ePublishing

Open Schools Journal for Open Science

Vol 3, No 6 (2020)

Μία εφαρμογή τεχνητής νοημοσύνης σε γλώσσα προγραμματισμού Java

Αναστάσιος Ταχτατζής, Απόστολος Φιδάνης, Λαμπρινός Χατζηιωάννου, Ιωάννης Ζήκου

doi: 10.12681/osj.24310

Copyright © 2020, Αναστάσιος Ταχτατζής, Απόστολος Φιδάνης, Λαμπρινός Χατζηιωάννου, Ιωάννης Ζήκου

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0.

To cite this article:

Ταχτατζής Α., Φιδάνης Α., Χατζηιωάννου Λ., & Ζήκου Ι. (2020). Μία εφαρμογή τεχνητής νοημοσύνης σε γλώσσα προγραμματισμού Java. *Open Schools Journal for Open Science*, *3*(6). https://doi.org/10.12681/osj.24310

https://epublishing.ekt.gr | **e-Publisher:** EKT | **Downloaded at:** 01/01/2026 23:50:43

Open Schools Journal
for Open Science

Μία εφαρμογή τεχνητής νοημοσύνης σε γλώσσα προγραμματισμού Java.

Ταχτατζής Αναστάσιος[1], Φιδάνης Απόστολος[1], Χατζηιωάννου Λαμπρινός[1], Ζήκου Ιωάννης[2]

[1]1ο Πειραματικό ΓΕΛ Θεσσαλονίκης «Μανόλης Ανδρόνικος», Θεσσαλονίκη, Ελλάδα,

[2]Πληροφορικός ΠΕ19, 1ο Πειραματικό ΓΕΛ Θεσσαλονίκης «Μανόλης Ανδρόνικος», Θεσσαλονίκη, Ελλάδα

ΠΕΡΙΛΗΨΗ

Η τεχνητή νοημοσύνη και οι έξυπνες εφαρμογές αποτελούν στις μέρες μας αναπόσπαστο κομμάτι της καθημερινότητας μας, από τα κινητά και τους υπολογιστές μέχρι και τα ψυγεία και άλλες συσκευές καθημερινής χρήσης, με απεριόριστες δυνατότητες επέκτασης στο μέλλον. Με αφορμή τις εφαρμογές τεχνητής νοημοσύνης στην καθημερινή μας ζωή (π.χ. Siri, Cortana, Google Assistant) αποφασίσαμε να δημιουργήσουμε μια πρωτόγονη μορφή τεχνητής νοημοσύνης, προγράμματος δηλαδή που να λαμβάνει αποφάσεις βασισμένο σε δεδομένα που δόθηκαν σε κάποια προηγούμενη χρονική στιγμή, με τη χρήση της υψηλού επιπέδου γλώσσας προγραμματισμού Java. Το πρόγραμμα μας, Remin, στοχεύει τόσο στη βελτιστοποίηση του εβδομαδιαίου προγράμματος του χρήστη όσο και στην έγκαιρη ενημέρωση του για τις επικείμενες δραστηριότητες του. Με την κατάλληλη αποθήκευση του προγράμματος του χρήστη σε βάση δεδομένων έγινε επιτρεπτή η χρήση τους για τη λήψη αποφάσεων, ενώ η συγγραφή του κώδικά της εφαρμογής έγινε με χρήση της αντικειμενοστραφούς γλώσσας προγραμματισμού Java. Σκοπός της συγκεκριμένης εφαρμογής είναι η μελέτη των δυνατοτήτων της τεχνητής νοημοσύνης και οι μελλοντικές της εφαρμογές στον μέλλον, στην καθημερινότητα

μας. Για να το επιτύχουμε αυτό θα μελετήσουμε σύγχρονες μορφές τεχνητής νοημοσύνης, και θα δούμε σε ποιό στάδιο έχει εξελιχτεί αυτή τη στιγμή ο κλάδος αυτός, καθώς και τα σχέδια που υπάρχουν για μελλοντικές καινοτομίες. Με αυτό τον τρόπο θα φτάσουμε σε ένα συμπέρασμα για τα όρια που μπορεί να φτάσει κανείς μέσω της τεχνητής νοημοσύνης και αν είναι εφικτή η κατασκευή ρομπότ με εντελώς ανθρώπινη συμπεριφορά. Τα αποτελέσματα αναμένεται να είναι θετικά, αλλά όχι στο εγγύς μέλλον.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ

Java, Artificial Intelligence, Object-Oriented Programming

ΕΙΣΑΓΩΓΗ

Στα πλαίσια του ομίλου προγραμματισμού Java που λειτουργεί στο σχολείο μας την φετινή χρονιά, αποφασίσαμε να ασχοληθούμε με την εκμάθηση της γλώσσας προγραμματισμού Java. Η ομάδα μας έκανε συναντήσεις μετά την λήξη του σχολικού ωραρίου στο εργαστήριο πληροφορικής που διαθέτει το σχολείο μας και παρακολουθούσε μαθήματα Java από την Oracle. Επιπλέον, από ενδιαφέρον για τον ταχύτατα αναπτυσσόμενο κλάδο της Τεχνητής Νοημοσύνης παρακολουθούσαμε σχετικά διαδικτυακά μαθήματα (online-courses).

Όταν λοιπόν συνειδητοποιήσαμε ότι θα μας ήταν ιδιαίτερα χρήσιμη μία εφαρμογή "Ημερολόγιο¨με λειτουργίες παρόμοιες με των υπολοίπων εφαρμογών αυτού του είδους αλλά με βελτιστοποιήσεις για τη μαθητική ζωή δεν διστάσαμε, μα αρχίσαμε κατευθείαν να δουλεύουμε πάνω στην ανάπτυξή της.

ΤΟ ΘΕΩΡΗΤΙΚΟ ΠΛΑΙΣΙΟ ΤΗΣ ΕΦΑΡΜΟΓΗΣ ΜΑΣ

Η γλώσσα προγραμματισμού Java

Γενικά στοιχεία:

Η **Java** είναι μια από τις πλέον διαδεδομένες γλώσσες προγραμματισμού παγκοσμίως. Ξεκίνησε τη δεκαετία του 1990 ως τρόπος σύνδεσης των υπολογιστών με τις διάφορες καταναλωτικές συσκευές που κυκλοφορούσαν από μία ομάδα της Sun Microsystems. Η προσπάθεια αυτή δυστυχώς τερματίστηκε πρώιμα εξαιτίας της προοδευτικότητας της, μα το 1995 με την εφαρμογή της στο φυλλομετρητή Netscape Navigator επανήλθε στο προσκήνιο *(Oracle "The History Of Java Technology", 2018)* και από τότε η ανάπτυξή της συνέχισε, με αποτέλεσμα να χρησιμοποιείται σε πάνω από 7 δισεκατομμύρια συσκευές *(Ivonum "Java Is The Number One Programming Language", 2018)*.

Γιατί επιλέχθηκε:

Ένα από τα βασικά πλεονεκτήματα της Java έναντι των περισσότερων άλλων γλωσσών, και ένας βασικός λόγος για τον οποίο καταλήξαμε στη χρήση της συγκεκριμένης γλώσσας αντί άλλων με τις οποίες έχουμε φαινομενικά περισσότερη εμπειρία (όπως η C και η C++) είναι η ανεξαρτησία του λειτουργικού συστήματος και πλατφόρμας. Αναλυτικότερα, τα προγράμματα που είναι γραμμένα σε *Java* τρέχουνε ακριβώς το ίδιο σε Windows, Linux, Unix και Macintosh (σύντομα θα τρέχουν και σε Playstation καθώς και σε άλλες κονσόλες παιχνιδιών) χωρίς να χρειαστεί να ξαναγίνει μεταγλώττιση (compiling) ή να αλλάξει ο πηγαίος κώδικας για κάθε διαφορετικό λειτουργικό σύστημα *(Gosling, McGilton "The Java Language Environment : a White Paper", 1995)*.

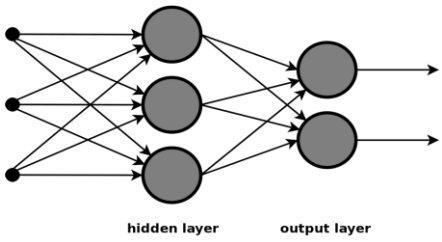
Σχήμα 1: Το λογότυπο της Java

Η τεχνητή νοημοσύνη

Ο όρος **τεχνητή νοημοσύνη** αναφέρεται στον κλάδο της πληροφορικής ο οποίος ασχολείται με τη σχεδίαση και την υλοποίηση υπολογιστικών συστημάτων που μιμούνται στοιχεία της ανθρώπινης συμπεριφοράς τα οποία υπονοούν έστω και στοιχειώδη ευφυΐα: μάθηση, προσαρμοστικότητα, εξαγωγή συμπερασμάτων, κατανόηση από συμφραζόμενα, επίλυση προβλημάτων κλπ. *(Technopedia, "What Is Artificial Intelligence (AI)?)*. Ο Τζον Μακάρθι όρισε τον τομέα αυτόν ως «*επιστήμη και μεθοδολογία της δημιουργίας νοούντων μηχανών*»

Η συμβατική τεχνητή νοημοσύνη εμπλέκει μεθόδους μηχανικής μάθησης (machine learning), που χαρακτηρίζονται από αυστηρούς μαθηματικούς αλγόριθμους και στατιστικές μεθόδους ανάλυσης. Διακρίνεται σε:

- Έμπειρα ή Εξειδικευμένα συστήματα (Expert systems), που εφαρμόζουν προγραμματισμένες ρουτίνες λογικής, σχεδιασμένες αποκλειστικά για μία συγκεκριμένη εργασία, προκειμένου να εξαχθεί κάποιο συμπέρασμα. Για το σκοπό αυτό, διεξάγεται επεξεργασία μεγάλων ποσοτήτων γνωστών πληροφοριών.
- Λογική κατά περίπτωση (Case based reasoning). Η επίλυση ενός προβλήματος βασίζεται στην προηγούμενη επίλυση παρόμοιων προβλημάτων.
- Μπαϋεσιανά δίκτυα (Bayesian networks). Βασίζονται στη στατιστική ανάλυση για τη λήψη αποφάσεων.
- Συμπεριφορική τεχνητή νοημοσύνη (Behavior based AI). Μέθοδος τεμαχισμού της λογικής διαδικασίας και στη συνέχεια χειροκίνητης οικοδόμησης του αποτελέσματος.



Σχήμα 2: Διάγραμμα αναζήτησης στην Τεχνητή νοημοσύνη

Επίλυση προβλημάτων με τεχνητή νοημοσύνη

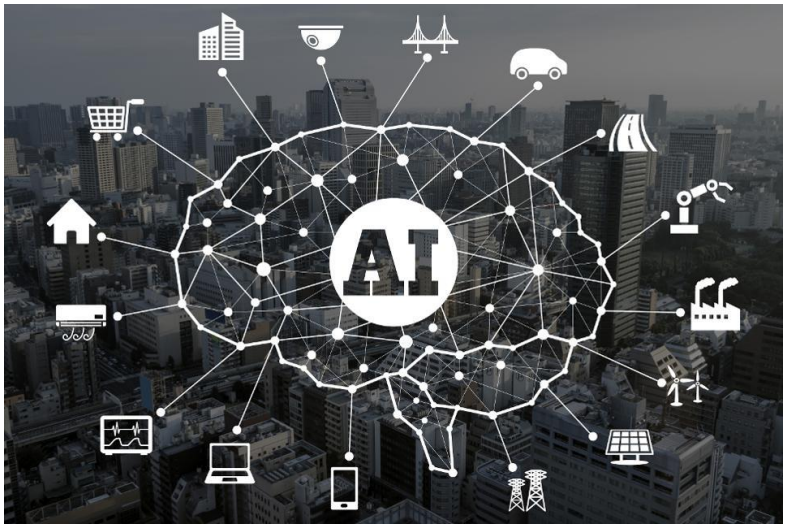
Κύρια λειτουργία και λόγος ύπαρξης της τεχνητής νοημοσύνης είναι η επίλυση προβλημάτων. Αυτό γίνεται με την βοήθεια διαφόρων αλγόριθμων όπως:

- Αναζήτηση κατά πλάτος

Αναζήτηση κατά πλάτος (BFS - Breadth First Search) είναι ένας αλγόριθμος για διάσχιση ή αναζήτηση σε δομές δεδομένων τύπου δέντρου ή γράφου. Η αναζήτηση ξεκινά από τη ρίζα του δέντρου και διερευνά πρώτα τους γειτονικούς κόμβους, προτού μεταβεί στους γείτονες του επόμενου επίπεδο.

- Αναζήτηση κατά βάθος

Ο αλγόριθμος **Αναζήτησης Κατά Βάθος** (**DFS - Depth-first search**) επιτυγχάνει διάσχιση ή αναζήτηση σε δέντρο ή γράφημα. Η διάσχιση ξεκινά από τη ρίζα και εξερευνά όσο το δυνατόν περισσότερο κατά μήκος κάθε κλαδί του δέντρου μέχρι να φτάσει σε αδιέξοδο.



Σχήμα 3: Δείγματα εφαρμογών τεχνητής νοημοσύνης στην καθημερινή μας ζωή

Ο Πράκτορας στην τεχνητή νοημοσύνη

Πράκτορας είναι ένα υπολογιστικό σύστημα που έχει και ιδιότητες που συναντάμε στους ανθρώπους, όπως γνώση, πεποίθηση, υποχρέωση ή ακόμα και συναισθήματα. Μερικές από τις

ιδιότητες που εντάσσονται στην ισχυρή θεώρηση του πράκτορα είναι οι παρακάτω:

1. Ικανότητα Μετακίνησης (mobility): Είναι η ικανότητα του πράκτορα να μετακινείται σε διάφορες τοποθεσίες μέσα σε ένα δίκτυο.

2. Καλοσύνη (benevolence): Οι πράκτορες δεν έχουν αλληλοσυγκρουόμενους στόχους, οπότε ο κάθε πράκτορας προσπαθεί να κάνει ό,τι του ζητηθεί.

3. Ορθολογικότητα (rationality): Ο πράκτορας ενεργεί έτσι ώστε να επιτυγχάνει τους στόχους του.

4. Προσαρμοστικότητα (adaptivity): Η ικανότητα του πράκτορα να προσαρμόζεται στις συνθήκες του περιβάλλοντος.

Είναι αρκετά δύσκολο να ορίσουμε τι ακριβώς είναι αυτό που κάνει έναν πράκτορα "ευφυή". Ωστόσο, μπορούμε γενικά να πούμε ότι ένας πράκτορας είναι ευφυής όταν έχει την ικανότητα να επιτελεί τους στόχους και τα καθήκοντα που έχει επιφορτιστεί, καθώς επίσης να προσαρμόζεται και να μαθαίνει τα ενδιαφέροντα του χρήστη, ώστε να του προσφέρει μια ουσιαστική υπηρεσία. Για αυτόν τον λόγο μπορούμε να πούμε ότι ένας πράκτορας είναι ευφυής όταν συνδυάζει τεχνικές που παρέχουν αυτές τις δυνατότητες, δηλαδή κυρίως τεχνικές από τους χώρους της Τεχνητής ή Υπολογιστικής Νοημοσύνης (Artificial or Computatinal Intelligence)

ΟΙ ΕΦΑΡΜΟΓΕΣ ΤΗΣ ΤΕΧΝΗΤΗΣ ΝΟΗΜΟΣΥΝΗΣ

Η τεχνητή νοημοσύνη στην καθημερινότητά μας

Η πρόοδος της τεχνητής νοημοσύνης σε συνδυασμό με την ταχύτατα αυξανόμενη δυνατότητα αποθήκευσης και επεξεργασίας δεδομένων, οδηγεί μία σπουδαία τεχνολογική επανάσταση. Σε ένα κοντινό σενάριο και στα πλαίσια του Internet of Things, δισεκατομμύρια διασυνδεδεμένες συσκευές θα αποστέλλουν, συστηματικά, δεδομένα και events που θα αποθηκεύονται και επεξεργάζονται αυτόματα από προηγμένες τεχνολογίες Big Data, Machine Learning και Artificial Intelligence. Ο πλούτος των δεδομένων σε συνδυασμό με τις δυνατότητες αξιοποίησής τους, θα δημιουργήσει πρωτοφανείς ευκαιρίες για την βελτίωση της *υγείας, της καθημερινότητας και του τρόπου ζωής μας, των μεταφορών και του περιβάλλοντος*. Οι εφαρμογές είναι πραγματικά απεριόριστες.

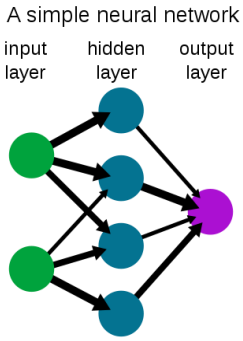
Η τεχνητή νοημοσύνη έχει επιτυχημένα χρησιμοποιηθεί στους παρακάτω τομείς:

- Οικονομικά (Για τον υπολογισμό μεγάλων ποσών με ακρίβεια)
- Ρομποτική (Για την δημιουργία αυτόνομων ρομπότ)
- Παιχνίδια (Εικονικοί αντίπαλοι σε παιχνίδια)
- Φαρμακευτική (Κυρίως για την διάγνωση ασθενειών)
- Διαδίκτυο (Για την αναζήτηση πληροφοριών στο ίντερνετ)

Η υπολογιστική τεχνητή νοημοσύνη

Η υπολογιστική νοημοσύνη, αποτελεί κλάδο της τεχνητής νοημοσύνης και ασχολείται με την αλγοριθμική επίλυση προβλημάτων. Αυτό επιτυγχάνεται με διάφορους τρόπους ένας από τους οποίους τα νευρωνικά δίκτυα. Το νευρωνικό δίκτυο είναι ένα δίκτυο από απλούς υπολογιστικούς κόμβους (νευρώνες, νευρώνια), διασυνδεδεμένους μεταξύ τους. Είναι εμπνευσμένο από το Κεντρικό Νευρικό Σύστημα (ΚΝΣ), το οποίο προσπαθεί να προσομοιώσει. Τα νευρωνικά δίκτυα αποτελούνται από τρία είδη νευρώνων:

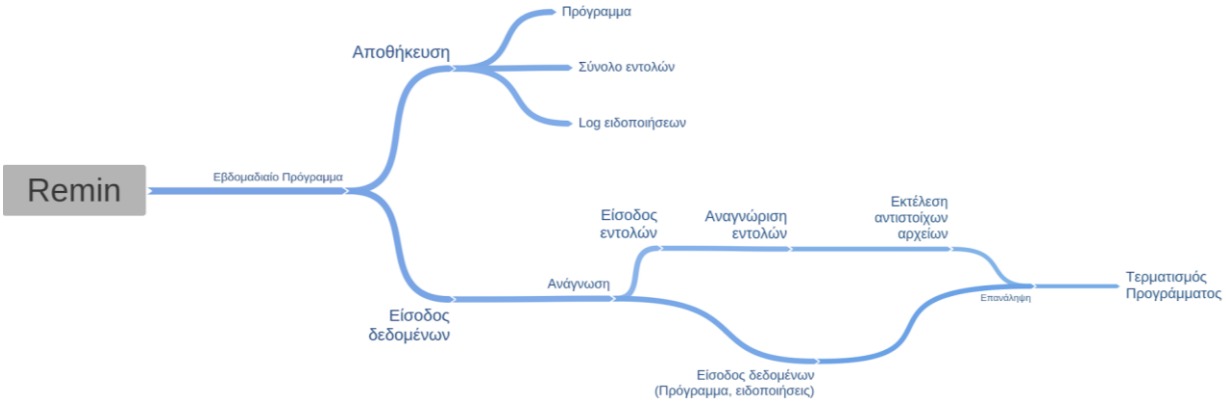
- Τους νευρώνες εισόδου, οι οποίοι μεσολαβούν ανάμεσα στις περιβαλλοντικές εισόδους του δικτύου και στους υπολογιστικούς νευρώνες.
- Τους νευρώνες εξόδου, οι οποίοι διοχετεύουν στο περιβάλλον τις τελικές αριθμητικές εξόδους του δικτύου.
- Τους υπολογιστικούς νευρώνες, των οποίων ρόλος είναι η επεξεργασία των δεδομένων των νευρώνων εισόδου και η διοχετευση τους στους νευρώνες εισόδου. Η επεξεργασία αυτή γίνεται μέσω των προεπιλεγμένων συναρτήσεων που χρησιμοποιεί ο κόμβος.



Σχήμα 5: Το πιο απλό νευρωνικό δίκτυο

Τα νευρωνικά δίκτυα εκπαιδεύονται με παραδείγματα. Ο χρήστης συγκεντρώνει αντιπροσωπευτικά δεδομένα και στη συνέχεια, καθώς τα τροφοδοτεί συστηματικά στο δίκτυο μέσω των κατάλληλων αλγορίθμων εκπαίδευσης, το δίκτυο «αντιλαμβάνεται» αυτομάτως τη δομή των δεδομένων. Για αυτό τον λίγο είναι πολύ εύχρηστα σε πολλούς επιστημονικούς κλάδους όπως, στην ιατρική και τα χρηματοοικονομικά.

Η ΕΦΑΡΜΟΓΗ

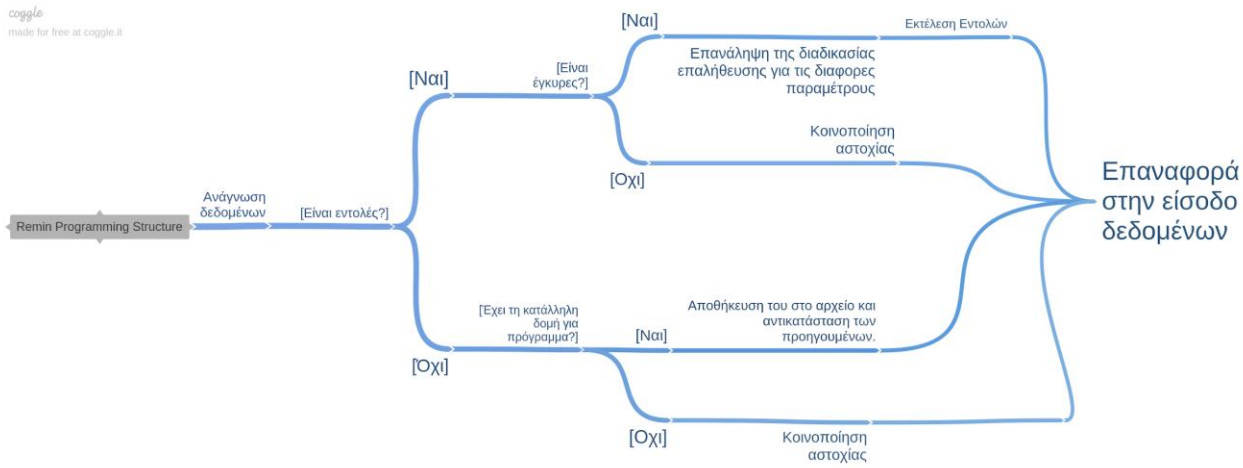


Σχήμα 6: Διάγραμμα με τη γενική δομή του προγράμματος. Στο κλάδο αποθήκευση καταγράφονται τα τρία διαφορετικά αρχεία τα οποία χρησιμοποιεί η εφαρμογή ενώ στο κάτω μέρος φαίνεται η απλότερη εκτέλεση του προγράμματος.

Τρόπος λειτουργίας

Το Remin έχει μερικές πολύ ενδιαφέρουσες λειτουργίες, όπως για παράδειγμα την ανάθεση ξυπνητηριών/reminders και τη κατασκευή ενός ολοκληρωμένου προγράμματος του χρήστη και ενημέρωση του για το αν το ακολουθεί πιστά ή υπάρχει κάτι που θα πρέπει να διορθωθεί . Με την εκκίνηση του προγράμματος, ελέγχεται η πιθανή διαθεσιμότητα μιας προηγουμένως κατασκευασμένης βάσης δεδομένων στον φάκελο του προγράμματος με σκοπό την άντληση των απαραίτητων πληροφοριών. Αν το πρόγραμμα δεν την βρει τότε κατασκευάζει μία νέα και απαιτεί την εισαγωγή των απαραίτητων πληροφοριών για την λειτουργία του από όπου και

περνάει στο βήμα 3.Εκείνο τρέχει πάντοτε ανεξαρτήτως του αποτελέσματος του if statement που παρουσιάστηκε προηγουμένως και εκτελείται κατά την εκκίνηση της εφαρμογής. Δηλαδή, το πρόγραμμα πλέον έχει τυπώσει στο terminal μήνυμα στο οποίο ενημερώνει το χρήστη για την λειτουργία της εφαρμογής και προτρέποντας το χρήστη να επιλέξει μία εντολή από τη συνεχώς αυξανόμενη λίστα εντολών που έχει το πρόγραμμα μας.(Παραδείγματα: remind για την κατασκευή reminder οπου μπορεί να πάρει μέχρι και 4 παραμέτρους για ωρα, μέρα, όνομα και περιγραφή, q για το τερματισμό του προγράμματος, m για την εμφάνιση των οδηγιών και περιγραφών των εντολών και το smartrem, το οποίο ενημερώνει τον χρήστη μέχρι το τερματισμό του συγκεκριμένου παραθύρου της γραμμής εντολών για τις επικείμενες (με βάση τη ρουτίνα του) δραστηριότητές του.Από εκεί και μετά όλα καθορίζονται από τις επιλογές του χρήστη αν και υπάρχουν περιορισμένες ακόμα δυνατότητες.



Σχήμα 7: Διάγραμμα δομής της εφαρμογής. Περιγράφει την διαδικασία από την είσοδο των δεδομένων μέχρι και τον τερματισμό του προγράμματος.

Μία βαθύτερη διερεύνηση του προγράμματος

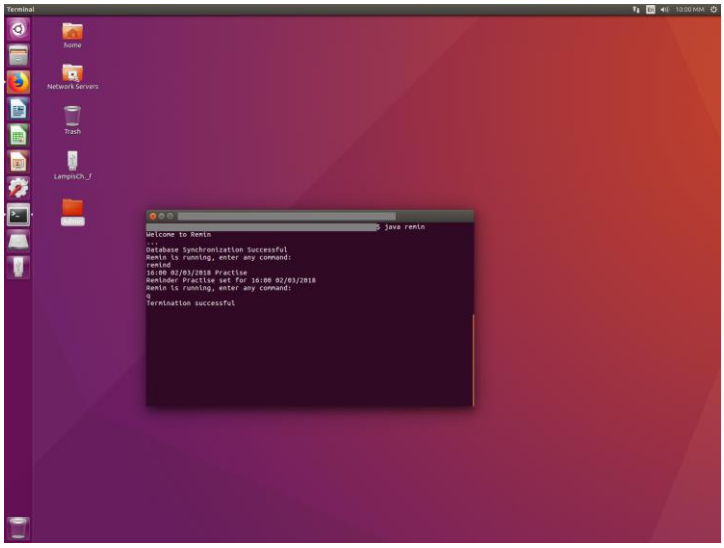
Το πρόγραμμα αποτελείται από τα packages reminData και reminUse, φτιαγμένα ολοκληρωτικά από εμάς και από τη driver class reminDriver. Ειδικότερα, στο πρώτο package έχουν ενταχθεί όλες οι κλάσεις, πιο συγκεκριμένα 3 (Command, ReadInput, WriteOutput), οι οποίες να ασχολούνται αποκλειστικά με τον έλεγχο των δεδομένων (Ανάγνωση από αρχείο και

από τη γραμμή εντολών, επαλήθευση εγκυρότητας δεδομένων και εκτύπωση σε αρχείο και στη γραμμή εντολών) ενώ το δεύτερο ασχολείται αποκλειστικά με την επεξεργασία των δεδομένων. Περιέχει κλάσεις που ελέγχουν την το Ημερολόγιο (οχι τη δομή αλλά τις λειτουργίες του), τις Ειδοποιήσεις (οι οποίες είναι το μόνο μέρος του κώδικά μας με γραφικό περιβάλλον, έτσι ώστε ο χρήστης να μπορεί να ειδοποιείται ακόμα και αν έχει άλλο παράθυρο ανοιχτό). Τέλος στο δεύτερο πακέτο βρίσκεται και η κλάση η οποία αποτελεί τον καρπό της έρευνάς μας στην Τεχνητή Νοημοσύνη, η reminAI. Το όνομα της αν και αναμφίβολα φιλόδοξο δεν είναι εντελώς ψευδές. Υπολογίζοντας και μετέπειτα χρησιμοποιώντας στατιστικά σχετικά με τον χρόνο μετακίνησης του χρήστη στις διάφορες δραστηριότητές του, δημιουργεί "έξυπνες" ενημερώσεις, έτσι ώστε να μην χρειάζεται ο χρήστης να θυμάται να εισάγει τις διάφορες ειδοποιήσεις αλλά να τις έχει έτοιμες από το πρόγραμμα και πάντα την σωστή ώρα. Παρακάτω παρατίθεται κομμάτι του κώδικα, σχεδιασμένο για επαλήθευση των εντολών:

```
boolean commandVerification(String passedCommand,command[] activeCommands,int passedArgs, String secArgs){
	int tempPtr = 0;
	boolean tempBool = true;
	for(; tempPtr <=  activeCommands.length; tempPtr++){
		if(passedCommand == activeCommands[tempPtr].name){
			if(activeCommands[tempPtr].numberOfParams >= passedArgs){
				/*paramQualification method returns a boolean and takes as argument a string of consecutive characters and a command name to check whether they are functional parameters for the desired command */
				return paramQualification(passedArgs, activeCommands[tempPtr).name);
			return false;
			}
	return false;
}
```

Περιβάλλον χρήστη

Την αρχική ιδέα αποτελούσε η συλλογή δεδομένων μέσω φωνητικών εντολών μας στην πορεία καταλάβαμε πως κάτι τέτοιο όχι μόνο απαιτούσε πολλά δείγματα αλλά θα στερούσε από το πρόγραμμά μας και ένα μέρος της αποδοτικότητας που θέλουμε να το χαρακτηρίζει (εξαιτίας της πολυπλοκοτητας του), για αυτό και αποφασίσαμε τελικά πως θα τρέχει μέσα από τη γραμμή εντολών, μέχρι στιγμής μόνο σε υπολογιστές με λειτουργικό Linux και Unix (Macintosh). Δηλαδή, μέχρι στιγμής η εφαρμογή μας δεν διαθέτει GUI (Graphic User Interface) αλλά εκμεταλλεύεται την τρομακτική δύναμη του terminal προκειμένου να συλλέγει άμεσα και αποτελεσματικά τα δεδομένα που εισάγει ο χρήστης μέσω του πληκτρολογίου και να δρα άμεσα. Σε μελλοντική έκδοση σκοπεύουμε με την εκμετάλλευση του πακέτου JavaFx να κατασκευάσουμε ένα ανεπτυγμένο περιβάλλον για το χρήστη ετσι ώστε να μπορεί η εφαρμογή να απευθυνθεί και σε άτομα με λιγότερες γνώσεις του terminal..



Σχήμα 8: Εκτέλεση της εφαρμογής μας για τη δημιουργία ενός ξυπνητηριού.

ΕΠΙΛΟΓΟΣ

Συνοψίζοντας τα παραπάνω, η δημιουργία μιας τέτοιας εφαρμογής, ακόμα και αυτής της απλότητας είχε τις δυσκολίες της. Αποδείχτηκε όμως, ότι είναι πιθανή η χρήση τεχνητής νοημοσύνης σε εφαρμογές όλων των ειδών και δεν είναι απαραίτητη η επαγγελματική

ενασχόληση με το συγκεκριμένο τομέα για την υλοποίηση ενός τέτοιου προγράμματος. (Εννοείται ότι το πρόγραμμα μας δεν μπορεί να θεωρηθεί τίποτα παραπάνω από εισαγωγικό στην Τεχνητή Νοημοσύνη εξαιτίας του περιορισμού του σε Machine Learning λειτουργίες, όπως προαναφέρθηκαν τα παραδείγματα των έξυπνων ειδοποιήσεων)

ΕΥΧΑΡΙΣΤΙΕΣ

Θα θέλαμε να ευχαριστήσουμε τους καθηγητές μας, κύριο Ν. Μιχαηλίδη και κύριο Ι. Ζήκου, που μας στήριξαν σε αυτή τη προσπάθεια, ενθαρρύνοντας μας και προμηθεύοντάς μας με χρήσιμο υλικό και συμβουλές όποτε το χρειαζόμασταν.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Για τη δημιουργία των διαγραμμάτων χρησιμοποιήθηκε η εφαρμογή Coggle: *https://coggle.it*

[2] Java Is the Number One Programming Language, *www.inovum-solutions.com/wp/news/java-is-the-number-one-programming-language/*

[3] Artificial Intelligence: a Modern Approach. Pearson Education Limited, 2013.

[4] “The History of Java Technology.” Slowly Changing Dimensions, Oracle, *www.oracle.com/technetwork/java/javase/overview/javahistory-index-198355.html*

[5] Lin, Patrick, et al. Robot Ethics: the Ethical and Social Implications of Robotics. MIT Press, 2014.

[6] “What Is Artificial Intelligence (AI)?” Techopedia.com, *www.techopedia.com/definition/190/artificial-intelligence-ai.*