



Πανεπιστήμιο
Ιωαννίνων

ΤΜΗΜΑ: ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΑΥΤΟΜΑΤΗ ΕΚΤΙΜΗΣΗ ΕΓΓΥΤΗΤΑΣ ΕΠΙΣΤΗΜΟΝΙΚΟΥ
ΕΡΓΟΥ ΕΡΕΥΝΗΤΩΝ ΜΕ ΧΡΗΣΗ ΤΕΧΝΙΚΩΝ
ΚΑΤΑΝΟΗΣΗΣ ΦΥΣΙΚΗΣ ΓΛΩΣΣΑΣ

ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΦΟΙΤΗΤΗ: ΓΚΕΣΟΥΛΗΣ ΛΑΜΠΡΟΣ

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΓΚΟΓΚΟΣ ΧΡΗΣΤΟΣ

ΑΡΤΑ 2021



Πανεπιστήμιο
Ιωαννίνων

DEPARTMENT: COMPUTER SCIENCE & TELECOMMUNICATION

THESIS

Automated estimation of researchers' scientific work using NLP techniques

STUDENT: GESOULIS LAMBROS

SUPERVISOR: GOGOS CHRISTOS

ARTA 2021

ΔΗΛΩΣΗ ΜΗ ΛΟΓΟΚΛΟΠΗΣ

Δηλώνω ενυπογράφως και γνωρίζοντας τις συνέπειες του νόμου περί πνευματικών δικαιωμάτων ότι είμαι αποκλειστικός συγγραφέας της παρούσας Πτυχιακής Εργασίας και δεν αποτελεί προϊόν αντιγραφής. Έχω αναφέρει όλες τις πηγές που χρησιμοποιήθηκαν για την συγγραφή της και περιλαμβάνονται στην βιβλιογραφία της.

ΓΚΕΣΟΥΛΗΣ ΛΑΜΠΡΟΣ

ΑΜ: 11135

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα πτυχιακή εργασία αποτελεί το επιστέγασμα των σπουδών μου στο Τμήμα Πληροφορικής και Τηλεπικοινωνιών του Πανεπιστημίου Ιωαννίνων. Θα ήθελα να ευχαριστήσω όλους τους καθηγητές του τμήματος για όλες αυτές τις γνώσεις που μου μετέδωσαν και κυρίως τον κ. Γκόγκο Χρήστο για την εμπιστοσύνη που μου έδειξε για την υλοποίηση αυτής της εργασίας.

Θα ήθελα επίσης να απευθύνω τις ευχαριστίες μου σε όλους εκείνους που πίστεψαν στην επίτευξη του στόχου μου και ιδιαίτερα στην οικογένεια μου για όλη την υποστήριξη που είχα τα χρόνια των σπουδών μου.

ΠΕΡΙΛΗΨΗ

Η πεποίθηση για την ολοένα ταχύτερη ανάπτυξη της τεχνολογίας είναι a priori εδραιωμένη στη σύγχρονη ζωή των ανθρώπων του 21^{ου} αιώνα. Απόρροια αυτής της συνθήκης είναι η δημιουργία εργαλείων και εφαρμογών, τα οποία εξυπηρετούν και διευκολύνουν κάθε πλώνα της ανθρώπινης ζωής, σε ατομικό, κοινωνικό και επαγγελματικό επίπεδο. Ειδικότερα η ανάπτυξη εφαρμογών απαιτεί λεπτομερή οργάνωση και ακριβή σχεδιασμό με απώτερο σκοπό την εύρυθμη λειτουργία τους για την εξυπηρέτηση των ανθρώπινων αναγκών.

Η συγκεκριμένη πτυχιακή εργασία σχετίζεται με τη δημιουργία μίας εφαρμογής, η οποία αφορά την αυτόματη εκτίμηση εγγύτητας του επιστημονικού έργου συγκεκριμένων ερευνητών με την αξιοποίηση τεχνικών κατανόησης της φυσικής γλώσσας. Η διαδικασία, η οποία ακολουθήθηκε κατά τη διάρκεια εκπόνησης της πτυχιακής αυτής εργασίας αποτελείται από τρία στάδια: την μελέτη, τη σχεδίαση και τέλος την υλοποίηση της διαδικτυακής εφαρμογής. Σε ένα πιο αναλυτικό επίπεδο, η εφαρμογή θα παρουσιάζει τα ποσοστά εγγύτητας μεταξύ των επαγγελματικών προφίλ ερευνητών, οι οποίοι συμπεριλαμβάνονται στην εφαρμογή και με βάση τα δημοσιευμένα άρθρα τους θα δίνονται ακριβή διαγράμματα, τα οποία θα απεικονίζουν τα άνωθεν ποσοστά εγγύτητάς τους.

Λέξεις κλειδιά: *NLP, Python, Flask Framework, bag of words, MySql, research, επιστημονικά άρθρα, ερευνητές, διαδικτυακή εφαρμογή.*

ABSTRACT

The belief in the ever more rapid development of technology is a priori embedded in the modern life of people in the 21st century. The result of this condition is the creation of tools and applications that serve and facilitate every pillar of human life, on an individual, social and professional level. In particular, the development of applications requires detailed organization and precise planning with the ultimate goal of their smooth operation to serve human needs.

This thesis is related to the creation of an application, which concerns the automatic estimation of the proximity of the scientific work of specific researchers using natural language understanding techniques. The process that was followed during the preparation of this thesis consists of three stages: the study, the design and finally the implementation of the web application. At a more detailed level, the application will present the proximity rates between the professional profiles of researchers included in the application and, based on their published articles, precise charts will be provided, which will illustrate their upper proximity rates.

Keywords: *NLP, Python, Flask Framework, bag of words, MySql, research, scientific articles, researchers, web application.*

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1 ΕΙΣΑΓΩΓΗ	x
1.1 Σκοπός της πτυχιακής εργασίας.....	x
1.2 Δομή της πτυχιακής εργασίας	xi
2 ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ	xiii
2.1 Λογισμικό και Τεχνολογία Λογισμικού	xiii
2.2 Μοντέλα Σχεδιασμού Λογισμικού.....	xiv
2.2.1 Το Μοντέλο Καταρράκτη (Waterfall Model)	xv
2.2.1.1 Η λειτουργία του μοντέλου καταρράκτη (waterfall model)	xv
2.2.1.2 Τα πλεονεκτήματα του μοντέλου καταρράκτη (Waterfall Model)	xvi
2.2.1.3 Τα μειονεκτήματα του μοντέλου καταρράκτη (Waterfall Model).....	xvii
2.2.2 Το Ελικοειδές Μοντέλο (Spiral Model).....	xviii
2.2.3 Το Αστεροειδές Μοντέλο	xix
2.2.4 Το μοντέλο του ανθρωποκεντρικού σχεδιασμού κατά ISO9241-210:2010.....	xx
3. ΕΠΙΣΤΗΜΟΝΙΚΑ ΑΡΘΡΑ	xxiii
3.1 Εισαγωγή.....	xxiii
3.2 Επιστημονικό/ Ερευνητικό Άρθρο.....	xxiii
3.3 Δομή Επιστημονικού Άρθρου.....	xxiv
3.4 Αξιολόγηση Επιστημονικού Άρθρου	xxvi
3.5 Πλατφόρμες Πηγών Επιστημονικών Άρθρων	xxvii
4. ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ ΚΑΙ NLP	xxx
4.1 Η Μάθηση.....	xxx
4.2 Μηχανική Μάθηση	xxxι
4.2.1 Είδη Μηχανικής Μάθησης.....	xxxii
4.2.2 Αλγόριθμοι Μηχανικής Μάθησης	xxxiii
4.3 Ο όρος NLP	xxxiii
4.3.1 Ο Ορισμός του NLP.....	xxxiii
4.3.2 NLP – Ιστορική Εξέλιξη	xxxiv
4.3.3 NLP – Στάδια.....	xxxv
4.3.4 NLP & Μηχανική Μάθηση	xxxvi
5. Ο ΣΧΕΔΙΑΣΜΟΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ	xxxvii
5.1 Εισαγωγή.....	xxxvii
5.2 Απαιτήσεις λογισμικού	xxxvii
5.2.1 Περιπτώσεις χρήσης (use cases).....	xxxvii
5.3 Τα εργαλεία ανάπτυξης λογισμικού.....	xlii
5.3.1 Η γλώσσα προγραμματισμού Python	xlii

5.3.2 Η γλώσσα SQL	xliv
5.3.3 Frameworks	xlvi
5.3.3.1 Τα πλεονεκτήματα των frameworks.....	xlvii
5.3.3.2 Τα μειονεκτήματα των Frameworks	xlvii
5.3.3.3 Η λειτουργία	xlviii
5.3.3.4 Model.....	xlviii
5.3.3.5 View	xlviii
5.3.3.6 Template.....	xlviii
5.3.4 HTML.....	xliv
5.3.5 CSS	1
5.3.6 JavaScript	1
6. Η βιβλιοθήκη Scikit-learn	lii
6.1 Bag of words	lii
6.2 Cosine Similarity	liv
7.Συμπεράσματα – Περιορισμοί της έρευνας – Μελλοντικές Προεκτάσεις.....	lv

Πίνακας περιεχομένων Εικόνων

Εικόνα 1: Μοντέλα Κύκλου Ζωής Λογισμικού	xiv
Εικόνα 2: Μοντέλο Καταρράκτη (Waterfall Model)	xviii
Εικόνα 3: Το Ελικοειδές Μοντέλο (Spiral Model)	xix
Εικόνα 4: Το Αστεροειδές Μοντέλο	xx
Εικόνα 5: Το μοντέλο του ανθρωποκεντρικού σχεδιασμού κατά ISO9241-210	xxi
Εικόνα 6: Βασικά Μέρη ενός Επιστημονικού Άρθρου	xxv
Εικόνα 7: Επιπλέον κατηγορίες ως μέρη ενός επιστημονικού άρθρου	xxvi
Εικόνα 8: Ορισμοί Βιβλιογραφικών Δεικτών	xxvii
Εικόνα 9: Δημοφιλείς Γλώσσες προγραμματισμού	xliv
Εικόνα 10: Python & Classes	xlv
Εικόνα 11: Λειτουργία MVT Model	xlix
Εικόνα 12: Bag of words – Παράδειγμα	lii
Εικόνα 13: Κώδικας Bag of words	liii
Εικόνα 14: Κώδικας cosine similarity	liv

1 ΕΙΣΑΓΩΓΗ

1.1 Σκοπός της πτυχιακής εργασίας

Ο αιώνας που διανύουμε είναι ο αιώνας της τεχνολογίας και της πληροφόρησης. Οτιδήποτε χρειάζεται ο μέσος χρήστης είναι πια διαθέσιμο στο Διαδίκτυο. Κάθε άνθρωπος είναι δυνατόν να καλύψει διαφορετικές ανάγκες μέσα από την αξιοποίηση των διαδικτυακών εργαλείων, ανεξάρτητα από το μέρος, στο οποίο βρίσκεται καθώς το Διαδίκτυο μας παρέχει τη δυνατότητα χρησιμοποίησής του από κάθε πιθανό σημείο σύνδεσης με αυτό. Ειδικότερα, εκατομμύρια χρήστες φαίνεται να χρησιμοποιούν σε καθημερινή και συστηματική βάση υπηρεσίες όπως:

- Το Ηλεκτρονικό ταχυδρομείο: πρόκειται για μια υπηρεσία με εκατομμύρια χρήστες και πολλαπλά πλεονεκτήματα έναντι του συμβατικού ταχυδρομείου,
- Η VoIP: γνωστή και ως τηλεφωνία μέσω διαδικτύου, η οποία ειδικά το πρώτο εξάμηνο του 2020, γνώρισε ιδιαίτερη άνθιση με τις διάφορες μορφές τηλεκπαίδευσης και τηλεδιάσκεψης, που είναι δυνατόν να προσφέρει στους χρήστες της,
- Ο Παγκόσμιος Ιστός: γνωστός και ως World Wide Web. Πρόκειται για τη διασημότερη υπηρεσία του Διαδικτύου. Οι ιστοσελίδες της συγκεκριμένης υπηρεσίας διαθέτουν επισκεψιμότητα, που ξεπερνά τα τρία εκατομμύρια χρήστες καθημερινά. Στον Παγκόσμιο Ιστό υπάρχει κάθε είδους πληροφορία, είτε αυτή σχετίζεται με την ενημέρωση, είτε με την διασκέδαση, ή έγκειται σε επιστημονικούς σκοπούς. Ειδικότερα, ο Παγκόσμιος Ιστός είναι η δεξαμενή για διάφορες επιστημονικές έρευνες, οι οποίες είναι εφικτό να αποτελέσουν τη διερευνητική αφετηρία για τη μελλοντική διεξαγωγή και άλλων ερευνών. Υπάρχει πληθώρα ιστοσελίδων που παρέχουν πρόσβαση σε δημοσιευμένες έρευνες και επιστημονικά άρθρα (scientific paper). Το Google Scholar, το ScienceDirect, το Scopus αποτελούν σημαντικά παραδείγματα ιστοσελίδων του Παγκόσμιου Ιστού, οι οποίες περιέχουν πληθώρα ερευνών καλύπτοντας μάλιστα ένα ευρύ φάσμα επιστημονικού ενδιαφέροντος.

Στην παρούσα εργασία, θα γίνει μια προσπάθεια αξιοποίησης των δυνατοτήτων, που ο Παγκόσμιος Ιστός παρέχει αναφορικά με την αναζήτηση πληροφοριών για συγκεκριμένες επιστημονικές δημοσιεύσεις. Συγκεκριμένα, ο σκοπός της εργασίας αυτής, είναι η

ανάπτυξη ενός λογισμικού, το οποίο θα συγκρίνει τις ερευνητικές δημοσιεύσεις συγγραφέων ενός συγκεκριμένου επιστημονικού πεδίου και θα εντοπίζει πιθανές συγκλίσεις τους. Το λογισμικό αυτό θα λειτουργήσει με βάση έναν προκαθορισμένο αριθμό συγγραφέων. Ο σχεδιασμός και η λειτουργία του συγκεκριμένου λογισμικού είναι δυνατόν να προωθήσουν την πιθανή συνεργασία μεταξύ των ερευνητών ενός κοινού επιστημονικού πεδίου ενώ συγχρόνως μπορεί να συμβάλει και στον εντοπισμό στοιχείων λογοκλοπής μεταξύ των ήδη δημοσιευμένων επιστημονικών άρθρων.

1.2 Δομή της πτυχιακής εργασίας

Η παρούσα πτυχιακή εργασία διαιρείται σε επτά κεφάλαια και στη συνέχεια παρουσιάζεται μια σύνοψη των περιεχομένων του εκάστοτε κεφαλαίου.

Το πρώτο κεφάλαιο είναι εισαγωγικό και περιλαμβάνει τη διατύπωση του ερευνητικού μας σκοπού καθώς ακόμη και τον ακριβή καθορισμό της δομής της έρευνάς μας.

Στο δεύτερο κεφάλαιο παρουσιάζονται οι βασικές μέθοδοι ανάπτυξης λογισμικού που ακολουθούνται στα πληροφοριακά συστήματα στο Διαδίκτυο

Το τρίτο κεφάλαιο αφορά τον καθορισμό της έννοιας του επιστημονικού άρθρου. Η αξιοποίηση των επιστημονικών άρθρων αποτελεί την αφετηρία για την ολοκλήρωση του πρακτικού κεφαλαίου της έρευνας. Για τον λόγο αυτό κρίνεται ως αναγκαίος ο ακριβής ορισμός του.

Εν συνεχεία, στο τέταρτο κεφάλαιο της παρούσας πτυχιακής εργασίας θα γίνει μία προσπάθεια να παρουσιαστεί η έννοια της μηχανικής μάθησης. Ειδικότερα, το ενδιαφέρον μας θα προσανατολιστεί στο NLP καθώς αξιοποιήθηκε πλήρως για την εκπόνηση της έρευνάς μας. Ο ορισμός, η ιστορική εξέλιξη του NLP καθώς και τα στάδια εφαρμογής τους θα παρουσιαστούν σε ακόλουθο κεφάλαιο.

Στο πέμπτο κεφάλαιο, θα γίνει πλήρης αναφορά στο σχεδιασμό της διαδικτυακής εφαρμογής. Ειδικότερα, θα αναφερθούμε στις προδιαγραφές, τις περιπτώσεις χρήσης αλλά και στα εργαλεία ανάπτυξης που χρησιμοποιούνται στο πρακτικό κομμάτι. Συγκεκριμένα, θα δοθεί έμφαση στην γλώσσα προγραμματισμού Python, η οποία τα τελευταία χρόνια χαίρει ιδιαίτερης προτίμησης από την προγραμματιστική κοινότητα,

καθώς είναι σε θέση να παρέχει σημαντικές λειτουργίες χωρίς να απαιτεί περίπλοκες εντολές και συντακτικό. Επίσης, θα γίνει αναφορά και στην SQL διότι θα αξιοποιηθεί ερευνητικά για την αποθήκευση των άρθρων προς σύγκριση. Συγχρόνως, στο συγκεκριμένο κεφάλαιο, θα αναφερθούμε και σε άλλα χρήσιμα εργαλεία προγραμματισμού όπως είναι η γλώσσα Java Script, καθώς και τα εργαλεία HTML, CSS και Flask Framework.

Στο έκτο κεφάλαιο, θα διατυπωθούν αναλυτικά τα βήματα υλοποίησης του λειτουργικού λογισμικού, το οποίο αποτέλεσε και τον κεντρικό άξονα της παρούσας εργασίας.

Στο έβδομο κεφάλαιο θα γίνει η παρουσίαση του λογισμικού ενώ θα πραγματοποιηθεί και η ανάλυση των αποτελεσμάτων, που προκύπτουν από την χρήση του.

Στα δύο τελευταία κεφάλαια της παρούσας έρευνας, αναλογικά δηλαδή το όγδοο και ένατο κατά σειρά κεφάλαιο, θα παρουσιαστούν τα συμπεράσματα, τα οποία προέκυψαν από την εκπόνηση της πτυχιακής μας εργασίας καθώς ακόμη και οι διάφορες αναφορές μας σχετικά με τυχόν βελτιώσεις ή και περιορισμούς του λειτουργικού λογισμικού, που σχεδιάσαμε και φέραμε εις πέρας.

2 ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ

Η δημιουργία ενός λογισμικού με απώτερο σκοπό τη διερεύνηση επιστημονικών άρθρων και τον εντοπισμό των πιθανών συγκλίσεων τους, αποτέλεσε τον βασικότερο ερευνητικό σκοπό της παρούσας πτυχιακής εργασίας. Στο σημείο αυτό κρίνεται επομένως αναγκαίο να αναφερθούμε στον ευρύτερο ορισμό της έννοιας ενός λογισμικού καθώς ακόμη και στους τρόπους ανάπτυξής του.

2.1 Λογισμικό και Τεχνολογία Λογισμικού

Με τη χρήση του όρου «λογισμικό» εννοούμε:

- Προγράμματα ηλεκτρονικού υπολογιστή, τα οποία όταν εκτελούνται επιτυγχάνουν επιθυμητά αποτελέσματα και επιδόσεις,
- Δομές δεδομένων που επιτρέπουν στα προγράμματα να διαχειρίζονται κάθε είδους δεδομένα,
- Κείμενα, διαγράμματα, μορφές παράστασης, που περιγράφουν την δομή, την λειτουργία και την χρήση των προγραμμάτων. (Βεσκούδης, 2015)

Ο κλάδος της Πληροφορικής που ασχολείται με τον εντοπισμό και τη θεμελίωση μεθόδων για την περιγραφή, τη δημιουργία και τη συντήρηση ενός λογισμικού είναι γνωστός στην βιβλιογραφία με την ονομασία «Τεχνολογία Λογισμικού». (Βεσκούδης, 2015)

Η Τεχνολογία Λογισμικού δεν ασχολείται με ένα μεμονωμένο κομμάτι του λογισμικού. Αντίθετα ακολουθεί τη διαδικασία για την υλοποίηση και την ολοκλήρωση της κατασκευής ενός λογισμικού από τα αρχικά βήματα σχεδιασμού του. Ακολουθεί, δηλαδή, όλο το κύκλο ζωής του λογισμικού.

Για το κύκλο ζωής ενός λογισμικού έχουν σχεδιαστεί αρκετά μοντέλα. Ένα μοντέλο κύκλου ζωής ενός λογισμικού περιλαμβάνει όλα τα βήματα της υλοποίησής του, από την γέννηση της ιδέας μέχρι το τελικό αποτέλεσμα. Σε κάθε στάδιο ζωής του λογισμικού, το μοντέλο αναφέρει τις ενέργειες που έγιναν είτε αυτό συμπεριλαμβάνει UML διαγράμματα είτε προγραμματισμού είτε οποιαδήποτε άλλη ενέργεια.

Τα μοντέλα κύκλου ζωής λογισμικού διακρίνονται σε ακολουθιακά και επαναληπτικά. Στα ακολουθιακά οι διάφορες φάσεις είναι διακριτές και ξεχωριστές μεταξύ τους, ενώ στα επαναληπτικά μοντέλα η ανάπτυξη λογισμικού πραγματοποιείται ανά τμήματα. (Βεσκούδης, 2015)



Εικόνα 1: Μοντέλα Κύκλου Ζωής Λογισμικού

2.2 Μοντέλα Σχεδιασμού Λογισμικού

Το μοντέλο κύκλου ζωής ενός λογισμικού περιλαμβάνει ένα σύνολο από μεθοδολογίες, οι οποίες περιγράφουν τις διαφορετικές φάσεις εργασίας, τις ενέργειες και τα προϊόντα, που παράγονται σε κάθε φάση εργασίας, τη διαδοχή των φάσεων και τέλος την έκταση.

Αναφέρθηκε ο όρος μοντέλο κύκλου ζωής λογισμικού. Σε αυτά τα μοντέλα συγκαταλέγονται διάφορες μεθοδολογίες οι οποίες αποτυπώνουν με τη σειρά τους τα βήματα και τις ενέργειες που απαιτούνται για την τελική μορφή του λογισμικού.

Τα μοντέλα είναι τα εξής:

- Μοντέλο καταρράκτη,
- Ελικοειδές Μοντέλο,
- Αστεροειδές Μοντέλο,
- Ανθρωποκεντρική Μεθοδολογία

Το κάθε μοντέλο ανάπτυξης λογισμικού έχει τις δικές του αρχές με βάση τις οποίες πραγματοποιείται η ανάπτυξή του.

2.2.1 Το Μοντέλο Καταρράκτη (Waterfall Model)

Ένα από τα πιο ευρέως διαδεδομένα μοντέλα κύκλου ζωής είναι το μοντέλο καταρράκτη (waterfall model), το οποίο και απεικονίζεται στο σχήμα ένα (1) κάτωθι. Το μοντέλο αυτό δομείται γύρω από μία κεντρική ιδέα με βάση την οποία το σύστημα λογισμικού αναπτύσσεται περνώντας στην ολότητά του από διαδοχικές και διακριτές μεταξύ τους φάσεις (Βεσκούδης, 2015). Η κάθε επιμέρους φάση θεωρείται ως ολοκληρωμένη με την παραγωγή συστατικών λογισμικού. Αξίζει ακόμη να σημειώσουμε το εξής: κατά την περάτωση, όπως ήδη αναφέραμε, μίας επιμέρους φάσης ακολουθεί και μία διαδικασία επαλήθευσης και επικύρωσης των παραγόμενων προϊόντων/ συστατικών λογισμικού. Κατά την εξέλιξη αυτής της διαδικασίας, αποφασίζεται η μετάβαση ή μη στην επόμενη διακριτή φάση του συγκεκριμένου μοντέλου (Boehm, 1984). Το λογισμικό παρουσιάζεται πλήρες από την επιμέρους φάση της συνένωσης και έπειτα. Σύμφωνα μάλιστα με τον Scacchi (1987) το πιο σημαντικό χαρακτηριστικό του συγκεκριμένου μοντέλου κύκλου ζωής είναι το ακόλουθο: για να μπορέσει να ξεκινήσει μία φάση πρέπει να έχει ήδη ολοκληρωθεί πλήρως η προηγούμενή της στην ακολουθία του μοντέλου καταρράκτη. Η ανάπτυξη επομένως, χαρακτηρίζεται ως ακολουθιακή καθώς όπως έχει ήδη επισημανθεί, οι επιμέρους φάσεις από τις οποίες περνάει το λογισμικό, είναι διακριτές και ακολουθούν η μία από την άλλη (Boehm, 1984).

2.2.1.1 Η λειτουργία του μοντέλου καταρράκτη (waterfall model)

Η λειτουργία του συγκεκριμένου μοντέλου στηρίζεται σε διακριτές μεταξύ τους φάσεις, οι οποίες είναι οι ακόλουθες:

- ✚ **Καθορισμός απαιτήσεων:** η λειτουργία του μοντέλου καταρράκτη έχει ως αφετηρία τον καθορισμό των απαιτήσεων από το ίδιο το σύστημα και το λογισμικό. Κατά το στάδιο αυτό της σχεδίασης, ορίζονται επακριβώς οι μονάδες, που θα αποτελέσουν το λογισμικό. Παράλληλα σε αυτή τη φάση του καθορισμού, γίνεται εξίσου αναφορά για τις συσχετίσεις μεταξύ των μονάδων (Williams, 1984). Θα ήταν ακριβές, στο σημείο αυτό, να αναφερθούμε στο εξής: ο καθορισμός πραγματοποιείται σε περισσότερα από ένα επίπεδα λεπτομέρειας. Επίπεδα, τα οποία εξαρτώνται φυσικά από την πολύπλοκη φύση του λογισμικού (Williams, 1984).

- ✚ **Λεπτομερής σχεδίαση:** κατά την εξελικτική διαδικασία του συγκεκριμένου μοντέλου, το επόμενο βήμα στην σειρά των φάσεων κατέχει η λεπτομερής σχεδίασή του. Στο σημείο αυτό της σχεδίασης, αποφασίζεται η εσωτερική δομή, που θα διαθέτει κάθε ξεχωριστή μονάδα του λογισμικού. Με τη σειρά της η κάθε μονάδα λογισμικού, κατά τρόπο πρακτικό, αντιστοιχίζεται με μία μονάδα πηγαίου κώδικα (Βεσκούδης, 2015). Η διαδικασία αυτή του καθορισμού συνάδει με την συγκέντρωση όλων των απαραίτητων στοιχείων, όπως είναι για παράδειγμα οι αλγόριθμοι ή οι διάφορες δομές δεδομένων, με απώτερο σκοπό το αμέσως επόμενο στάδιο, δηλαδή η συγγραφή του πηγαίου κώδικά, να αναδειχθεί σε μία διαδικασία διεκπεραίωσης και μόνο (Williams, 1984).
- ✚ **Υλοποίηση λογισμικού:** ακολουθεί η υλοποίηση του κώδικα και μετέπειτα η συνένωση όλων των μονάδων σε ένα σύστημα.
- ✚ **Έλεγχος λογισμικού:** στη συνέχεια, απαραίτητη φάση του μοντέλου καταρράκτη (waterfall model) αποτελεί ο έλεγχος του λογισμικού. Σε αυτό το σημείο, ελέγχεται η σχεδίαση και κατά πόσο είναι λειτουργικό το λογισμικό καθώς και εάν ικανοποιεί τις απαιτήσεις που προαναφέρθηκαν. Καθώς ολοκληρώνεται και η φάση του ελέγχου, το λογισμικό είναι έτοιμο για να παραδοθεί στον πελάτη. Ξεκινά λοιπόν η παραγωγική λειτουργία του λογισμικού.
- ✚ **Διαχείριση και συντήρηση:** καθώς αρχίζει η παραγωγική λειτουργία, την οποία και αναφέραμε, συγχρόνως ξεκινάει η διαχείριση και η συντήρησή του. Στο μοντέλο καταρράκτη (waterfall model) το λογισμικό παραμένει λειτουργικό ενώ παράλληλα ενημερώνεται ή συντηρείται ανάλογα τις ανάγκες που προκύπτουν. (Μητρόπουλος & Δουληγέρης, 2015). Το μοντέλο αυτό, συνήθως, προτείνεται σε συστήματα προσανατολισμένα για το Διαδίκτυο.

2.2.1.2 Τα πλεονεκτήματα του μοντέλου καταρράκτη (Waterfall Model)

- ✚ Αποτελεί μία από τις παλαιότερες μεθόδους, αναφορικά με την ανάπτυξη του λογισμικού (Συμεωνίδης, 2015),
- ✚ Η διαδικασία, που ακολουθείται στο μοντέλο καταρράκτη (Waterfall Model) είναι ιδιαίτερα κατανοητή (Furht, 2008),
- ✚ Επιπρόσθετα, η μέθοδος αυτή είναι σύμφωνη με τις προδιαγραφές του πελάτη,
- ✚ Τα διακριτά βήματα του μοντέλου καταρράκτη (Waterfall Model) αποτέλεσαν εν συνεχεία τη βάση, επάνω στην οποία στηρίχθηκαν και άλλες μέθοδοι,

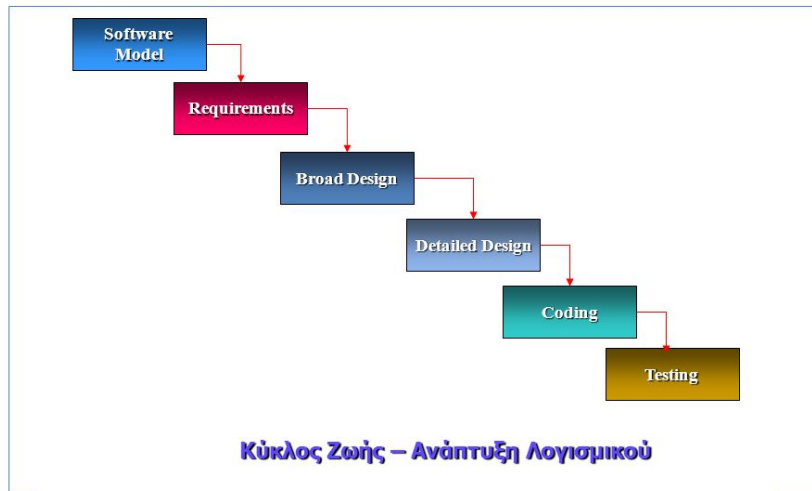
- ✦ Ο τρόπος λειτουργίας του εν λόγω μοντέλου, εξυπηρετεί τον καταμερισμό εργασίας μεταξύ των διάφορων φορέων, δηλαδή μεταξύ των προγραμματιστών, των αναλυτών και των πελατών (Συμεωνίδης, 2015).

2.2.1.3 Τα μειονεκτήματα του μοντέλου καταρράκτη (Waterfall Model)

Το μοντέλο αυτό διαθέτει ωστόσο και ορισμένα μειονεκτήματα:

- ✦ Το μοντέλο αυτό παρουσιάζει μία γραμμικότητα, η οποία ωστόσο συναντάται αρκετά σπάνια σε ένα πραγματικό έργο,
- ✦ Η ανάλυση αλλά και ο προσδιορισμός των απαιτήσεων δεν μπορούν συνήθως να οδηγηθούν σε περάτωση στην αρχή ενός έργου,
- ✦ Από τη χρονική στιγμή, κατά την οποία θα ξεκινήσει το έργο μέχρι και την πρώτη παράδοσή του, μεσολαβεί ένα αρκετά μεγάλο διάστημα. Στη διάρκεια αυτού του διαστήματος, μπορούμε μόνο να παρουσιάσουμε και να παραδώσουμε στον πελάτη την τεκμηρίωσή του,
- ✦ Κατά συνέπεια, οι πελάτες καθώς και οι διάφοροι χρήστες του συστήματος καθυστερούν στο να λάβουν μία λειτουργική του εικόνα. Το στοιχείο αυτό καθιστά εύκολη την υποτίμηση της συνολικής προσπάθειας,
- ✦ Ο καταμερισμός εργασίας, ο οποίος μάλιστα έχει ήδη αναφερθεί, μεταξύ των εμπλεκόμενων προγραμματιστών, αναλυτών και πωλητών, ενδέχεται να επιφέρει αρνητικά αποτελέσματα στη διεκπεραίωση του λογισμικού (Pressman, 2010).

Μοντέλο Καταρράκτη (*Waterfall Model*)



Παναγιώτης Σφέτσος, Μεθοδολογίες Προγραμματισμού II

3

Εικόνα 2: Μοντέλο Καταρράκτη (*Waterfall Model*)

2.2.2 Το Ελικοειδές Μοντέλο (*Spiral Model*)

Το ελικοειδές Μοντέλο προτάθηκε από τον Boehm το (1988). Το ελικοειδές ή σπειροειδές μοντέλο ,το οποίο και απεικονίζεται μάλιστα κάτωθι στο σχήμα 2, πρόκειται για μία επαναληπτική διαδικασία, η οποία χρήζει αναγνώρισης σε πολυμεσικές και διαδικτυακές εφαρμογές καθώς ακόμη και σε όλα τα συστήματα διάδρασης ανθρώπου και μηχανής (Τζοβάρας, 2007). Το ελικοειδές ή σπειροειδές μοντέλο (*spiral model*) διακρίνεται από τέσσερα ολοκληρωμένα βήματα:

- ✚ Τον καθορισμό των στόχων,
- ✚ Την αναγνώριση καθώς και την αντιμετώπιση ρίσκων,
- ✚ Την ανάπτυξη, τον έλεγχο και την επαλήθευση και
- ✚ Τον σχεδιασμό των επόμενων επαναλήψεων (Αβούρης, 2015).

Κάθε φάση του συγκεκριμένου μοντέλου παρουσιάζεται ως μια σπείρα. Οι σπείρες είναι διαδοχικές. Κάθε φάση είναι και μια μικρογραφία του κύκλου ζωής του λογισμικού. Οι βασικές εργασίες της ανάλυσης και του σχεδιασμού, όπως επίσης και η υλοποίηση επαναλαμβάνονται σε κάθε έναν από τους έλικες του σπινάλ αυτού ξεχωριστά. Έτσι, οι στόχοι επανακαθορίζονται στη συνέχεια προσαρμόζεται και επικυρώνεται ο αντίστοιχος κώδικας και τελικά η φάση της αξιολόγησης αποτελεί τον θεμελιώδη λίθο για τον

σχεδιασμό και τη μετάβαση στον επόμενο κύκλο ανάπτυξης του εν λόγω μοντέλου (Χατζάρας, 2013). Η αναγνώριση των πιθανών ρίσκων αντιμετωπίζεται σε κάθε διακριτό κύκλο του ελικοειδούς ή σπειροειδούς μοντέλου (spiral model). Ειδικότερα, κατά την αναγνώριση αυτή, επιτελείται οργανωμένη προσπάθεια να εντοπισθούν οι βασικότεροι κίνδυνοι. Συγχρόνως, διενεργούνται κινήσεις για την συλλογή πληροφοριών προκειμένου οι κίνδυνοι να αντιμετωπισθούν ταχέως (Χατζάρας, 2013).

Το ελικοειδές ή σπειροειδές μοντέλο (spiral model) είναι κατάλληλο στις αντικειμενοστρωφικές μεθοδολογίες ανάλυσης και σχεδιασμού. Επιπρόσθετα, αξίζει να σημειώσουμε πως το μοντέλο αυτό είναι ιδιαίτερα συμβατό με τα συστήματα, τα οποία αλληλεπιδρούν με τους χρήστες τους (Αβούρης, 2015).



Εικόνα 3: Το Ελικοειδές Μοντέλο (Spiral Model)

2.2.3 Το Αστεροειδές Μοντέλο

Το αστεροειδές μοντέλο των Hix & Hartson (1993) σχεδιάστηκε και αποδίδει μεγάλη σημασία στην ανάπτυξη των διαδραστικών συστημάτων καθώς επίσης και στην εναργέστερη συμμετοχή των χρηστών σε αυτό. Το αστεροειδές μοντέλο, σε αντίθεση με άλλα μοντέλα που προηγήθηκαν (όπως το μοντέλο του καταρράκτη ή και το ελικοειδές ακόμη μοντέλο), δεν περιλαμβάνει με βάση το σχεδιασμό του μία αυστηρή ακολουθία φάσεων. Ως κεντρική δραστηριότητα ορίζεται η αξιολόγηση του συστήματος (Αβούρης, 2015). Αυτό πρακτικά μεταφράζεται ως εξής: κάθε φάση του συγκεκριμένου μοντέλου, είτε πρόκειται για τη φάση του σχεδιασμού, είτε για την υλοποίηση ή για κάποια άλλη πρόσθετη φάση του μοντέλου, οφείλει να προστεθεί ακόμη μία φάση αξιολόγησης, η οποία πραγματοποιείται με τη συμμετοχή είτε χρηστών του αναπτυσσόμενου συστήματος

είτε των ειδικών (Αβούρης, 2015). Επιπρόσθετα, το αστεροειδές μοντέλο, εκτός του ότι δεν επιβάλλει μία αυστηρή ακολουθία φάσεων, όπως ήδη έχουμε επισημάνει, συγχρόνως δεν ορίζει και μονοσήμαντα το σημείο έναρξης της διαδικασίας. Το συγκεκριμένο μοντέλο απαιτεί την εμπλοκή αναλυτικών και συνθετικών μεθόδων του σχεδιασμού και μπορεί να ενσωματώσει στην εξέλιξη της λειτουργίας του τους ίδιους τους χρήστες. Τονίζεται ότι σε όλη το κύκλο ζωής του λογισμικού, σε κάθε φάση συμμετέχουν ενεργά οι χρήστες. (Αβούρης, 2015)



Εικόνα 4: Το Αστεροειδές Μοντέλο

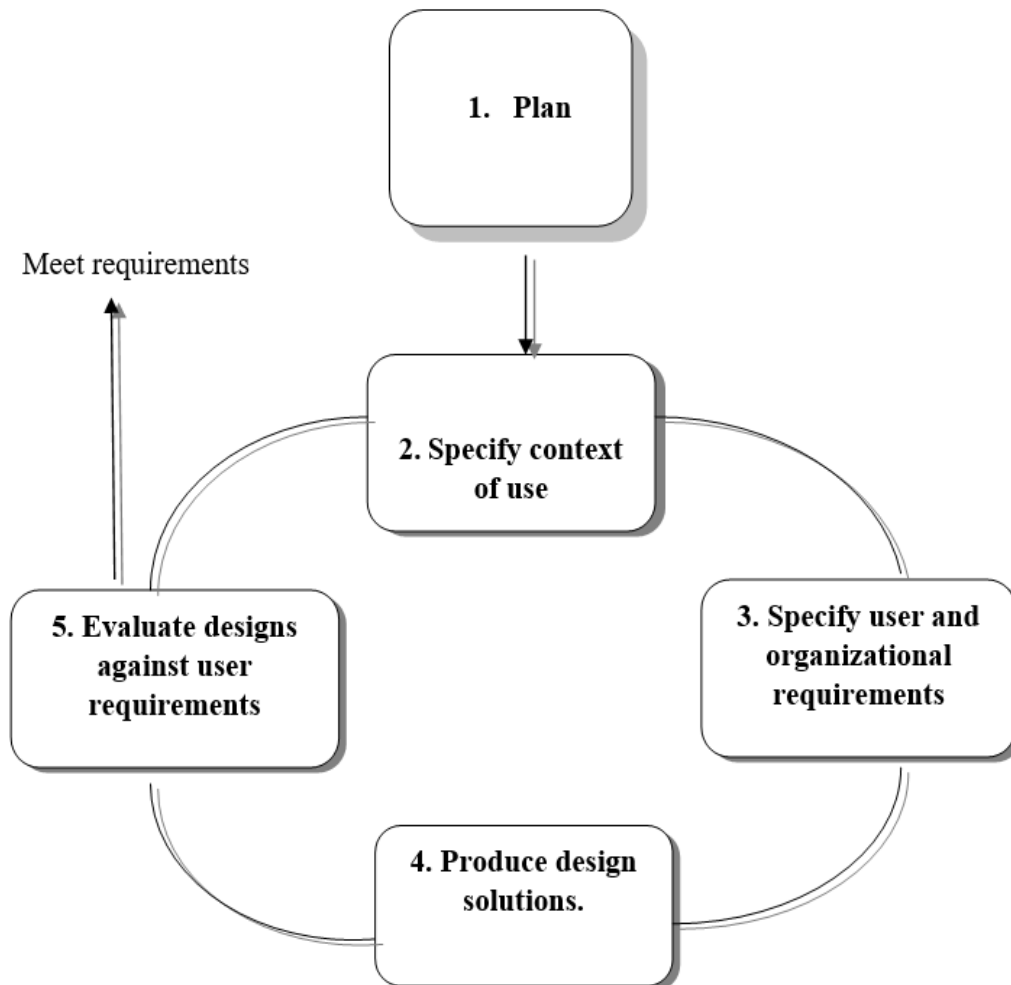
2.2.4 Το μοντέλο του ανθρωποκεντρικού σχεδιασμού κατά ISO9241-210:2010

Η μεθοδολογία αυτή είναι γνωστή και με τους αγγλικούς όρους «Human – Centered Design for Interactive Systems». Βασικός κορμός της συγκεκριμένης μεθοδολογίας είναι οι χρήστες που συμμετέχουν ενεργά στην ανάπτυξη του λογισμικού. Και μάλιστα συμμετέχουν ενεργά από τις απαρχές του λογισμικού. Για αυτό άλλωστε και ονομάζεται ανθρωποκεντρική καθώς όλες οι φάσεις εξαρτώνται από τις αποφάσεις και κυρίως από τις αξιολογήσεις των χρηστών. Με βάση το πρότυπο ISO9241-210:2010, το οποίο για την ακρίβεια αποτελεί μία βελτιωμένη και νεότερη έκδοση του προτύπου ISO 13407:1999, το μοντέλο ανθρωποκεντρικού σχεδιασμού αποτελείται από τις ακόλουθες φάσεις:

- ✚ Ο σχεδιασμός της Ανθρωποκεντρικής Διαδικασίας,
- ✚ Ο προσδιορισμός του πλαισίου χρήσης,
- ✚ Ο προσδιορισμός απαιτήσεων των χρηστών και των οργανωτικών απαιτήσεων,

- ✚ Η παραγωγή σχεδιαστικών λύσεων και
- ✚ Η αξιολόγηση της τελικής σχεδίασης βάση των προβλεπόμενων απαιτήσεων (Αβούρης, 2015).

Η Εικόνα 5 αποδίδει με μεγαλύτερη ευκρίνεια τα βήματα της διαδικασίας με βάση το ανθρωποκεντρικό μοντέλο σχεδιασμού.



Εικόνα 5: Το μοντέλο του ανθρωποκεντρικού σχεδιασμού κατά ISO9241-210:2010

Αξίζει να τονίσουμε στο σημείο αυτό πως το συγκεκριμένο πρότυπο του ανθρωποκεντρικού σχεδιασμού, βασίζεται σε ορισμένες σημαντικές αξίες, οι οποίες

σαφώς και το προσδιορίζουν. Για το λόγο αυτό θα αναφερθούμε εν συντομία σε αυτές τις βασικές αρχές:

- ✚ Το συγκεκριμένο μοντέλο, δίνει έμφαση στη σαφή κατανόηση των χρηστών , των διαφόρων εργασιών καθώς και του περιβάλλοντος χρήσης.
- ✚ Οι χρήστες εμπλέκονται σε όλη τη διαδικασία και τη διάρκεια του σχεδιασμού καθώς ακόμη και στην ανάπτυξη. Η βασική αυτή αρχή υπονοεί πως οι χρήστες οφείλουν να εμπλέκονται σε ολόκληρη την εξέλιξη της διαδικασίας και όχι μονάχα στην αρχή και στο τέλος αυτής. Αυτό πρακτικά σημαίνει πως η ομάδα σχεδιασμού οφείλει να παρουσιάζει τις έτοιμες σχεδιαστικές λύσεις προς τους χρήστες σε όλες τις φάσεις της διαδικασίας.
- ✚ Ο σχεδιασμός, στο συγκεκριμένο μοντέλο, οργανώνεται και σταδιακά βελτιώνεται μέσα από μία ανθρωποκεντρική αξιολόγηση. Η αξιολόγηση αυτή θα πρέπει να γίνεται σε κάθε μέρος της διαδικασίας. Με τον τρόπο αυτό αξιολογούνται τόσο τα αρχικά πρότυπα και οι αρχικές ιδέες της σχεδιαστικής ομάδας όσο και οι προχωρημένες, ολοκληρωμένες εκδόσεις.
- ✚ Η διαδικασία που ακολουθείται χαρακτηρίζεται ως επαναληπτική. Η αρχή αυτή αντικατοπτρίζει την πεποίθηση πως είναι δύσκολο να αποδοθεί από την αρχή η εξήγηση σχετικά με το τι επιθυμούν οι χρήστες από το σύστημα (Χατζάρας, 2013).

3. ΕΠΙΣΤΗΜΟΝΙΚΑ ΑΡΘΡΑ

Η παρούσα εργασία θα επικεντρωθεί στην κατασκευή μίας εφαρμογής διαδικτύου (web application), η οποία θα συγκρίνει επιστημονικά άρθρα διαφορετικών συγγραφέων και θα εντοπίζει τις πιθανές τους συγκλίσεις. Στο συγκεκριμένο σημείο λοιπόν κρίνεται φρόνιμο να γίνει μία αναλυτική παρουσίαση της έννοιας και της δομής του επιστημονικού / ερευνητικού άρθρου.

3.1 Εισαγωγή

Άρθρο ονομάζεται ένα γραπτό κείμενο, το οποίο έχει ορισμένη έκταση. Συγκεκριμένα, πρόκειται για *«μια δημοσιευμένη μικρού μεγέθους, πολύ εξειδικευμένη και πρωτότυπη εργασία που αναφέρεται σε ένα ειδικό θέμα. Οι συγγραφείς ενός άρθρου μπορεί να είναι και περισσότεροι του ενός»*. (Λεύι, 2017). Ένα άρθρο αναφέρεται σε ένα θέμα της επικαιρότητας αποκτώντας με τον τρόπο αυτό ειδησεογραφικό χαρακτήρα. Συγχρόνως ένα άρθρο μπορεί να κάνει λόγο για ένα ειδικό / επιστημονικό θέμα. Το άρθρο είναι εντεταγμένο σε ένα επικοινωνιακό γλωσσικό περιβάλλον και έχει ως απώτερο σκοπό του την πειθώ.

3.2 Επιστημονικό/ Ερευνητικό Άρθρο

Ένα επιστημονικό άρθρο πραγματεύεται θέματα του ευρύτερου επιστημονικού ενδιαφέροντος, τα οποία απασχόλησαν για κάποια δεδομένη χρονική στιγμή την επιστημονική κοινότητα και συνεχίζουν να αποτελούν αντικείμενο διερεύνησης και στο σύγχρονο παρόν. Παρέχει επιστημονικά και δεδομένα και αποτελέσματα, τα οποία προέκυψαν μετά από μία ουσιαστική επιστημονική – ερευνητική προσπάθεια. Το επιστημονικό άρθρο δημοσιεύεται σε περιοδικά επιστημονικού ενδιαφέροντος και είναι δυνατόν να αποτελέσει αντικείμενο μελέτης και συζήτησης σε συνέδρια και σεμινάρια, ανάλογα με το θέμα που πραγματεύονται. Τα επιστημονικά άρθρα εστιάζουν στην ενημέρωση της επιστημονικής κοινότητας για τα πιο σύγχρονα επιτεύγματα που προκύπτουν από τη διεξαγωγή νεότερων ερευνών. Τα επιτεύγματα αυτά είναι στη διάθεση των νέων ερευνητών, οι οποίοι ενδέχεται να τα αξιοποιήσουν ως βάση αφετηρίας και να

διεξάγουν νέες ερευνητικές διαδικασίες. Εδώ αξίζει να αναφερθεί ότι υπάρχουν δύο είδη επιστημονικών άρθρων:

- ✚ Τα πλήρη άρθρα (full articles): αποτελούν το σημαντικότερο είδος ενός επιστημονικού άρθρου. Είναι εκτενή (3.000 έως 5.000 λέξεις) και στηρίζονται στην πραγματοποίηση μίας πρωτότυπης επιστημονικής έρευνας.
- ✚ Τα review papers: το συγκεκριμένο είδος δεν παρέχει στο σύνολο της επιστημονικής κοινότητας νέες πληροφορίες αλλά αναφέρεται σε ήδη γνωστές έρευνες κοινοποιώντας τα αποτελέσματά τους. (Φλούρης, 2003)

3.3 Δομή Επιστημονικού Άρθρου

Τα τελευταία χρόνια έχει υιοθετηθεί, κυρίως από τα επιστημονικά περιοδικά, μια συγκεκριμένη δομή για την συγγραφή και τη δημοσίευση ενός ερευνητικού άρθρου. Η δομή αυτή αποτελείται από τέσσερις κεντρικούς άξονες, πάνω στους οποίους ο εκάστοτε συγγραφέας πρέπει να στηριχτεί.

Οι τέσσερις άξονες είναι:

- Η εισαγωγή (Introduction): στην εισαγωγή ενός επιστημονικού άρθρου, ο συγγραφέας θα πρέπει να απαντήσει στο εξής ερώτημα: «Για ποιο λόγο πραγματοποιήθηκε η συγκεκριμένη έρευνα ή μελέτη;» (Γαλάνης, 2014)
- Οι μέθοδοι (Methods): στις μεθόδους, ο συγγραφέας θα πρέπει να δώσει απάντηση στα εξής δύο ερωτήματα: α) Ποια είναι η μέθοδος που ακολούθησε και β) πώς η μέθοδος αυτή εφαρμόστηκε στη πράξη. Ο συγκεκριμένος άξονας μπορεί να περιλαμβάνει και άλλες υποκατηγορίες έτσι ώστε να γίνει περισσότερο κατανοητή η διαδικασία που ακολούθησε ο συγγραφέας για την έρευνα του. Αυτές μπορεί να είναι οι εξής:
 - Ο σχεδιασμό της μελέτης,
 - Ο μελετώμενος πληθυσμός,
 - Οι μεταβλητές,
 - Το μέγεθος του δείγματος,
 - Η στατιστική Ανάλυση (Γαλάνης, 2014)

- **Αποτελέσματα (Results):** το πιο σημαντικό κομμάτι σε ένα επιστημονικό άρθρο είναι η συγγραφή των αποτελέσματα που επέφερε η ερευνητική διαδικασία. Επομένως, βασικό ερώτημα στο οποίο πρέπει να απαντήσει ο συγγραφέας είναι ποια είναι τα αποτελέσματα του και να προχωρήσει στην αναλυτική παρουσίασή τους με βάση τα μεθοδολογικά εργαλεία που αξιοποίησε.
- **Συζήτηση (Discussion):** η συζήτηση ως τελευταίο στάδιο της συγγραφής ενός επιστημονικού άρθρου περιλαμβάνει την παρουσίαση των αποτελεσμάτων σε συνδυασμό με ανάλογα επιστημονικά ευρήματα. Στο συγκεκριμένο δηλαδή στάδιο πραγματοποιείται μία επισκόπηση των αποτελεσμάτων σε σχέση με το ευρύτερο επιστημονικό πεδίο, στο οποίο αυτά εντάσσονται. (Γαλάνης, 2014)



Εικόνα 6: Βασικά Μέρη ενός Επιστημονικού Άρθρου

(Γαλάνης, 2014)

Οι τέσσερις άξονες αυτοί, όπως προαναφέρθηκε, είναι βασικοί για την συγγραφή ενός άρθρου. Ο κάθε ένας από αυτούς περιλαμβάνει επιπλέον κατηγορίες όπως φαίνεται και στην Εικόνα 7.

-
- Τίτλος (συνήθως απαιτείται και ένας σύντομος τίτλος)
 - Συγγραφείς με τις ιδιότητές τους
 - Συγγραφέας υπεύθυνος για την επικοινωνία
 - Περίληψη
 - Λέξεις-κλειδιά (ευρετηρίου)
 - Εισαγωγή
 - Μέθοδοι
 - Αποτελέσματα (κείμενο, πίνακες και γραφήματα)
 - Συζήτηση
 - Ευχαριστίες, αντικρουόμενα συμφέροντα και οικονομική υποστήριξη
 - Βιβλιογραφικές παραπομπές
-

Εικόνα 7: Επιπλέον κατηγορίες ως μέρη ενός επιστημονικού άρθρου

(Γαλάνης, 2014)

3.4 Αξιολόγηση Επιστημονικού Άρθρου

Τα επιστημονικά άρθρα, όπως προαναφέρθηκε, αποτελούν ένα σημαντικό σταθμό σε κάθε έρευνα και σε κάθε επιστήμονα ή επιστημονική ομάδα, καθώς ενημερώνουν την επιστημονική κοινότητα για τα ευρήματα. Το επιστημονικό άρθρο περνάει μια διαδικασία αξιολόγησης. Η διαδικασία αξιολόγησης ενός επιστημονικού άρθρου περιλαμβάνει κάποια θεμελιώδη ερωτήματα.

- Για ποιο λόγο πραγματοποιήθηκε η έρευνα για την οποία έγινε η συγγραφή του άρθρου;
- Ποιες μέθοδοι και ποια ερευνητικά εργαλεία αξιοποιήθηκαν κατά την επιστημονική έρευνα;
- Ποια ήταν τα ερευνητικά δεδομένα και σε ποια ζητούμενα ήθελα να καταλήξει;

Επίσης κατά τη διαδικασία αξιολόγησης ενός επιστημονικού άρθρου λαμβάνονται υπόψη και άλλα κριτήρια όπως είναι:

- Η επιστημονική Ποιότητα,
- Η περιεκτικότητα και
- Η πρωτοτυπία (Κουτραφούρη Δ.& Κωνσταντινίδης Π & Μυριανθεύς)

Η επιστημονική ποιότητα κριτήριο ,που προαναφέρθηκε ως βασικό στοιχείο για την αξιολόγηση ενός επιστημονικού άρθρου, περιλαμβάνει μεταξύ άλλων την εγκυρότητα και την αξιοπιστία των βιβλιογραφικών παραπομπών. Για τον έλεγχο της εγκυρότητας και της αξιοπιστίας των βιβλιογραφικών παραπομπών υπάρχουν ορισμένοι δείκτες. Οι δείκτες αυτοί διευκολύνουν την αξιολόγηση των αναφορών και οδηγούν κατ' επέκτασιν στην διασφάλιση της εγκυρότητας και της αξιοπιστίας της ερευνητικής διαδικασίας. Οι δείκτες αυτοί παρουσιάζονται αναλυτικά στην ακόλουθη εικόνα :

Citation number	αριθμός αναφορών	δείχνει τον αριθμό των εργασιών στις οποίες αναφέρεται η εργασία του συγγραφέα στη διεθνή βιβλιογραφία μέχρι σήμερα
Impact factor	συνολικός συντελεστής απήχησης	προκύπτει από το άθροισμα των συντελεστών απήχησης των περιοδικών στα οποία έχουν δημοσιευθεί οι εργασίες του συγγραφέα. (Ο συντελεστής απήχησης ενός περιοδικού καθορίζει την ποιότητα του περιοδικού και διαφοροποιείται ετησίως; Πχ., New England Journal of Medicine 51.3, Science 30.0, Cell 29.2, Journal of Clinical Investigation 15.8)
H factor		ο μεγαλύτερος αριθμός δημοσιευμένων εργασιών (H) ενός συγγραφέα, έτσι ώστε, κάθε μια από αυτές τις H-εργασίες έχει τουλάχιστον H-αναφορές στη διεθνή βιβλιογραφία. Η factor 105 σημαίνει ότι καθεμία από τις 105 πρώτες, από άποψη αναφορών, εργασίες, αναφέρεται στη βιβλιογραφία τουλάχιστον 105 φορές. Ο H factor είναι μέτρο της “καλής διασποράς” των αναφορών του συγγραφέα
M factor		ο H factor διαιρούμενος με τα χρόνια που μεσολαβούν από την πρώτη δημοσίευση
G factor		ο μεγαλύτερος αριθμός δημοσιευμένων εργασιών (G) ενός συγγραφέα, έτσι ώστε όλες οι G-εργασίες έχουν τουλάχιστον G ² -αναφορές (αθροιστικά) στη διεθνή βιβλιογραφία. G factor 105 σημαίνει ότι οι 105 πρώτες, από άποψη αναφορών, εργασίες, αναφέρονται στη βιβλιογραφία τουλάχιστον 105x105= 11025 φορές συνολικά. Ο G factor εξαρτάται από δημοσιεύσεις με πολύ μεγάλο αριθμό αναφορών
R factor		διορθώνει για την ύπαρξη πολλαπλών συγγραφέων σε μεγάλες συνεργασιακές μελέτες ή consortia και για την σχετική θέση του συν-συγγραφέως στην δημοσίευση.

Εικόνα 8: Ορισμοί Βιβλιογραφικών Δεικτών

(Χρούσος, 2009)

3.5 Πλατφόρμες Πηγών Επιστημονικών Άρθρων

Το επιστημονικό/ερευνητικό άρθρο αποτελεί εναρκτήριο λάκτισμα για έναν επιστήμονα. Ο ερευνητής/ επιστήμονας χρειάζεται πληροφορίες για να μπορέσει να δομήσει τα ερευνητικά του δεδομένα και να προχωρήσει ακολούθως στην υλοποίηση της ερευνητικής

του προσπάθειας. Το διαδίκτυο παρέχει στους επιστήμονες - ερευνητές τη δυνατότητα να έρθουν σε επαφή με πληθώρα επιστημονικών άρθρων. Τα άρθρα αυτά περιλαμβάνονται σε ηλεκτρονικές πλατφόρμες. Ορισμένες από αυτές τις πλατφόρμες διαθέτουν ελεύθερη είσοδο για κάθε χρήστη, που ενδιαφέρεται να διαβάσει τα επιστημονικά άρθρα. Από την άλλη πλευρά ορισμένες πλατφόρμες προσφέρουν πρόσβαση μόνο σε πιστοποιημένα μέλη της ακαδημαϊκής κοινότητας. Κάποιες μάλιστα διαθέτουν χρηματικό αντίτιμο για την παροχή των επιστημονικών πληροφοριών. Οι πιο διαδεδομένες πλατφόρμες είναι οι εξής:

- Openarchives.gr στην ηλεκτρονική διεύθυνση <http://www.openarchives.gr/> :η συγκεκριμένη πλατφόρμα παρέχει τη δυνατότητα σε κάθε επισκέπτη να έχει πρόσβαση σε επιστημονικά άρθρα και όχι μόνο. Δηλαδή, μπορεί να έχει πρόσβαση σε αρχεία περιοδικών, διπλωματικές εργασίες και στην γκρίζα βιβλιογραφία συνολικά. Συγκεκριμένα έχει αποθηκευμένα 750.226 τεκμήρια από 59 φορείς περιεχομένου.
- Συλλογικός κατάλογος ελληνικών ακαδημαϊκών βιβλιοθηκών στην ηλεκτρονική διεύθυνση <http://www.unioncatalog.gr/ucportal/>.
- Βιβλιοθήκη του Πάντειου Πανεπιστημίου: https://opac.seab.gr/search~S13*gre . Στην ίδια φιλοσοφία κινείται και η πλατφόρμα της ακόλουθης ηλεκτρονικής διεύθυνσης: pandemos.panteion.gr
- Συλλογικός Κατάλογος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών: <http://www.unioncatalog.gr/ucportal/>
- Από τον σύλλογο Elsevier: <https://www.elsevier.com/>
- <https://www.doabooks.org/>. Πρόκειται για πλατφόρμα αναζήτησης εφημερίδων-journal καθώς και επιστημονικών βιβλίων.
- Μια πλατφόρμα, η οποία είναι ιδιαίτερα δημοφιλής στην αναζήτηση επιστημονικών άρθρων, αναφορών και εισηγήσεων σε συνέδρια είναι το Google Scholar.<https://scholar.google.com/>
(Μηχανές αναζήτησης Επιστημονικών Άρθρων)
- Academia Edu. Πρόκειται για μια σχετικά πρόσφατη πλατφόρμα αναζητήσεων ακαδημαϊκών ερευνών και άρθρων. Αξίζει να αναφερθεί ότι η αξιοποίηση της συγκεκριμένης πλατφόρμας είναι διττή. Από την μία πλευρά ο εκάστοτε χρήστης μπορεί να πραγματοποιήσει ελεύθερα την αναζήτηση των επιστημονικών άρθρων που τον ενδιαφέρουν. Έχει τη δυνατότητα μάλιστα να κατεβάσει (download) και

να αποθηκεύσει τα επιστημονικά άρθρα της αρεσκείας του, δίχως να πληρώσει κάποιο αντίτιμο. Από την άλλη πλευρά ο κάθε χρήστης μπορεί να δημιουργήσει έναν premium account στην πλατφόρμα αυτή. Ο λογαριασμός αυτός του επιτρέπει να ανεβάζει και δικά του άρθρα/ papers, στα οποία αντίστοιχα έχουν ελεύθερη πρόσβαση και οι υπόλοιπες χρήστες του διαδικτύου. <https://www.academia.edu/>

- <https://www.sciencedirect.com/> ή αλλιώς Science Direct. Είναι μια πλατφόρμα, την οποία διαχειρίζεται ο όμιλος Elsevier και περιλαμβάνει πάνω από έντεκα εκατομμύρια επιστημονικά άρθρα.
- Εκδόσεις Κάλλιπος. Πρόκειται για μια πλατφόρμα αναζήτησης επιστημονικών άρθρων, βιβλίων και αναφορών ελλήνων συγγραφέων/ επιστημόνων από τα ελληνικά πανεπιστήμια. <https://repository.kallipos.gr/>

Αυτές, λοιπόν είναι ορισμένες από τις πλατφόρμες στις οποίες και θα βασιστεί η παρούσα εργασία για να προχωρήσει στην σύγκριση μεταξύ των επιστημονικών δημοσιεύσεων καθηγητών του τμήματος Πληροφορικής.

4. ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ ΚΑΙ NLP

Η συγκεκριμένη εργασία βασίζεται σε τεχνικές προγραμματισμού καθώς ακόμη και στην αξιοποίηση των γλωσσών προγραμματισμού. Επιπρόσθετα, η παρούσα έρευνα στηρίζεται και στην έννοια της μηχανικής μάθησης, διότι το υπό μελέτη λογισμικό μας θα κάνει συσχέτιση μεταξύ των συγγραφέων και των επιστημονικών δημοσιεύσεών τους.

Για αυτό το λόγο στο παρόν κεφάλαιο, θα γίνει μία προσπάθεια αναφοράς στη μηχανική μάθηση και ειδικότερα σε αυτό που ονομάζουμε NLP, Natural Language Processing καθώς ακόμη και στη μέθοδο «Bags of Words». Αξίζει ωστόσο να αναφερθούμε, έστω και σε ένα επιγραμματικό επίπεδο, στον γενικότερο ορισμό, που δίνεται στη διαδικασία της μάθησης, πριν γίνει εκτενέστερη αναφορά στην αξιοποίησή της από την επιστήμη της πληροφορικής.

4.1 Η Μάθηση

Η μάθηση αποτελεί ένα σύνθετο εσωτερικό, βιολογικό και πνευματικό φαινόμενο, το οποίο μάλιστα έχει μελετηθεί κατά καιρούς από διαφορετικούς μεταξύ τους επιστημονικούς κλάδους, από την ψυχολογία και την παιδαγωγική έως την ιατρική και τη βιολογία. Η ουσία της μάθησης καθώς και οι διαδικασίες, οι οποίες και την αποτελούν, χαρακτηρίζονται ως ποικιλόμορφες. Το γεγονός πως είναι τόσο διαφορετικές επομένως, καθιστά δύσκολη την κατάταξή τους σε μία και μοναδικά αληθή κατηγορία. Όπως εύστοχα επισημαίνει και ο Φλουρής (2003) η μάθηση σαν διαδικασία, παρά την πληθώρα των επιστημονικών ερευνών οι οποίες και διεξάγονται προς την αποσαφήνισή της, παραμένει μία επιστημονική διαδικασία, η οποία δεν έχει κατανοηθεί και δεν έχει ερμηνευτεί πλήρως κατά τρόπο παραδεκτό από ολόκληρη την επιστημονική κοινότητα, που την ερευνά.

Η μάθηση έχει οριστεί από τον Pavlov (1927) ως η διαδικασία δημιουργίας αντανακλαστικών. Ακολούθως διαφορετικοί επιστήμονες όπως ο Thorndike (1932) έκανε λόγο για τη μάθηση ως δοκιμή και πλάνη ενώ ο Skinner (1933) όρισε την μάθηση ως την επανάληψη μίας αντίδρασης μετά από ορισμένη θετική ενίσχυση. Ο Kohler (1997) ορίζει τη μάθηση ως ενόραση και ο Bandura (1977) ως μίμηση προτύπων. Ο Gagne (1975)

αναφέρει χαρακτηριστικά πως η μάθηση είναι μία διαδικασία, η οποία υποβοηθά τους διάφορους οργανισμούς να τροποποιήσουν τη συμπεριφορά τους σε ένα σύντομο χρονικό διάστημα, με έναν μόνο τρόπο, έχοντας ως αποτέλεσμα η τροποποίηση αυτή ή η αλλαγή που επέρχεται να μην συμβαίνει καθ' επανάληψη σε κάθε μία νέα περίπτωση. Κατά την εξέλιξη των ετών και με βάση τις νέες επιστημονικές μελέτες, οι οποίες διεξάγονται συνεχώς για τη μάθηση και τις ιδιότητές της, διαπιστώνουμε πως δημιουργούνται νέοι ορισμοί και νέες σχολές, αναφορικά με τα σημαντικότερα στοιχεία της. Συνοψίζοντας, η μάθηση σε μία γενικότερη προσπάθεια ορισμού της, μπορούμε να αναφέρουμε πως πρόκειται για ένα φαινόμενο, το οποίο περιλαμβάνει διαδικασίες τόσο στο βιολογικό όσο και στο πνευματικό επίπεδο των εμπλεκομένων σε αυτή.

4.2 Μηχανική Μάθηση

Η Μηχανική Μάθηση (Machine Learning) αποτελεί ξεχωριστό πεδίο της Πληροφορικής, της επιστήμης που σχετίζεται με τους υπολογιστές. Η Μηχανική Μάθηση (Machine Learning) αναπτύχθηκε μέσα από την μελέτη της ανάπτυξης προτύπων και της υπολογιστικής θεωρίας μάθησης στην Τεχνητή νοημοσύνη. Η Μηχανική Μάθηση (Machine Learning) ασχολείται με την μελέτη και την κατασκευή αλγόριθμων, οι οποίοι μπορούν να μαθαίνουν από τα δεδομένα και να πραγματοποιούν προβλέψεις με αυτά. Έχουν διατυπωθεί αρκετοί ορισμοί κατά καιρούς για τη Μηχανική Μάθηση, οι οποίοι επιχειρούν να αναλύσουν τη διαδικασία καθώς ακόμη και τον τρόπο με τον οποίο πραγματοποιείται η μάθηση μέσα σε ένα σύστημα

«Μηχανική Μάθηση είναι το φαινόμενο κατά το οποίο ένα σύστημα βελτιώνει την απόδοση του κατά την εκτέλεση μιας συγκεκριμένης εργασίας, χωρίς να υπάρχει η ανάγκη να προγραμματιστεί εκ νέου» (Γεωργούλη, 2015). Ο ορισμός αυτός επισημαίνει ότι η μηχανική μάθηση εκπαιδεύει μηχανές με απώτερο σκοπό οι μηχανές να «μάθουν» να αλλάζουν και να βελτιώνονται χωρίς όμως να χρειάζεται να παρέμβει ο προγραμματιστής για την εκ νέου εισαγωγή εντολών που να συμβαδίζουν με τις παραμέτρους του εκάστοτε γεγονότος.

Ένας άλλος ορισμός που δίνεται από τον Carbonell(1987) σημειώνει πως: *«Μηχανική Μάθηση είναι η μελέτη υπολογιστικών μεθόδων για την απόκτηση νέας γνώσης, νέων*

δεξιοτήτων και νέων τρόπων οργάνωσης της υπάρχουσας γνώσης». Ο Mitchell (1997) ορίζει πως «Ένα πρόγραμμα υπολογιστή λέγεται ότι μαθαίνει από εμπειρία E ως προς μια κλάση εργασιών T και ένα μέτρο επίδοσης P , αν η επίδοσή του σε εργασίες της κλάσης T , όπως αποτιμάται από το μέτρο P , βελτιώνεται με την εμπειρία E ». Ο ορισμός είναι ιδιαίτερα σημαντικός για τον καθορισμό της Μηχανικής Μάθησης σε βασικό λειτουργικό πλαίσιο παρά με γνωστικούς όρους. Συγχρόνως ο ορισμός αυτός ενισχύει την πεποίθηση πως κατά την εξέλιξη του επιστημονικού κλάδου της Πληροφορικής η ερώτηση αν τα μηχανήματα μπορούν να σκεφτούν, είναι εφικτό να αντικατασταθεί από το ερώτημα αν μπορούν οι μηχανές να κάνουν αυτό που οι άνθρωποι, ως σκεπτόμενα όντα, μπορούν να κάνουν.

Εν κατακλείδι, όλοι οι παραπάνω ορισμοί, ουσιαστικά, παρουσιάζουν τη μηχανική μάθηση έως ένα σύνολο από μεθόδους που σκοπό έχουν να εκπαιδεύσουν το σύστημα να είναι ικανό να μεταβάλλεται για να ανταποκρίνεται στις μεταβολές του περιβάλλοντός του. Ακόμα και η παραμικρή μεταβολή αποτελεί δείγμα μάθησης αφού το σύστημα «αντιλήφθηκε» την ανάγκη της αλλαγής.

4.2.1 Είδη Μηχανικής Μάθησης

Η Μηχανική Μάθηση, όπως προαναφέρθηκε, είναι η διαδικασία εκπαίδευσης ενός συστήματος. Η εκπαίδευση αυτή διαιρείται σε τρεις μεγάλες κατηγορίες ανάλογα με τη φύση του εκπαιδευτικού «σήματος» ή την πιθανή «ανατροφοδότηση» σε ένα σύστημα εκμάθησης. Οι κατηγορίες αυτές είναι:

- Επιβλεπόμενη Μάθηση (Supervised Learning): με βάση αυτό το είδος της μάθησης, ο αλγόριθμος κατασκευάζει μια συνάρτηση βασιζόμενος στα δεδομένα και στα ήδη γνωστά αποτελέσματα. Σκοπός αυτού του είδους είναι η γενίκευση αυτής της συνάρτησης για την περίπτωση εισόδου αλλά με μη γνωστή έξοδο. Αυτή η μάθηση χρησιμοποιείται συνήθως σε προβλήματα: α) ταξινόμησης (classification), β) πρόγνωσης (prediction), γ) διερμηνείας (interpretation).
- Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning): εδώ ο αλγόριθμος δεν γνωρίζει εκ των προτέρων τις εξόδους. Χρησιμοποιείται σε προβλήματα που απαιτούν α) ανάλυση Συσχετισμών (Association Analysis) και β) ομαδοποίηση (clustering).

- Ενισχυτική Μάθηση: εδώ ο αλγόριθμος μαθαίνει μέσα από αλληλεπιδράσεις με το περιβάλλον και η κατηγορία αυτή χρησιμοποιείται σε προβλήματα Σχεδιασμού (Planning), (Γεωργούλη, 2015)

4.2.2 Αλγόριθμοι Μηχανικής Μάθησης

Για κάθε μία από τις παραπάνω κατηγορίες μάθησης, έχουν αναπτυχθεί και οι αντίστοιχοι αλγόριθμοι, οι οποίοι αποτελούν σημείο αναφοράς. Οι αλγόριθμοι αυτοί χρησιμοποιούνται ευρέως όχι μόνο στον τομέα της Πληροφορικής. Τα τελευταία χρόνια βρίσκουν χώρο εφαρμογής και σε άλλους, διαφορετικούς επιστημονικούς κλάδους όπως είναι η Ιατρική. Ορισμένοι από τους ευρέως διαδεδομένους αλγόριθμους είναι:

- Τα Δέντρα Αποφάσεων,
- Ο Αλγόριθμος ID3,
- Ο Αλγόριθμος C4.5,
- Ο Αλγόριθμος Bayes.

4.3 Ο όρος NLP

Η βασική ιδέα για τον σχεδιασμό του λογισμικού με βάση το οποίο οργανώθηκε και η παρούσα έρευνα, συνίσταται στην προσπάθεια αναγνώρισης των επιστημονικών κειμένων και την ανάθεσή τους σε ορισμένους επιστήμονες – συγγραφείς. Για να πραγματοποιηθεί η ενέργεια αυτή, αξιοποιήσαμε κατά τον σχεδιασμό του λογισμικού μας το σύστημα NLP σε συνδυασμό με αλγόριθμους της Μηχανικής Μάθησης. Το σύστημα NLP ή διαφορετικά Natural Language Processing είναι μια υπολογιστική μέθοδος για την ανάλυση κειμένων. Η ανάλυση κειμένου βασίζεται σε μια πληθώρα τεχνικών και θεωριών που θα αναλυθούν στις επόμενες παραγράφους.

4.3.1 Ο Ορισμός του NLP

«Η NLP (Natural Language Processing) είναι μια πληθώρα υπολογιστικών μεθόδων και διαδικασιών που στοχεύουν στην ανάλυση και στην αναπαράσταση φυσικών κειμένων σε ένα ή περισσότερα επίπεδα γλωσσικής ανάλυσης με απώτερο σκοπό τη χρήση τους στο

πλαίσιο του ανθρώπινου τύπου επεξεργασίας γλωσσών σε ένα εύρος εφαρμογών»(Liddy,2001). Για να γίνει πιο κατανοητός ο ορισμός αυτός, οφείλουμε να επεξηγήσουμε ορισμένα κομβικά του σημεία. Σε ένα πρώτο επίπεδο αξίζει να αναφερθούμε στα φυσικά κείμενα. Η έννοια αυτή περιλαμβάνει κάθε είδους κείμενο δίχως να υπάρχουν περιορισμοί αναφορικά με την γλώσσα στην οποία είναι γραμμένο ή με την μορφή με την οποία και παρουσιάζεται γραπτά σε εμάς. Ο μοναδικός περιορισμός είναι ο εξής: το κείμενο, το οποίο και θα επεξεργαστούμε πρέπει να είναι γραμμένο σε μία γλώσσα, η οποία χρησιμοποιείται από τους ανθρώπους. Επιπρόσθετα ο ορισμός αυτός αναφέρει ξεκάθαρα τη σύνδεση της NLP με την υπολογιστική νοημοσύνη (AL – Artificial Intelligence).

4.3.2 NLP – Ιστορική Εξέλιξη

Η NLP δεν αποτελεί μία νέα καινοτομία στο τομέα της πληροφορικής. Αντιθέτως, είναι μια προσπάθεια η οποία ξεκίνησε αρκετές δεκαετίες νωρίτερα και συγκεκριμένα περί τα τέλη της δεκαετίας του '40. Η αρχική προσπάθεια ανάλυσης της φυσικής γλώσσας ήταν γνωστή ως Machine Translation (MT). Συγκεκριμένα, το 1946 οι Weaver και Booth ξεκίνησαν τις πρώτες προσπάθειες βασιζόμενοι στην MT.

Κατά τη δεκαετία του 50' και ειδικότερα το 1957, ο Chomsky με το βιβλίο του Syntactic Structures βοήθησε στην εξέλιξη της MT καθώς κατέδειξε τον τρόπο με τον οποίο η επιστήμη της γλωσσολογίας θα μπορούσε να συμβάλλει στην ανάπτυξή της. Την ίδια χρονική περίοδο, νέες περιοχές εφαρμογών της NLP αναδύονταν όπως η αναγνώριση του λόγου. (Liddy, 2001)

Ο επόμενος σημαντικός σταθμός της ιστορικής διαδρομής είναι το 1965. Τη χρονιά αυτή ο Chomsky εισήγαγε ένα μοντέλο γνωστό ως το «μετασχηματιστικό μοντέλο γλωσσικής ικανότητας». Τη δεκαετία του '70 , οι Mann και Thompson διατύπωσαν τη δική τους θεωρία γνωστή ως «Rhetorical Structure Theory» αποδίδει ιεραρχική δομή στο λόγο.

Η μεγαλύτερη όμως εξέλιξη πραγματοποιήθηκε κατά τα τελευταία χρόνια. Η αλματώδης ανάπτυξη της τεχνολογίας και της επιστήμης της Πληροφορικής συνέβαλε αποφασιστικά στην εξέλιξη αυτή. Οι υπολογιστές στην σύγχρονη εποχή διαθέτουν μεγάλη χωρητικότητα και γρήγορη ταχύτητα στην επεξεργασία των δεδομένων. Συγχρόνως υπάρχει μία πληθώρα ηλεκτρονικών διαθέσιμων κειμένων, τα οποία και επιτρέπουν στο σύστημα να

μπορεί να εκπαιδεύεται σε σύντομο χρονικό διάστημα. Τα στοιχεία αυτά οδήγησαν στην δημιουργία NLP συστημάτων που είναι σε θέση να αναγνωρίσουν κείμενα και να αντιμετωπίσουν με μεγάλη άνεση την διαφορετικότητα και την ασάφεια της ανθρώπινης γλώσσας (Liddy, 2001)

4.3.3 NLP – Στάδια

Η NLP είναι ουσιαστικά η προσπάθεια των υπολογιστικών συστημάτων να καταλάβουν έννοιες, εκφράσεις, λέξεις της φυσικής γλώσσας των ανθρώπων. Συνδυάζει την υπολογιστική νοημοσύνη καθώς επίσης και τις αρχές της γλωσσολογικής επιστήμης. Ο εντοπισμός μιας λέξης ή μίας έκφρασης από το σύστημα είναι ένα συνονθύλευμα από κανόνες, οι οποίοι οργανώνονται σε στάδια. Τα στάδια που ακολουθούνται είναι στο συνολικά πέντε:

- Η Μορφολογική και λεκτική ανάλυση (Morphological and Lexical Analysis): το πρώτο στάδιο είναι η μορφολογική και λεκτική ανάλυση, η οποία στοχεύει στον εντοπισμό, την ανάλυση, την αναγνώριση και την περιγραφή της δομής των λέξεων ή των εκφράσεων που συμπεριλαμβάνονται σε ένα κείμενο (Chopra, 2013).
- Η Συντακτική Ανάλυση (Syntactic Analysis): σε αυτό το στάδιο το σύστημα αναγνωρίζει και διαχωρίζει το κείμενο στα επιμέρους δομικά του σημεία δηλαδή σε παραγράφους, προτάσεις και λέξεις (Chopra, 2013).
- Η Σημασιολογική Ανάλυση (Semantic Analysis): αυτό το στάδιο, το τρίτο κατά σειρά, αναλύει τις προτάσεις και παρουσιάζει τη δομή τους. Σε αυτό το στάδιο το σύστημα μπορεί να απορρίψει μια πρόταση εάν αυτή δεν πληροί τα δομικά και συντακτικά κριτήρια σχηματισμού των προτάσεων. Ένα παράδειγμα πρότασης που δεν θα γίνει κατανοητή από ένα τέτοιο σύστημα είναι η ακόλουθη: «Το κορίτσι το πάει για χωριό» κατά την οποία δεν έχει εφαρμοστεί η ακολουθία Υποκείμενο – Ρήμα – Κατηγορημα και Λοιποί προσδιορισμοί, ακολουθία με βάση την οποία δομείται κάθε πρόταση (Chopra, 2013).
- Η Ολοκλήρωση ομιλίας (Discourse Integration): το σύστημα σε αυτό το τέταρτο κατά σειρά στάδιο εξετάζει τη συσχέτιση ή αλλιώς το νόημα μεταξύ των διαφορετικών δομών μίας πρότασης ή μίας παραγράφου. Ελέγχει δηλαδή το νόημα μεταξύ αυτών των δομών, καθώς πρέπει να είναι συναφές και να μην υποπίπτει σε αντίθετες μεταξύ τους έννοιες. Χαρακτηριστικό παράδειγμα αποτελεί η έκφραση «άχρωμη μπλε

θάλασσα» καθώς η λέξη άχρωμη (= δίχως να υπάρχει κάποιο χρώμα) αναιρεί και «αφαιρεί» την λέξη μπλε, η οποία δίνει ξεκάθαρα την έννοια ενός χρώματος (Chopra, 2013).

- Η Ρεαλιστική ανάλυση (Pragmatic Analysis): το στάδιο αυτό αφορά την απεικόνιση του σκοπού της πρότασης. Κάθε πρόταση δημιουργείται με σκοπό να επιτελέσει μία συγκεκριμένη λειτουργία. Η λειτουργία αυτή αναδεικνύεται στο συγκεκριμένο στάδιο.

4.3.4 NLP & Μηχανική Μάθηση

Η μηχανική μάθηση αποτελεί το βασικό στοιχείο για την προώθηση της τεχνητής νοημοσύνης. Ένα στοιχείο, το οποίο αναπτύσσεται στο χρονικό παρόν και αποτελεί ακόμη έναν μελλοντικό τομέα δράσης των επιστημόνων που ασχολούνται με τον τομέα της Πληροφορικής. Επομένως, είναι εύλογο τα NLP προβλήματα και οι εφαρμογές να βασίζονται και να χρησιμοποιούν τη μάθηση για την εκπαίδευσή τους. Η μηχανική μάθηση, όπως χρησιμοποιείται στα NLP προβλήματα, βασίζεται σε αλγόριθμους γενικής μάθησης. Ο αλγόριθμος αυτός εκπαιδεύεται μέσω την αφομοίωσης των γλωσσικών κανόνων από μια τεράστια πηγή παραδειγμάτων.

Αρκετοί αλγόριθμοι μηχανικής μάθησης έχουν δοκιμαστεί σε NLP όπως είναι τα δέντρα αποφάσεων ή οι κανόνες «if-else». Παρά την ευρεία χρήση τους όμως δεν αποδείχτηκαν ιδιαίτερα παραγωγικοί. Τα τελευταία χρόνια λοιπόν καθιερώθηκε ένας διαφορετικός τρόπος προσέγγισης, ο οποίος συνδυάζει δυναμικά την NLP και την Μηχανική Μάθηση. Ο τρόπος αυτός στηρίζεται σε στατιστικά μοντέλα, τα οποία δημιουργούνται και σχεδιάζονται σε αποφάσεις με βάση τις πιθανότητες και τις ειδικές τιμές, τα λεγόμενα βάρη, που εφαρμόζονται σε κάθε είσοδο στο προς εκπαίδευση σύστημα (Chopra, 2013).

5. Ο ΣΧΕΔΙΑΣΜΟΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

5.1 Εισαγωγή

Το παρόν κεφάλαιο σχετίζεται με τον σχεδιασμό της διαδικτυακής μας εφαρμογής. Πιο συγκεκριμένα, η παρούσα πτυχιακή εργασία πραγματοποιήθηκε με γνώμονα τη δημιουργία μίας εφαρμογής, η οποία και θα διέθετε ως βασική της λειτουργία τη σύγκριση των επιστημονικών προφίλ ερευνητών – καθηγητών, με βάση τα δημοσιευμένα επιστημονικά τους άρθρα. Θα μπορούσαμε ακόμη να προσθέσουμε στο σημείο αυτό πως μέσω της σύγκρισης, την οποία και επιτελεί η εφαρμογή μας, παράγονται στο τέλος γενικά στατιστικά δεδομένα αναφορικά με τα ποσοστά εγγύτητας των άνωθεν επιστημονικών άρθρων, των ερευνητών – καθηγητών, οι οποίοι και συμπεριλαμβάνονται στη διαδικτυακή εφαρμογή.

5.2 Απαιτήσεις λογισμικού

Οι απαιτήσεις ενός νέου λογισμικού, είναι μία ιδιαίτερα δύσκολη διαδικασία. Συγχρόνως, αποτελεί και ακρογωνιαίο λίθο για την θεμελίωση και την μετέπειτα λειτουργία κάθε νέου λογισμικού. Στο σημείο αυτό, αξίζει να σημειωθεί ότι συχνά καταγράφονται πολλά προβλήματα κατά την υλοποίηση μίας νέας διαδικτυακής εφαρμογής, προβλήματα τα οποία έχουν ως αφετηρία τα ελλείψεις που παρουσιάζονται αναφορικά με τις προδιαγραφές και τις απαιτήσεις κάθε νέου παραγόμενου προϊόντος. Οι απαιτήσεις λοιπόν, θα μπορούσαμε να ισχυριστούμε ότι ορίζονται ως συγκεκριμένες προδιαγραφές για το τι θα πρέπει να εφαρμοστεί. Κατά τρόπο επαγωγικό, οι απαιτήσεις οδηγούν τις επιλογές του σχεδιασμού. Με βάση τις βιβλιογραφικές αναφορές υπάρχουν τρία μοντέλα καταγραφής απαιτήσεων ενός νέου λογισμικού: 1) οι λειτουργικές και μη λειτουργικές απαιτήσεις λογισμικού, 2) οι περιπτώσεις χρήσης (use cases) και 3) οι ιστορίες χρηστών (user stories) (Anon, 2007).

5.2.1 Περιπτώσεις χρήσης (use cases)

Για την εκπόνηση της παρούσας πτυχιακής εργασίας αξιοποιήθηκαν, στο πλαίσιο του καθορισμού των απαιτήσεων, οι περιπτώσεις χρήσης – use cases. Οι περιπτώσεις χρήσης, αποτελούν ένα σύνολο σεναρίων, τα οποία συνδέονται μεταξύ τους με έναν κοινό για τον

χρήστη σκοπό (Μπεληγιάννης, 2015). Το σύνολο αυτό περιλαμβάνει το πρωταρχικό σενάριο, που διαθέτει μία ακολουθία βημάτων, καθώς και εναλλακτικές περιπτώσεις (Μπεληγιάννης, 2015).

Η χρησιμοποίηση του συγκεκριμένου μοντέλου προσφέρει αρκετά θετικά πλεονεκτήματα καθώς οι περιπτώσεις χρήσης:

- ✚ Βοηθούν στον καθορισμό των λειτουργικών απαιτήσεων του συστήματος,
- ✚ Δίνουν μία ακριβέστερη εικόνα για το τι πρέπει να εκτελεί το σύστημα,
- ✚ Διαμορφώνουν μία βάση αναφοράς για τον έλεγχο του συστήματος,
- ✚ Διευκολύνουν τον εντοπισμό λειτουργικών απαιτήσεων μέσα στις κλάσεις του συστήματος (Μπεληγιάννης, 2015).

Στη συνέχεια, θα αναφερθούμε στις περιπτώσεις χρήσης – use cases του λογισμικού μας έτσι ώστε να προχωρήσουμε μετέπειτα στις λειτουργίες που πρέπει να ολοκληρώσουμε προγραμματιστικά.

Περίπτωση χρήσης «Σύνδεση διαχειριστή»

Δράστης – actor : Διαχειριστής

Αρχική Συνθήκη: Η κατάσταση του διαχειριστή είναι «ασύνδετος»

Βασική ροή:

- ✓ 1α. Ο πελάτης ζητά να συνδεθεί.
 - 1β. Το λογισμικό ζητά από τον διαχειριστή να δώσει τα απαραίτητα δεδομένα σύνδεσής του.
- ✓ 2α. Ο διαχειριστής δίνει τα δεδομένα για τη σύνδεσή του.
 - 2β. Το λογισμικό προβαίνει σε έλεγχο των στοιχείων του διαχειριστή και επιβεβαιώνει την εγκυρότητά τους.
- ✓ Ένα τα στοιχεία κριθούν ως έγκυρα, τότε γίνεται η σύνδεση.

Τελική συνθήκη μετά τη βασική ροή: η κατάσταση του διαχειριστή είναι τώρα «συνδεδεμένος».

Εναλλακτικές ροές: εάν τα δεδομένα σύνδεσης δεν εμφανιστούν ως έγκυρα, τότε εμφανίσε μήνυμα «λανθασμένα στοιχεία σύνδεσης παρακαλώ προσπαθήστε ξανά». Με τον τρόπο αυτό η ροή επιστρέφει ξανά στο 1 α βήμα. Συγχρόνως στο βήμα 1 α ο διαχειριστής επιλέγει να ακυρώσει τη διαδικασία σύνδεσης. Το λογισμικό ακυρώνει τη λειτουργία σύνδεσης και επιστρέφει στις προηγούμενες λειτουργίες.

Περίπτωση χρήσης «Εγγραφή καθηγητή στη βάση δεδομένων»

Δράστης – Actor: Διαχειριστής.

Αρχική συνθήκη: Κανένας καθηγητής στη βάση δεδομένων.

Βασική ροή:

- ✓ 1 α. Ο διαχειριστής ζητά να εισάγει ένα καθηγητή στη βάση δεδομένων.
 - 1β. το λογισμικό στην περίπτωση αυτή ζητά από τον διαχειριστή να εισάγει τα στοιχεία ενός καθηγητή.
- ✓ 2 α. Ο διαχειριστής εισάγει τα στοιχεία ενός καθηγητή.
 - 2β1. Το λογισμικό ελέγχει εάν έχουν συμπληρωθεί όλα τα απαραίτητα στοιχεία.
 - 2β2. Εάν όλο τα στοιχεία έχουν συμπληρωθεί το λογισμικό πραγματοποιεί καταχώρηση, εισαγωγή του καθηγητή στη βάση δεδομένων.

Τελική συνθήκη μετά τη βασική ροή: Τα στοιχεία του εκάστοτε καθηγητή έχουν καταχωρηθεί επιτυχώς στη βάση δεδομένων.

Εναλλακτικές ροές: στο βήμα 1β σε περίπτωση που κάποιο προσωπικό στοιχείο του καθηγητή δεν έχει συμπληρωθεί σωστά εμφανίσε μήνυμα «Δώσε όλα τα απαιτούμενα προσωπικά στοιχεία». Η ροή τότε επιστρέφει στο βήμα 1 α.

Στο βήμα 1β ο διαχειριστής επιλέγει να ακυρώσει τη διαδικασία εγγραφής του καθηγητή στη βάση δεδομένων. Το λογισμικό τότε ακυρώνει τη λειτουργία και η ροή επιστρέφει στην προηγούμενη λειτουργία.

Περίπτωση χρήσης «Τροποποίηση καθηγητή»

Δράστης – Actor: Διαχειριστής.

Αρχική συνθήκη: Ο διαχειριστής είναι συνδεδεμένος.

Βασική ροή:

- ✓ 1 α. Ο διαχειριστής ζητά να αλλάξει τα στοιχεία ενός συγκεκριμένου καθηγητή από τον κατάλογο των καθηγητών – ερευνητών .
 - 1 β. Το λογισμικό στην περίπτωση αυτή ζητά από τον διαχειριστή να τροποποιήσει τα στοιχεία ενός συγκεκριμένου καθηγητή από τον κατάλογο των καθηγητών – ερευνητών.
- ✓ 2 α. Ο διαχειριστής τροποποιεί τα στοιχεία του συγκεκριμένου καθηγητή.
 - 2 β. Το λογισμικό ελέγχει εάν έχουν συμπληρωθεί όλα τα απαραίτητα στοιχεία για τον συγκεκριμένο καθηγητή.
 - 2 β2. Εάν τα στοιχεία έχουν συμπληρωθεί ορθά το λογισμικό καταχωρεί τα νέα τροποποιημένα στοιχεία του συγκεκριμένου καθηγητή στην βάση δεδομένων.

Τελική συνθήκη μετά τη βασική ροή: Τα στοιχεία του συγκεκριμένου καθηγητή έχουν τροποποιηθεί.

Εναλλακτικές ροές: Στο βήμα 1β σε περίπτωση που κάποιο στοιχείο δεν έχει συμπληρωθεί σωστά η ροή επιστρέφει στο βήμα 1 α. Στο βήμα 1β ο διαχειριστής επιλέγει να ακυρώσει τη διαδικασία. Το λογισμικό επιστρέφει στην προηγούμενη λειτουργία.

Περίπτωση χρήσης « Διαγραφή καθηγητή »

Δράστης – Actor: Διαχειριστής.

Αρχική συνθήκη: Ο διαχειριστής είναι συνδεδεμένος.

Βασική ροή:

- ✓ 1 α. Ο διαχειριστής ζητά να διαγράψει τα στοιχεία ενός συγκεκριμένου καθηγητή από τη βάση δεδομένων.
 - 1β. Το λογισμικό στην περίπτωση αυτή ζητά από τον διαχειριστή να επιβεβαιώσει εάν επιθυμεί να προβεί σε ολοκλήρωση της διαγραφής των στοιχείων ενός συγκεκριμένου καθηγητή από την βάση δεδομένων.
- ✓ 2 α. Ο διαχειριστής επιβεβαιώνει τη διαγραφή.
 - 2β. Το λογισμικό διαγράφει τον καθηγητή από τη βάση δεδομένων.

Εναλλακτικές ροές: Στο βήμα 1β ο διαχειριστής ακυρώνει την ενέργεια. Το λογισμικό επιστρέφει στην προηγούμενη λειτουργία του.

Τελική συνθήκη μετά τη βασική ροή: Τα στοιχεία του συγκεκριμένου καθηγητή έχουν διαγραφεί επιτυχώς από τη βάση δεδομένων.

Περίπτωση χρήσης « Αναζήτηση άρθρων»

Δράστης – actor: Διαχειριστής

Αρχική συνθήκη: ο διαχειριστής είναι συνδεδεμένος.

Βασική ροή:

- ✓ 1 α. από τον πίνακα των άρθρων ο διαχειριστής είναι σε θέση να πραγματοποιήσει την αναζήτησή του με βάση το εξής κριτήριο: αναζήτηση με κριτήριο ενός String , όπου String μπορεί να είναι το όνομα, επώνυμο, ονοματεπώνυμο, website και άλλα.

1β. Το λογισμικό εμφανίζει τον κατάλογο με τα άρθρα, τα οποία πληρούν τα κριτήρια αναζήτησης που χρησιμοποιήθηκαν από τον διαχειριστή.

Εναλλακτική ροή: στο βήμα 1β είναι εφικτό να μην εντοπιστούν άρθρα, τα οποία να είναι σύμφωνα με τα κριτήρια αναζήτησης. Σε αυτή την περίπτωση το λογισμικό εμφανίζει το ανάλογο μήνυμα.

Περίπτωση χρήσης « Υπολογισμός εγγύτητας μεταξύ των επιστημονικών δημοσιεύσεων των καθηγητών – ερευνητών»

Δράστης – Actor: Διαχειριστής.

Αρχική συνθήκη: Ο διαχειριστής είναι συνδεδεμένος.

Βασική ροή:

- ✓ 1 α. Μέσα από τον πίνακα των καθηγητών – ερευνητών ο διαχειριστής ζητά να υπολογιστεί η εγγύτητα των δημοσιευμένων άρθρων ενός συγκεκριμένου καθηγητή με τις δημοσιεύσεις των υπόλοιπων καθηγητών, που υπάρχουν στη βάση δεδομένων.

1β. Το λογισμικό εμφανίζει έναν πίνακα, ο οποίος περιλαμβάνει σε ποσοστά την ομοιότητα των δημοσιευμένων άρθρων του συγκεκριμένου καθηγητή που

αιτηθήκαμε στο βήμα 1 α σε σύγκριση με τους υπόλοιπους καθηγητές, οι οποίοι περιλαμβάνονται στη βάση δεδομένων. Συγχρόνως το λογισμικό παρουσιάζει τα ποσοστά αυτά σε διαγραμματική μορφή.

Τελική συνθήκη μετά τη βασική ροή: Ο διαχειριστής βλέπει στην οθόνη τα ποσοστά εγγύτητας που ζητήθηκαν στο βήμα 1 α.

5.3 Τα εργαλεία ανάπτυξης λογισμικού

Κατά την συγγραφή του παρόντος κεφαλαίου αναφερθήκαμε σε σημαντικά στοιχεία, τα οποία καθόρισαν τον σχεδιασμό της διαδικτυακής μας εφαρμογής. Αξίζει επομένως να αναφερθούμε με τρόπο αναλυτικό και στα εργαλεία ανάπτυξης των λογισμικών, τα οποία χρησιμοποιήθηκαν για την συγκεκριμένη έρευνα. Τα εργαλεία λοιπόν που αξιοποιήθηκαν ήταν: η γλώσσα προγραμματισμού Python καθώς και η γλώσσα Structured Query Language (SQL).

5.3.1 Η γλώσσα προγραμματισμού Python

Οι γλώσσες προγραμματισμού στο σύνολό τους εξυπηρετούν τις άμεσες ανάγκες ενός προγραμματιστή και επαγγελματία της επιστήμης της Πληροφορικής. Επιπρόσθετα, οι γλώσσες προγραμματισμού είναι εφικτό να χρησιμοποιηθούν και από εξειδικευμένους χρήστες των υπολογιστών. Οι γλώσσες προγραμματισμού επιτρέπουν στον εκάστοτε χρήστη να επικοινωνήσει με το σύστημα υπολογιστών καθώς υπαγορεύουν τις εντολές που το σύστημα καλείται να εκτελέσει.

Η «συνομιλία» ενός χρήστη με τον υπολογιστή είναι δύσκολη. Ο ηλεκτρονικός υπολογιστής αποτελείται από μία πληθώρα ηλεκτρικών κυκλωμάτων και καλωδίων. Ο υπολογιστής δεν αντιλαμβάνεται την γλωσσική νόρμα των ανθρώπων. Αντίθετα, κατανοεί μονάχα τα διάφορα ηλεκτρικά σήματα που λαμβάνει ή καταλαβαίνει την γλώσσα μηχανής, η οποία αποτελείται από συμβολοσειρές από 0 και 1, δηλαδή από μία ακολουθία από 0 και 1. Πρόκειται για το δυαδικό σύστημα που αξιοποιεί ο υπολογιστής. Ο προγραμματισμός με την χρησιμοποίηση της γλώσσας μηχανής απαιτεί πολύ καλή γνώση της λειτουργίας ενός υπολογιστικού συστήματος καθώς ακόμη και άριστη γνώση και χειρισμό της συγκεκριμένης γλώσσας. Η γλώσσα μηχανής είναι ιδιαίτερα απαιτητική και

πολλές φορές χαρακτηρίζεται ως δυσνόητη από έναν ανθρώπινο νου. Αποτελούσε επομένως τροχοπέδη για έναν επαγγελματία.

Επιτακτική ήταν η ανάγκη να δοθεί μία λύση σε αυτό το πρόβλημα που δυσχέραινε περισσότερο την επικοινωνία ενός χρήστη με τον υπολογιστή. Η λύση στο συγκεκριμένο πρόβλημα δόθηκε με τη δημιουργία μίας συμβολικής γλώσσας, η οποία προσπάθησε να αντικαταστήσει τις ακολουθίες των ψηφίων 0 και 1 με ορισμένες αγγλικές λέξεις που συμβόλιζαν την λειτουργία τους. Η συμβολική γλώσσα (assembly language) αποδείχθηκε χρονοβόρα και χαρακτηρίστηκε ως χαμηλού επιπέδου γλώσσα προγραμματισμού. Η τελευταία και η πιο επιτυχημένη λύση δόθηκε με τη δημιουργία των γλωσσών προγραμματισμού.

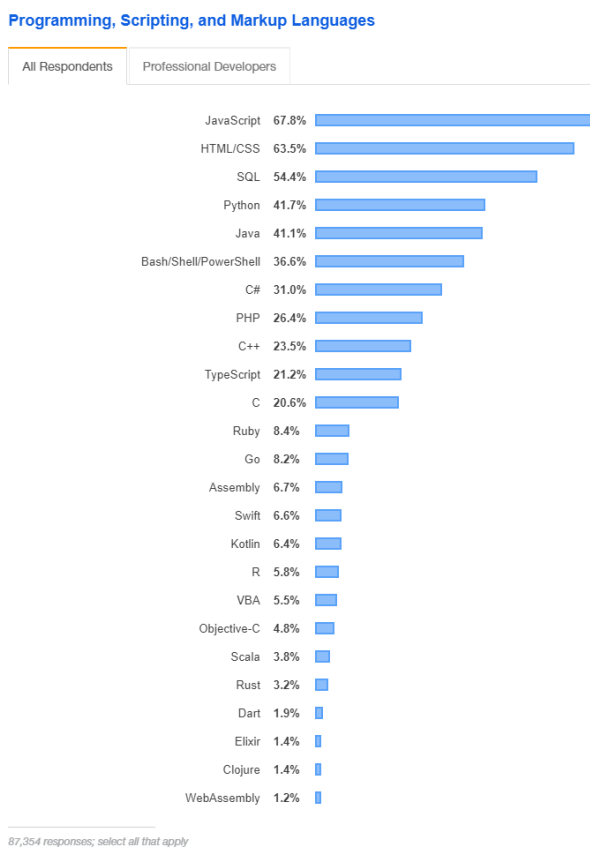
Οι γλώσσες προγραμματισμού είναι τεχνητές γλώσσες, οι οποίες είναι δυνατόν να αξιοποιηθούν για τον έλεγχο ενός υπολογιστή. Αποτελούνται από ένα συγκεκριμένο εννοιολογικό και συντακτικό δομικό σύστημα. Το σύστημα αυτό των κανόνων ορίζει τη δομή και την σημασία των προτάσεων μίας γλώσσας προγραμματισμού. Σε αυτές επομένως, ο εκάστοτε προγραμματιστής έχει τη δυνατότητα συγγραφής ενός προγράμματος χρησιμοποιώντας αγγλικές λέξεις με συγκεκριμένη σύνταξη. Με τον τρόπο αυτό το πρόγραμμα είναι εύκολο να διορθωθεί και να επιτελεστεί με τον πιο αποτελεσματικό τρόπο. Αξίζει να σημειώσουμε το εξής: τα προγράμματα που έχουν δημιουργηθεί με μία γλώσσα προγραμματισμού, για να γίνουν εκτελέσιμα και να μπορέσουν να πραγματοποιηθούν πρέπει να «μεταφραστούν». Η διαδικασία αυτή πραγματοποιείται με την βοήθεια ενός μεταφραστή (compiler).

Οι γλώσσες προγραμματισμού είναι πολλές σε πλήθος. Κάποιες είναι παρωχημένες, κάποιες «ήρθαν για να μείνουν» και κάποιες αποτελούν το μέλλον και την εξέλιξη των προηγούμενων γενιών. Στην τελευταία κατηγορία ανήκει και η γλώσσα προγραμματισμού Python, η οποία ανήκει στη τελευταία γενιά γλωσσών προγραμματισμού.

Η Python είναι μια γλώσσα η οποία χαρακτηρίζεται από ευκολία στη σύνταξη και στη γραμματική των εντολών της καθώς επίσης και από την υποστήριξη λειτουργιών που ανάγονται σε παλαιότερες γλώσσες όπως στη C καθώς και σε αντικειμενοστραφείς γλώσσες όπως C++. Η Python μάλιστα ήταν αυτή που πρωτοτύπησε όσο αφορά τις αναβαθμίσεις προς τα πίσω.

Η Python έχει αρκετά πλεονέκτημα. Ένα εξ αυτών είναι και ότι δεν απαιτείται η δήλωση των μεταβλητών στο κυρίως πρόγραμμα. Επίσης, δεν απαιτείται η χρήση αγκίστρων για την καθιέρωση των ορίων κάθε block κώδικα που μπορεί να ανήκει σε μια συνάρτηση ή στο κυρίως πρόγραμμα. Τέλος, όπως προαναφέρθηκε, στηρίζει αρκετές λειτουργίες παλαιότερων γλωσσών προγραμματισμού όπως είναι η δυναμική δέσμευση μνήμης καθώς επίσης και λειτουργίες των αντικειμενοστραφών γλωσσών όπως η ύπαρξη κλάσης, κληρονομικότητας καθώς και λεξικά και λίστες, οι οποίες δίνουν στον προγραμματιστή μια ευελιξία στην αποθήκευση και διαχείριση των δεδομένων.

Τέλος, η Python μπορεί εύκολα να χρησιμοποιηθεί σε συνδυασμό με άλλες γλώσσες όπως η SQL. Δεν είναι τυχαίο, λοιπόν, που αποτελεί μία από τις πρώτες προτιμήσεις των προγραμματιστών.



Εικόνα 9: Δημοφιλείς Γλώσσες προγραμματισμού

{Πηγή: <https://insights.stackoverflow.com/survey/2019#technology>}

```

>>> class Math:
    def __init__(self, x, y):
        self.x = x
        self.y = y
    def add(self):
        return self.x + self.y
    def subtract(self):
        return self.x - self.y

>>> class Math2:
    def __init__(self, x, y):
        self.x = x
        self.y = y
    def multiply(self):
        return self.x * self.y
    def divide(self):
        return self.x / self.y

>>>

```

Εικόνα 10: Python & Classes

{Πηγή: <https://blog.eduonix.com/software-development/learn-object-oriented-programming-in-python-composition/>}

5.3.2 Η γλώσσα SQL

Η γλώσσα SQL είναι γνωστή για τη χρήση της σε βάσεις δεδομένων. Οι εντολές της είναι έτσι δομημένες ώστε ο διαχειριστής τις βάσεις να μπορεί να ελέγχει και να εξάγει τα δεδομένα από αυτές.

Η ιστορία της SQL ξεκινάει το 1970 στην IBM. Οι δημιουργοί Andrew Richardson, Donald C. Messerly και Raymond F. Boyce είχαν σκοπό να καταλήξουν σε μια γλώσσα προγραμματισμού έτσι ώστε να είναι ικανή να διαχειριστεί στοιχεία αποθηκευμένα σε RDBMS – Relational DataBase Management Systems ή σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων. Αυτός ήταν ο σκοπός της SQL ή όπως ήταν αρχικά γνωστή SEQUEL. («Γλώσσες Τέταρτης Γενιάς», 2008)

Η SQL παρέχει, λοιπόν, δυνατότητες διαχείρισης βάσεων. Βασικά της στοιχεία είναι:

- Expression
- Clauses

- Queries («Γλώσσες Τέταρτης Γενιάς», 2008)

Η εξαγωγή συγκεκριμένων δεδομένων από τη βάση γίνονται με τη χρήση των πρόσθετων ερωτημάτων. Καθένα από αυτά βασίζεται στην εντολή select με την ακόλουθη σύνταξη:

```
select                <λίστα                πεδίων>
from                  <λίστα_πινάκων>
where <συνθήκες>
```

Βέβαια, υπάρχουν και παραλλαγές της άνωθεν διαδικασίας όπως :

```
select                <λίστα                πεδίων>
from                  <λίστα_πινάκων>
group                by                πεδίο
having                συνθήκη
order by <λίστα_πεδίων>
```

Η διαχείριση των παραπάνω εντολών προκειμένου να αποδώθουν τα επιθυμητά αποτελέσματα, απαιτεί εμπειρία για αυτό και διδάσκεται σε προπτυχιακό επίπεδο. Μετέπειτα, σε μεταπτυχιακό πια επίπεδο ο φοιτητής εξειδικεύεται σε βάθος στη συγκεκριμένη γλώσσα σε συνδυασμό μάλιστα και με άλλες εφαρμογές και προγράμματα.

5.3.3 Frameworks

Η δουλειά του προγραμματιστή δεν είναι τόσο απλή όσο φαντάζει σε κάποιους. Πρέπει να κατασκευάσει ένα πρόγραμμα, σωστό συντακτικά με βάση τη γλώσσα που θα αξιοποιήσει, ενώ ταυτόχρονα θα πρέπει το πρόγραμμα αυτό να επιλύει το πρόγραμμα. Αυτό σημαίνει ότι πρέπει να συνδυαστούν δομές δεδομένων και άλλα στοιχεία της γλώσσας προγραμματισμού που απαιτούν την προσοχή του προγραμματιστή και την επαγρύπνησή του.

Στις ταυτόχρονες δράσεις του προγραμματιστή έρχεται το software framework. Πρόκειται για μια πλατφόρμα που περιέχει βασικά στοιχεία, τα οποία έχουν υλοποιηθεί από άλλους προγραμματιστές και προσφέρουν μία άτυπη διευκόλυνση θα μπορούσαμε να πούμε για την μετέπειτα δημιουργία και λειτουργία μίας νέας εφαρμογής (Σκορδαλάκης, 2003). Πρακτικά, με την αξιοποίηση των frameworks ο προγραμματιστής δεν χρειάζεται να ανακαλύπτει κάθε φορά τον ακρογωνιαίο λίθο στον προγραμματισμό μίας νέας εφαρμογής. Τα frameworks παρέχουν τη δυνατότητα σε κάθε επαγγελματία

προγραμματιστή να χρησιμοποιεί ξανά κομμάτια κώδικα (Σκορδαλάκης, 2003). Συνήθως, τα frameworks προσφέρουν συντάκτη, μεταγλωττιστή/διερμηνευτή αλλά μπορεί και να παράγουν εξόδους που προσομοιώνουν τη λειτουργία του προγράμματος. Αξίζει στη σημείο αυτό να διευκρινίσουμε τα εξής: μία διαδικτυακή εφαρμογή ανάλογη με το λογισμικό που αναπτύσσουμε στη συγκεκριμένη πτυχιακή εργασία, απαιτεί μία βάση δεδομένων, φόρμες, session και αυθεντικοποίηση. Για τους λόγους αυτούς επιλέξαμε να εντάξουμε τα frameworks στα εργαλεία ανάπτυξης του λογισμικού μας.

5.3.3.1 Τα πλεονεκτήματα των frameworks

Τα πλεονεκτήματα των frameworks είναι πολλά για έναν προγραμματιστή. Στη συνέχεια θα προσπαθήσουμε να παραθέσουμε τα σημαντικότερα από αυτά:

- ✚ Συνεισφέρει στην αποτελεσματικότερη συγγραφή προγράμματος,
- ✚ Παράγεται κώδικας πιο ασφαλής,
- ✚ Παράγεται κώδικας με λιγότερα σφάλματα (bugs),
- ✚ Περιλαμβάνει όχι μόνο βιβλιοθήκες αλλά και σε αρκετές περιπτώσεις προ εγκατεστημένα κομμάτια κώδικα για συγκεκριμένο σκοπό, έτοιμα και ελεγμένα ως προς την αποδοτικότητα τους,
- ✚ Η εκσφαλμάτωση του κώδικα και ο έλεγχος λειτουργίας γίνεται πιο γρήγορος, λιγότερα επώδυνος και αυτό άμεσα έχει αποτέλεσμα και ένα πρόγραμμα πιο πιστό στη λύση που καλέστηκε να επιλύσει,
- ✚ Αυξάνει την παραγωγικότητα,
- ✚ Βελτιώνει την ποιότητα, την αξιοπιστία και την ορθή λειτουργία του κώδικα ενώ τέλος,
- ✚ Το framework είναι δυνατόν να αναβαθμιστεί άμεσα με απώτερο σκοπό την εισαγωγή νέων λειτουργιών, οι οποίες χρησιμοποιούνται άμεσα από τον προγραμματιστή (Singh, 2020).

5.3.3.2 Τα μειονεκτήματα των Frameworks

Η μέχρι τώρα αναφορά μας σχετικά με τα εργαλεία και ειδικότερα με τη χρήση των frameworks επεσήμανε τα θετικά πλεονεκτήματα, που είναι δυνατόν να προσφέρει το framework σε έναν προγραμματιστή αναφορικά με την υλοποίηση μίας διαδικτυακής εφαρμογής. Για να δημιουργήσουμε μία ολοκληρωμένη αναφορά, οφείλουμε να σημειώσουμε και ορισμένα από τα αρνητικά τους χαρακτηριστικά, τα οποία είναι δυνατόν να δυσχεραίνουν το έργο μας. Τα μειονεκτήματα λοιπόν είναι:

- ✚ Τα frameworks είναι χρονοβόρα σχετικά με την εκτέλεση του πηγαίου κώδικα – ειδικότερα στο χρόνο εκτέλεσής του – λόγω του γεγονότος πως το ίδιο το framework εκτελεί ορισμένες διεργασίες,
- ✚ Τα frameworks έχουν ορισμένους κανόνες και ο εκάστοτε προγραμματιστής καλείται να ακολουθήσει κατά γράμμα αυτούς τους κανόνες μη παρεκκλίνοντας – στοιχείο, το οποίο είναι ιδιαίτερα περιοριστικό (Ιωαννίδης , Λέπουρας, 2005).

5.3.3.3 Η λειτουργία

Τα περισσότερα frameworks στην πλειοψηφία τους χρησιμοποιούν ένα συγκεκριμένο μοντέλο αρχιτεκτονικής λογισμικού, το οποίο ονομάζεται MVC Model. Εν συνεχεία, το flask framework χρησιμοποιεί αντίστοιχα ένα παρεμφερές μοντέλο το MVT – Model View Template. Το μοντέλο MVT αποτελείται από τρία διακριτά μεταξύ τους μέλη:

- ✚ Κλάσεις για το μοντέλο – Model της βάσης,
- ✚ Κλάσεις για το View όπου είναι η λογική,
- ✚ Κλάσεις για το Template της εμφάνισης (Wieger, 2002).

5.3.3.4 Model

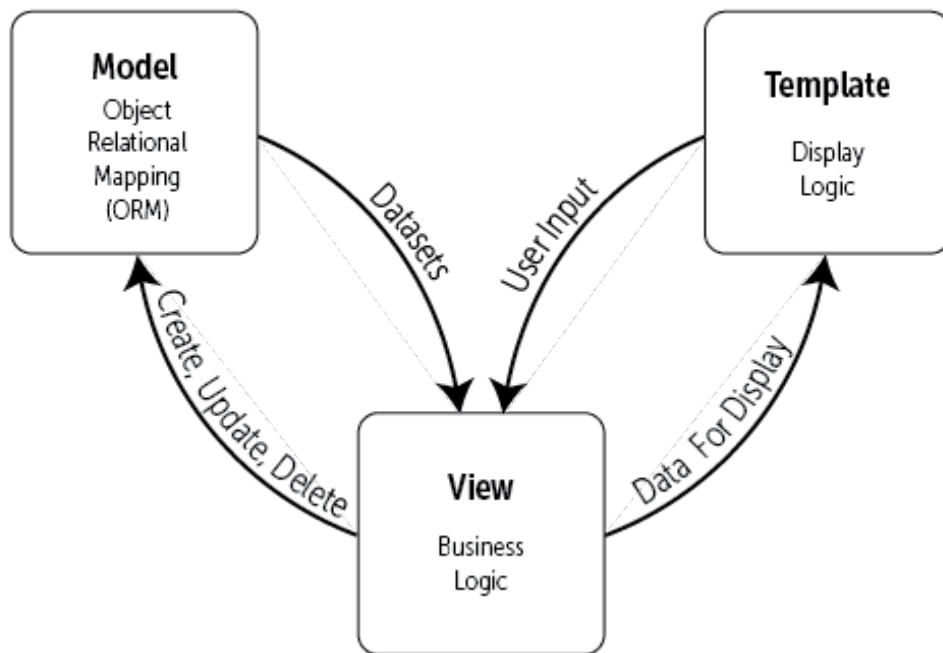
Κάθε διακριτό τμήμα του MVT model αξίζει να παρουσιαστεί στο συγκεκριμένο σημείο αυτής της πτυχιακής εργασίας, έστω και αν αυτό πραγματοποιείται περιληπτικά. Στο επίπεδο model λοιπόν κατατάσσουμε τις κλάσεις που σχετίζονται με τη βάση δεδομένων. Επαγωγικά λοιπόν διαθέτουμε τόσες κλάσεις όσοι και οι πίνακες που υπάρχουν στη βάση δεδομένων μας (Maguire & Bevan, 2002).

5.3.3.5 View

Συνεχίζοντας τις αναφορές μας το view αποτελεί εκείνο το επίπεδο, όπου υπάρχουν όλες οι κινήσεις καθώς και οι συμπεριφορές της εφαρμογής. Αποτελεί με άλλα λόγια τον εγκέφαλο, το βασικό νευραλγικό σημείο της εφαρμογής μας. Με βάση το url που δίνουμε καλούμε την κατάλληλη μέθοδο και με τον τρόπο αυτό μπορούμε εύκολα να πάρουμε, να στείλουμε ή ακόμη και να διαγράψουμε δεδομένα (Maguire & Bevan, 2002).

5.3.3.6 Template

Στο επίπεδο template έχουμε συγκεντρωμένα τα HTML αρχεία, όπου μέσα τους υπάρχει και κώδικας σε γλώσσα προγραμματισμού Python. Αυτό επιτυγχάνεται με τη χρήση ενός template engine – συγκεκριμένα με το Jinja T. E (Bahill & Henderson, 2004).



Εικόνα 11: Λειτουργία MVT Model

5.3.4 HTML

Η HTML (HyperText Markup Language) είναι μια γλώσσα σήμανσης και είναι η κύρια γλώσσα για τις ιστοσελίδες. Περιέχει τα βασικά δομικά στοιχεία των ιστοσελίδων που αποτελούνται από ετικέτες (tags). Οι ετικέτες λειτουργούν σε ζεύγη και περικλείονται μέσα στα σύμβολα <, > και ανάμεσά τους το όνομα της ετικέτας, για παράδειγμα <body> και </body>. Το σύμβολο «/» στην ετικέτα συμβολίζει το τέλος της. Ο Web browser με την βοήθεια της HTML χρησιμοποιεί τις ετικέτες αυτές για να μας παρουσιάσει το περιεχόμενό τους. Το περιεχόμενο αυτό μπορεί να είναι κείμενο, εικόνες, πίνακες, σύμβολα και άλλα.

(<https://el.wikipedia.org/wiki/HTML>, n.d.)

5.3.5 CSS

Η CSS (Cascading Style Sheets) είναι μία γλώσσα που χρησιμοποιείται για την εμφάνιση ενός αρχείου HTML και ανήκει στην κατηγορία των γλωσσών φύλλων ύφους. Η χρήση της γίνεται συνήθως για την μορφοποίηση και βελτίωση εμφάνισης ενός αρχείου HTML. Η γλώσσα αυτή κρίνεται απαραίτητη αν θέλουμε μια μοντέρνα και καλοσχεδιασμένη ιστοσελίδα μιας και αυτή οφείλεται στην διαμόρφωση των χαρακτηριστικών όπως είναι τα χρώματα, οι γραμματοσειρές, η στοίχιση και άλλα.

(<https://el.wikipedia.org/wiki/CSS>, n.d.)

5.3.6 JavaScript

Η JavaScript αποτελεί μία διερμηνευμένη γλώσσα προγραμματισμού για τους ηλεκτρονικούς υπολογιστές. Είναι ουσιαστικά μία γλώσσα σεναρίων ενώ έχει χαρακτηριστεί ιδιαίτερα δυναμική. Η συγκεκριμένη γλώσσα προγραμματισμού είναι γνωστό πώς έχει τυποποιηθεί με βάση τη γλωσσική προδιαγραφή της ECMAScript.

Η σύνταξή της αποτελεί λοιπόν ένα προϊόν διαμόρφωσης με επιρροές από τις πιο σημαντικές γλώσσες προγραμματισμού στην ιστορία της C. Αξίζει να σημειώσουμε πως η Java Script αξιοποιήθηκε, με ιδιαίτερα υψηλά ποσοστά επιτυχίας, από τη μεριά του πελάτη – client side. Σε μία προσπάθεια να ερμηνεύσουμε τι αποτελεί το client side οφείλουμε να πούμε ότι αναφέρεται στην διαδικασία της επεξεργασίας, η οποία γίνεται στον φυλλομετρητή μας και όχι στον server, που φιλοξενείται ο ιστότοπος.

Ακολούθως θα ήταν σκόπιμο να επισημάνουμε το σκοπό της συγκεκριμένης γλώσσας προγραμματισμού: ο σκοπός λοιπόν έγκειται στην εκτέλεση κώδικα σε ιστοσελίδες καθώς και στον χειρισμό των περιεχομένων μιας ιστοσελίδας με απώτερο σκοπό να παρέχεται η δυνατότητα αλληλεπίδρασης των χρηστών με τον ιστότοπο. Η διαδικασία αυτή έχει ως αποτέλεσμα ο server να μας παρέχει το περιεχόμενο του εκάστοτε ιστότοπου και στην συνέχεια εμείς να μπορούμε να επεξεργαζόμαστε, να διαφοροποιούμε και να ανακτούμε δεδομένα χωρίς να απασχολούμε τον server με διάφορα αιτήματά μας. Σε αυτή τη διαδικασία σημαντικό ρόλο παίζουν φυσικά και τα scripts, τα οποία είναι γραμμένα και ανεβασμένα σε κάθε ιστότοπο, τον οποίο καλούμαστε να χρησιμοποιούμε.

Συμπερασματικά μπορούμε ακόμη να σημειώσουμε πως η JavaScript είναι μία γλώσσα προγραμματισμού, που αξιοποιείται και σε εφαρμογές, που δεν σχετίζονται με το διαδίκτυο, όπως είναι τα έγγραφα PDF ή κάποιοι εξειδικευμένοι φυλλομετρητές. Τέλος η Java Script μαζί με την HTML και τη CSS θεωρούνται ως οι τρεις βασικότερες τεχνολογίες για την παραγωγή διαδικτυακών τόπων.

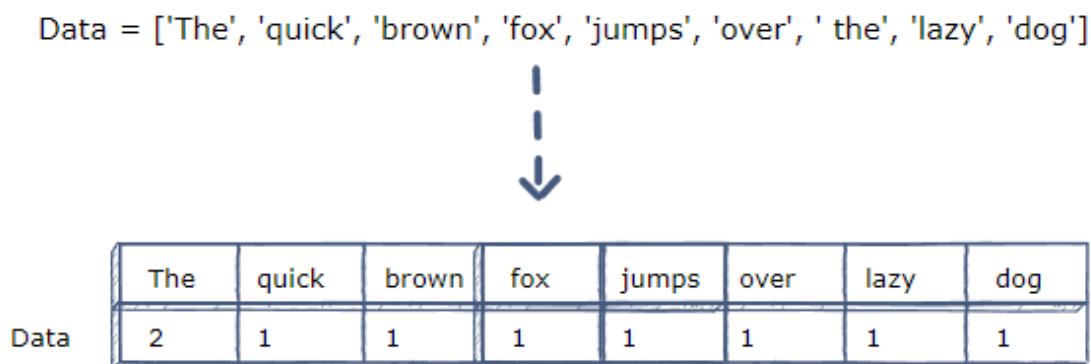
(<https://el.wikipedia.org/wiki/JavaScript>, n.d)

6. Η βιβλιοθήκη Scikit-learn

Η scikit-learn είναι μία δωρεάν βιβλιοθήκη μηχανικής μάθησης για την γλώσσα προγραμματισμού Python. Προσφέρει διάφορα εργαλεία για επεξεργασία και προεπεξεργασία δεδομένων, αξιολόγηση μοντέλου καθώς επίσης και αρκετούς αλγόριθμους για διαδικασίες όπως η ταξινόμηση, η παλινδρόμηση, η ομαδοποίηση δεδομένων. Μερικοί από αυτούς είναι οι k-means, DBSCAN, random forests και άλλοι. Επίσης έχει σχεδιαστεί για να λειτουργεί με κάποιες από τις πιο γνωστές βιβλιοθήκες της γλώσσας Python που είναι οι NumPy και SciPy.

6.1 Bag of words

Το Bag of words είναι μια τεχνική μοντελοποίησης κειμένου της φυσικής γλώσσας. Η διαδικασία έγκειται σε αυτό που περιγράφει η ίδια η φράση. Πρόκειται δηλαδή για μια «τσάντα» από λέξεις ενός κειμένου. Σε αυτό το κείμενο το ενδιαφέρον μας στρέφεται στον αριθμό των λέξεων. Ειδικότερα, επιθυμούμε να διακρίνουμε πόσες φορές περιέχεται μία ορισμένη μονάδα λέξης στο υπό μελέτη κείμενο ενώ αντίθετα δεν επικεντρώνουμε την προσοχή μας στη θέση της συγκεκριμένης λέξης στη συνολική έκταση του κειμένου. Αυτό γίνεται διότι τα γραπτά κείμενα θεωρούνται ακατάστατα και αδόμητα - αναφορικά με τη σύνταξή τους - ενώ οι αλγόριθμοι της μηχανικής μάθησης προτιμούν δομημένα κείμενα. Επίσης τα μοντέλα μηχανικής μάθησης λειτουργούν με αριθμητικά δεδομένα και όχι με δεδομένα κειμένων, έτσι χρησιμοποιώντας το bag of words μετατρέπουμε το κείμενο σε ένα διάνυσμα και το μεταβλητό του μήκος σε σταθερό.



Εικόνα 12: Bag of words – Παράδειγμα

Προκειμένου να επιτύχουμε με ορθό τρόπο την λειτουργία του bag of words, θα αξιοποιήσουμε το CountVectorizer. Καθώς το CountVectorizer αποτελεί στην ουσία του μια μέθοδο κατά την οποία μπορούμε να χρησιμοποιήσουμε διαφορετικές παραμέτρους, αξίζει να σημειώσουμε πως για την δημιουργία του συγκεκριμένου μοντέλου, λάβαμε ως

παράμετρο όλες τις αγγλικές λέξεις. Ωστόσο, η μέθοδος αυτή, είναι δυνατόν να περιοριστεί με την αξιοποίηση οποιασδήποτε παραμέτρου, σε συνάρτηση πάντα με τις απαιτήσεις για την δημιουργία ενός μοντέλου – για παράδειγμα ο προγραμματιστής μπορεί να χρησιμοποιήσει κάθε φυσική γλώσσα είτε να θέσει ορισμένο εύρος λέξεων.

Βασικός μας στόχος είναι να διανυσματοποιήσουμε τα κείμενα για να μπορέσουμε στην συνέχεια να τα συγκρίνουμε. Προκειμένου όμως να επιτύχουμε τον στόχο μας αυτό, απαιτείται μία μικρή προεργασία. Ο προγραμματιστής οφείλει να αφαιρέσει από τα κείμενα όλα τα σημεία στίξης καθώς ακόμη και τα άρθρα, που εμφανίζονται σε αυτά, για να επιτύχει την λειτουργία του bag of words.

```
# Remove '.', ',', ' caps and new line character(\n) from first paper
abst_one = abst_one.lower()
abst_one = re.sub(r'\W', ' ', abst_one)
abst_one = re.sub(r'\s+', ' ', abst_one)

abst_two = abst_two.lower()
abst_two = re.sub(r'\W', ' ', abst_two)
abst_two = re.sub(r'\s+', ' ', abst_two)

# insert to a list the Abstract of papers
paperlist = [abst_one, abst_two]

# Vectorizing the abstracts and make the bag of words from abstracts
vectorizer = CountVectorizer(stop_words='english')
bag_of_words = vectorizer.fit_transform(paperlist)

# Convert Bag of words to Pandas Dataframe to see the word frequencies.
doc_term_matrix = bag_of_words.todense()
df_docs = pd.DataFrame(doc_term_matrix,
                        columns=vectorizer.get_feature_names(),
                        index=['Paper one', 'Paper two'])
```

Εικόνα 13: Κώδικας Bag of words

Η άνωθεν εικόνα παρουσιάζει μια εξελικτική διαδικασία, η οποία αποτελεί μέρος του κώδικα μας. Πιο συγκεκριμένα, φαίνεται πως έχουμε πάρει δυο περιλήψεις άρθρων που θέλουμε να συγκρίνουμε. Σε πρώτο βήμα αφαιρέσαμε τα σημεία στίξης, τα άρθρα και μετατρέψαμε τα κεφαλαία γράμματα σε μικρά. Έτσι πραγματοποιήσαμε την διανυσματοποίηση των κειμένων αυτών με βάση το ισχύον bag of words.

6.2 Cosine Similarity

Το cosine similarity – ημιτονοειδής ομοιότητα συνίσταται σε μια μέτρηση, η οποία είναι δυνατόν να χρησιμοποιηθεί για να υπολογίσει ομοιότητες μεταξύ εγγράφων, ανεξάρτητα από το μέγεθός τους. Μιλώντας με μαθηματικούς όρους το cosine similarity υπολογίζει το συνημίτονο γωνίας μεταξύ δυο διανυσμάτων, τα οποία εμφανίζονται σε έναν πολυδιάστατο χώρο. Η μέτρηση αυτή παρουσιάζει αρκετά πλεονεκτήματα. Ακόμη και αν τα δύο υπό σύγκριση έγγραφα απέχουν από την ευκλείδεια απόσταση, δηλαδή έχουν διαφορετικό μέγεθος, οι πιθανότητες για την μεταξύ τους σύγκριση και ομοιότητα είναι αρκετά μεγάλες.

```
# Compute Cosine Similarity of two abstract text
sims = cosine_similarity(df_docs)
sim = float(sims[0][1]) * 100
```

Εικόνα 14: Κώδικας cosine similarity

Στην συνέχεια θα επιχειρήσουμε να εξηγήσουμε την χρήση του cosine similarity στα πλαίσια του δικού μας κώδικα. Κατά την προηγούμενη διαδικασία με το μοντέλο του Bag of Words δημιουργήσαμε δυο διανύσματα. Σε αυτό το στάδιο εισάγουμε ένα πλαίσιο δεδομένων - Data Frame στην μέθοδο του cosine similarity. Πρακτικά, το πλαίσιο δεδομένων - Data Frame περιέχει τα δύο διανύσματα. Η διαδικασία αυτή επιστρέφει την ημιτονοειδή ομοιότητα των δύο κειμένων.

7.Συμπεράσματα – Περιορισμοί της έρευνας – Μελλοντικές Προεκτάσεις

Καθώς ολοκληρώνουμε σταδιακά τη συγγραφή της παρούσας πτυχιακής εργασίας θα ήταν σκόπιμο και αναγκαίο να αναφερθούμε στα συμπεράσματα που εξάγαμε κατά τη διάρκεια του σχεδιασμού και της υλοποίησης της διαδικτυακής μας εφαρμογής. Αρχικά, οφείλουμε να επισημάνουμε την αναγκαιότητα χρησιμοποίησης των ειδικών εργαλείων. Ιδιαίτερη μνεία πρέπει να δώσουμε στην αξιοποίηση του flask framework. Η χρήση του βοήθησε στο να μειωθεί σημαντικά ο χρόνος ανάπτυξης μίας ανάλογου τύπου εφαρμογής διότι κατέστη εύκολη η μετάβαση σε αυτό καθώς είναι πολύ φιλικό προς τον χρήστη ως framework.

Δεν θα μπορούσαμε σαφώς να μην σταθούμε και στις δυσκολίες, που αντιμετωπίσαμε κατά τη διάρκεια εκπόνησης της παρούσας πτυχιακής εργασίας. Σε ένα πρώτο επίπεδο ήταν ιδιαίτερα απαιτητική και δύσκολη η ανάκτηση πληροφοριών από την πλατφόρμα Scopus καθώς δεν είχαμε ακαδημαϊκή πρόσβαση σε αυτήν με κάποιο public API, το οποίο θα μας διευκόλυνε να ανακτήσουμε τα απαραίτητα δεδομένα και τις πληροφορίες που χρειαζόμασταν από τη συγκεκριμένη πλατφόρμα. Παρόμοια δυσκολία αντιμετωπίσαμε και με το Google Scholar με αποτέλεσμα να δημιουργηθεί μία βάση δεδομένων, στην οποία η εκχώρηση των νέων στοιχείων γινόταν χειροκίνητα.

Τέλος θα επιθυμούσαμε να αναφερθούμε και σε πιθανές προεκτάσεις, που θα μπορούσε να έχει η συγκεκριμένη έρευνα. Σε πρωταρχικό επίπεδο, μία πιθανή επέκταση θα ήταν η δημιουργία ρόλων χρηστών, όπου ο κάθε χρήστης – συμπεριλαμβανομένων και των καθηγητών- θα μπορούσε να επεξεργαστεί τις πληροφορίες και τα δεδομένα πέραν του διαχειριστή. Ακολούθως, μία ενδεχόμενη προέκταση θα μπορούσε να είναι η εξής: η εφαρμογή μας θα ήταν εφικτό να σχεδιαστεί με τέτοιο τρόπο έτσι ώστε να προτείνει σε κάθε καθηγητή που συμπεριλαμβάνεται στη βάση δεδομένων πιθανή συνεργασία με άλλους καθηγητές – ερευνητές, με βάση την εγγύτητα των δημοσιεύσεών τους. Επιπρόσθετα, θα μπορούσαμε σε μία μελλοντική μας προσπάθεια να αξιοποιήσουμε έναν πιο εξειδικευμένο αλγόριθμο σχετικά με τη σύγκριση των κειμένων. Ειδικότερα, το Bag

of Words παρουσιάζει ορισμένους περιορισμούς, οι οποίοι έγκειται στα εξής σημεία: δεν αναγνωρίζει τις διαφορετικές μορφές σύνταξης που επηρεάζουν την σημασιολογία, δεν καταμετρά ως ίδιες συνώνυμες λέξεις και φαίνεται να αγνοεί σπάνιες λέξεις, οι οποίες ωστόσο είναι σημασιολογικά εύχρηστες.

9. Βιβλιογραφία

9.1 Ξενόγλωσση Βιβλιογραφία

- Bahill, T. A. & Henderson, S.J. *Requirements Development, verification and validation exhibited in famous failures*. 2004.
- Brownlee J. *A Gentle Introduction to the Bag-of- Words Model*. 2017 Διαθέσιμο στον ιστότοπο: <https://machinelearningmastery.com/gentle-introduction-bag-words-model/>.
- Carbonell N. et al. *A knowledge based approach to the design of a man-machine dialog system by voice*, Vandoeuvre les Nancy, France 1987.
- Choptra A. et al. *Natural Processing Language*, INTERNATIONAL JOURNAL OF TECHNOLOGY ENHANCEMENTS AND EMERGING ENGINEERING RESEARCH, VOL 1, ISSUE 4 131 2013 ISSN 2347-4289.
- Jin R. & Zhou Z-H. *Understanding bag-of-words: a statistical framework*, International Journal of Machine Learning and Cybernetics, DOI: 10.1007/s13042-010-0001-0.
- Liddy E. *Natural Processing Language*, Syracuse University 2001.
- Maguire, M. & Bevan, N. *User requirements analysis*. World Computer Congress: 2002.
- Mitchell T. *Machine Learning*, Professional Book Group 11 West 19th Street New York, NY,United States 1997.
- Singh V. *What is Frameworks?*, 2020. Διαθέσιμο στον ιστότοπο: <https://hackr.io/blog/what-is-frameworks>.
- Weigers,K. *Vision an Scope Document for cafeteria ordering system*. 2002.
- Zhou V. *A simple explanation of the Bag-of-Words*, 2019. Διαθέσιμο στον ιστότοπο: <https://towardsdatascience.com/a-simple-explanation-of-the-bag-of-words-model-b88fc4f4971>.
- Λεύι Α. «Επιστημονική εργασία» *Εύρεση πηγών Άξονες δομής επιστημονικού άρθρου (αναγνώριση) Κανόνες γραφής επιστημονικού άρθρου (αναγνώριση)*»2017. Διαθέσιμο στον ιστότοπο: <https://slideplayer.gr/slide/11124489/>.

- <https://el.wikipedia.org/wiki/HTML>
- <https://el.wikipedia.org/wiki/CSS>
- https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html
- <https://towardsdatascience.com/basics-of-countvectorizer-e26677900f9c>
- <https://www.educative.io/edpresso/countvectorizer-in-python>
- <https://www.mygreatlearning.com/blog/bag-of-words/>
- <https://towardsdatascience.com/natural-language-processing-count-vectorization-with-scikit-learn-e7804269bb5e>
- https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html
- <https://www.machinelearningplus.com/nlp/cosine-similarity/>

9.2 Ελληνική Βιβλιογραφία

- «Γλώσσες Τέταρτης Γενιάς», 2008. Διαθέσιμο στον ιστότοπο: http://cgi.di.uoa.gr/~std06249/pliroforiki&ekpaideusi/_5.html.
- «Μηχανές Αναζήτησης Επιστημονικών Άρθρων». Διαθέσιμο στον ιστότοπο: <https://www.anavasis.gr/blog/anazitisi-epistimonikon-arthron>
- Αβούρης Ν. et al., 2015, «Μοντέλα και μέθοδοι σχεδίασης διαδραστικών συστημάτων», Εκδόσεις Καλλίπος
- Βεσκούκης Β., 2015, «Στοιχεία Τεχνολογίας Λογισμικού», Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών, Εθνικό Μετσόβιο Πολυτεχνείο.
- Γαλάνης Π., 2014, «Βασικές Αρχές Συγγραφής Ερευνητικών Άρθρων», Αρχεία Ελληνικής Ιατρικής 2014, 31(1)97-107, Copyright ©Athens Medical Society, ARCHIVES OF HELLENIC MEDICINE: ISSN 11-05-3992
- Γεωργούλη Κ. *Τεχνητή Νοημοσύνη – Μια Εισαγωγική Προσέγγιση*, Εκδόσεις ΣΕΑΒ 2015.
- Ιωαννίδης , Ι. & Λέπουρας, Γ. *Σημειώσεις επικοινωνίας ανθρώπου – μηχανής*. Αθήνα: 2005.

- Κουτραφούρη Δ.& Κωνσταντινίδης Π & Μυριανθεύς Π., «Κριτική Ανάλυση Επιστημονικού Άρθρου», Πανεπιστημιακή Μ.Ε.Θ, Γ.ΟΝ.Κ Άγιοι Ανάργυροι.
- Μητρόπουλος Σ. & Δουληγέρης Χ., 2015, «Πληροφοριακά Συστήματα στο Διαδίκτυο», Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών, Εθνικό Μετσόβιο Πολυτεχνείο
- Μπεληγιάννης, Α. *Μια εισαγωγή στη βασική άλγεβρα*. Εκδόσεις: Κάλλιπος, Αθήνα 2015.
- Σκορδαλάκης, Ε. *Λογισμική Μηχανική*. Αθήνα: 2003.
- Φλούρης Α., «Συγγραφή Επιστημονικού Άρθρου». Διαθέσιμο στον ιστότοπο: <http://www.eevfa.gr/site/documents/Frontistirio%203.pdf>
- Χρούσος Γ.Π. & Ζουμάκης Ε., 2009, «Βιβλιογραφική αξιολόγηση ερευνητικού έργου θετικών επιστημών», Δελτ Α' Παιδιατρική Κλινική Πανεπιστημίου Αθηνών 2009, 56(1):473-475.