7.2 Μελλοντική Εργασία

Όπως σε κάθε σύστημα που αναπτύσσεται υπάρχουν κάποια περιθώρια βελτίωσης και επέκτασης, έτσι και σε αυτό το σύστημα υπάρχουν. Πιο συγκεκριμένα μπορούν να προστεθούν αρκετές λειτουργίες που να βελτιώσουν το σύστημα και να προσφέρουν περισσότερες πληροφορίες στους χρήστες.

Κύρια επέκταση αποτελεί η λειτουργία εργοδότη. Η λειτουργία αυτή θα παρέχει τη δυνατότητα σε εργοδότες να αναζητούν μηχανικούς λογισμικού μέσα από το σύστημα. Αναλυτικότερα θα ήταν καλό να υλοποιηθεί και μια διαπροσωπεία αναζήτησης όπου οι εργοδότες θα έχουν την ικανότητα εκτός από το να αναζητήσουν, να προσαρμόσουν την αναζήτηση σύμφωνα με τα δικά τους κριτήρια χρησιμοποιώντας κάποια πεδία. Αφότου πραγματοποιήσουν την αναζήτηση, θα εμφανίζονται τα αποτελέσματα τα οποία θα αποτελούνται από τους μηχανικούς λογισμικού τους οποίους ταιριάζουν με την αναζήτηση που έχει γίνει. Στα αποτελέσματα θα παρουσιάζονται συνοπτικά κάποια στοιχεία για κάθε μηχανικό και αφού ο εργοδότης επιλέξει κάποιον, θα μπορεί να δει λεπτομερώς το προφίλ του. Επίσης θα ήταν χρήσιμο εκτός από την αναζήτηση, να υπάρχει η επιλογή ώστε να εμφανίζονται οι κορυφαίοι μηχανικοί λογισμικού ανά κατηγορία.

Μια άλλη επέκταση που θα μπορούσε να υλοποιηθεί στο μέλλον είναι η βελτίωση του χρήστη κατά την πάροδο του χρόνου. Αναλυτικότερα, όταν ο χρήστης δημιουργήσει το προφίλ του σε μια δεδομένη χρονική στιγμή, να χρησιμοποιούνται τα δεδομένα αυτά ώστε την επόμενη φορά που θα δημιουργήσει το προφίλ του να του εμφανίζονται διάφορες εξελίξεις σχετικά με το προφίλ. Για παράδειγμα να του εμφανίζει ότι σε ένα συγκεκριμένο πεδίο αυξήθηκε η δραστηριότητα του αφού έχει δημοσιεύσει κάποια tweets ή ότι έχει βελτιώσει τις γνώσεις του αφού από την τελευταία στιγμή που δημιούργησε το προφίλ του έχει δώσει κάποιες επιπλέον απαντήσεις στο Stack Overflow.

Επιπρόσθετα, εάν το σύστημα αυτό χρησιμοποιηθεί από ένα μεγάλο αριθμό χρηστών στο μέλλον, θα έχουν συλλεγεί πολλά στοιχεία σε μεγάλους όγκους δεδομένων. Επομένως θα ήταν πολύ χρήσιμο χρησιμοποιώντας αυτά τα δεδομένα να γίνουν κάποιες αναλύσεις ώστε να ανακαλυφθούν χρήσιμες πληροφορίες και παράλληλα να εξαχθούν χρήσιμα συμπεράσματα.

Τέλος υπάρχουν δυο ερωτήσεις στο ερωτηματολόγιο σχετικές με την επέκταση του συστήματος. Η πρώτη προσφέρει την ευκαιρία στους χρήστες να προτείνουν κάποιες δικές

τους εισηγήσεις στο ερωτηματολόγιο που αφορούν την επέκταση. Αξίζει να τις αναφέρουμε επειδή είναι πολύ ενδιαφέρουσες. Μια ιδέα είναι να πραγματοποιούνται συστάσεις χρηστών από το σύστημα ώστε να μπορούν άλλοι χρήστες να τους ακολουθήσουν. Πιο συγκεκριμένα θα μπορούσε να υλοποιηθεί μια μέθοδος σύστασης χρηστών με βάση κάποιες παραμέτρους, όπως για παράδειγμα αν ο χρήστης είναι πιο έμπειρος σε μια συγκεκριμένη γλώσσα, τότε να του προτείνει χρήστες που είναι επίσης έμπειροι σε αυτή τη γλώσσα. Άλλη πολύ καλή ιδέα είναι η ανάλυση αρχείων κώδικα των χρηστών και εντοπισμός κοινών λαθών που τυχόν να έχουν στον κώδικα τους. Επιπρόσθετα μια πολύ καλή σύσταση για επέκταση αποτελεί η ιδέα να εδραιωθεί ένα δίκτυο στο σύστημα ώστε οι χρήστες που έχουν συνεργαστεί στο παρελθόν να μπορούν να συσχετίζονται μεταξύ τους. Αυτή η συσχέτιση μπορεί να λειτουργεί με τον τρόπο που λειτουργούν και οι ακόλουθοι ή να υπάρχει ένας τρόπος επικοινωνίας μεταξύ χρηστών όπως μηνύματα και ειδοποιήσεις ή ακόμα να επιτρέπεται η συμμετοχή τους σε διάφορες ομάδες. Η τελευταία ιδέα που έχει προταθεί είναι να δημιουργείται το βιογραφικό του χρήστη ή μέρος αυτού με μια αυτοματοποιημένη διαδικασία και το οποίο αφορά τα δεδομένα του χρήστη που εμφανίζονται στο προφίλ του.

Σχετικά με τη δεύτερη ερώτηση, αφορά το κατά πόσο οι χρήστες επιθυμούν να προστεθούν στο σύστημα οι λογαριασμοί του Facebook και του LinkedIn και από τα αποτελέσματα γνωρίζουμε ότι 9/15 και 11/15 χρήστες αντίστοιχα πιστεύουν ότι θα ήταν χρήσιμο. Οι λογαριασμοί αυτοί είναι χρήσιμοι ώστε να πλαισιώνουν τα μέχρι στιγμής δεδομένα και να κάνουν το προφίλ πιο πλήρες. Το LinkedIn συγκεκριμένα θα είναι κάτι νέο για το σύστημα αφού δεν χρησιμοποιείται κάποιο σύστημα που να αφορά καθαρά το βιογραφικό των χρηστών.

Βιβλιογραφία

- [1] G. Gousios, B. Vasilescu, A. Serebrenik και A. Zaidman, «Lean GHTorrent: GitHub Data on Demand,» σε *11th Working Conference on Mining Software Repositories (pp. 384-387)*. ACM, 2014.
- [2] J. Lawrance, S. Jung και C. Wiseman, «Git on the Cloud in the Classroom,» σε *44th ACM technical symposium on Computer science education*. ACM, 2013.
- [3] Wikipedia, «Stack Overflow,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Stack_Overflow.
- [4] A. Capiluppi, A. Serebrenik και L. Singer, «Assessing Technical Candidates on the Social Web,» σε *IEEE*, 30(1), pp.45-51, 2013.
- [5] S. Chacon και B. Straub, «CHAPTER 1: Getting Started,» σε *Pro Git*, Apress.
- [6] Wikipedia, «GitHub,» [Ηλεκτρονικό]. Available: <https://en.wikipedia.org/wiki/GitHub>.
- [7] S. O. Team, «Stack Overflow,» [Ηλεκτρονικό]. Available: <http://stackoverflow.com/company/about>.
- [8] Wikipedia, «Social media,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Social_media.
- [9] A. Java, X. Song, T. Finin και B. Tseng, «Why We Twitter: Understanding Microblogging,» σε *9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*. ACM, 2007.
- [10] H. Kwak, C. Lee, H. Park και S. Moon, «What is Twitter, a Social Network or a News Media?,» σε *19th international conference on World wide web*. ACM, Daejeon, Korea, 2010.
- [11] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach και T. Berners-Lee, *Hypertext Transfer Protocol -- HTTP/1.1*, Rfc 2616, June 1999.
- [12] Wikipedia, «POST (HTTP),» [Ηλεκτρονικό]. Available: [https://en.wikipedia.org/wiki/POST_\(HTTP\)](https://en.wikipedia.org/wiki/POST_(HTTP)).
- [13] «HTTPFUL,» [Ηλεκτρονικό]. Available: <http://phphttpclient.com/>.
- [14] N. Osman, «Stack.PHP,» [Ηλεκτρονικό]. Available: <http://stackphp.quickmediasolutions.com/>.
- [15] «Twitter REST API,» [Ηλεκτρονικό]. Available: <https://dev.twitter.com/rest/public>.

- [16] Twitter, «Application-only authentication,» [Ηλεκτρονικό]. Available: <https://dev.twitter.com/oauth/application-only>.
- [17] «Computer Hope,» [Ηλεκτρονικό]. Available: <http://www.computerhope.com/more.htm>.
- [18] O. Smith, «morris.js,» [Ηλεκτρονικό]. Available: <http://morrisjs.github.io/morris.js/>.

Παράρτημα Α

Στο παράρτημα αυτό υπάρχουν οι ετικέτες από τις εκατό κορυφαίες του Stack Overflow, οι οποίες επιλέγηκαν για να χρησιμοποιηθούν στο σύστημα.

- javascript
- java
- c#
- php
- android
- jquery
- python
- html
- c++
- ios
- mysql
- sql
- css
- asp.net
- objective-c
- ruby-on-rails
- .net
- iphone
- c
- arrays
- sql-server
- angularjs
- ruby
- json
- ajax
- regex
- xml

- **asp.net-mvc**
- **r**
- **ubuntu**
- **haskell**
- **unix**
- **linux**
- **opencv**
- **wpf**
- **powershell**
- **django**
- **node.js**
- **database**
- **xcode**
- **vb.net**
- **eclipse**
- **string**
- **windows**
- **wordpress**
- **matlab**
- **perl**
- **web-services**
- **scala**
- **delphi**
- **maven**
- **bash**
- **swing**
- **rest**
- **http**