

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ  
«ΣΧΕΔΙΑΣΗ ΚΑΙ ΥΛΟΠΟΙΗΣΗ ΡΟΜΠΟΤ  
ΣΥΝΟΜΙΛΙΑΣ ΓΙΑ ΤΗΝ ΥΠΟΒΟΗΘΗΣΗ  
ΦΟΙΤΗΤΩΝ ΣΕ ΘΕΜΑΤΑ ΣΠΟΥΔΩΝ»



Του φοιτητή  
Μαρκουλάκη Ευάγγελου  
Αρ. Μητρώου: 174966

Επιβλέπων  
Διαμαντάρας Κωνσταντίνος  
Καθηγητής

**Φεβρουάριος 2022**

Σχεδίαση και υλοποίηση ρομπότ συνομιλίας για την υποβοήθηση φοιτητών σε θέματα σπουδών  
21237

Μαρκουλάκης Ευάγγελος  
Διαμαντάρας Κωνσταντίνος

Μάρτιος 2021  
Φεβρουάριος 2022

*Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.*

*Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Μαρκουλάκη Ευάγγελου που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.*

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

*«Εἰς μνήμην τοῦ ἐκλιπόντα Εὐάγγελου Λιάπτη»*



## Πρόλογος

Οι συνεχόμενες τεχνολογικές εξελίξεις, καθώς και οι ανάγκες της καθημερινότητας δίνουν τη δυνατότητα στον κλάδο να αναπτυχθεί με σκοπό την ανταπόκριση στα ζητήματα των χρηστών.

Η παρούσα εργασία αποτελεί διπλωματική εργασία του Τμήματος Πληροφορικής και Μηχανικών Υπολογιστών και πραγματοποιήθηκε με σκοπό να παρέχει γρήγορα χρήσιμες πληροφορίες στους φοιτητές του τμήματος. Λαμβάνοντας υπόψη το μέγεθος επιρροής και ενσωμάτωσης των ευφών συνομιλιών στη καθημερινότητα, το συγκεκριμένο θέμα της εργασίας θεωρείται από τα πιο σύγχρονα και ενδιαφέροντα ζητήματα. Η διαδικασία εκπόνησης του πρακτικού μέρους απαιτεί αφοσίωση και συνεχή εξέλιξη για την καλύτερη εκπαίδευση και απόδοση του εργαλείου.

## Περίληψη

Στη σημερινή εποχή οι υπολογιστές αποτελούν αναπόσπαστο κομμάτι της κοινωνίας και ειδικότερα της καθημερινότητας των ανθρώπων. Καθώς η συνεχής ανάπτυξη των νέων τεχνολογιών, αλλά και οι απαιτήσεις των χρηστών για αποδοτικότερες λειτουργίες αυξάνονται, είναι πολύ σημαντική η δημιουργία τεχνολογιών που μπορούν να καλύψουν τις ανάγκες του κοινού. Μία από αυτές τις ανάγκες είναι αυτή της επικοινωνίας και η δημιουργία νέων τρόπων επικοινωνίας και διαμοιρασμού πληροφοριών. Ειδικότερα, τα Πανεπιστημιακά ιδρύματα αποτελούν κέντρα τα οποία έχουν τη δυνατότητα, το δυναμικό και την τεχνογνωσία για να αναπτύξουν παρόμοιες πρακτικές. Το Chatbot είναι μια σχετικά νέα τεχνολογία, που θα μπορούσε να φανεί χρήσιμη σε αρκετούς τομείς. Αν και η διάδοση και το εύρος του πεδίου είναι σε υψηλά επίπεδα, περιθώρια εξέλιξης υπάρχουν. Στα πλαίσια της παρούσας διπλωματικής θα αναπτυχθεί και θα σχεδιαστεί ένα ευφύες διαδραστικό λογισμικό συνομιλίας (chatbot). Η χρήση του συγκεκριμένου chatbot απευθύνεται στους φοιτητές της σχολής και αποτελεί προϊόν ενημέρωσης και πληροφόρησης. Ξεκινώντας από μια ιστορική αναδρομή, η εργασία καταλήγει στην διεξαγωγή ερωτηματολογίου, με σκοπό την απάντηση του κυρίου ερωτήματος, που αφορά την αναγκαιότητα και τη χρησιμότητα ενός τέτοιου, εργαλείου μέσα στο τμήμα μηχανικών πληροφορικής και ηλεκτρονικών συστημάτων.

«Design and implementation of conversational robot  
to assist student in studies»

«Evangelos Markoulakis»

## **Abstract**

Nowadays computers are an integral part of society and especially of everyday life. As the continuous development of new technologies and the demands of users for more efficient functions increase, it is very important to create technologies that can meet the needs of the public. One of these needs is that of communication and the creation of new ways of communicating and sharing information. In particular, University institutions are centers, that have the capacity, potential and know-how to develop similar practices. Chatbots are a relatively new technology that could be useful in several areas. Although the spread and range of the field is high, there is room for improvement. Within the framework of this diploma, an intelligent interactive chat software (chatbot) will be developed and designed. The use of this chatbot is addressed to the students of the school and is a product of information and information. Starting from a historical background, the work ends with conducting a questionnaire in order to answer the main question concerning the necessity and usefulness of such a tool within the department of computer engineering and electronic systems.

## Ευχαριστίες

Θέλω να ευχαριστήσω μέσα από την καρδιά μου αυτούς του ανθρώπους που καθ' όλη τη διάρκεια της συγγραφής αυτής της εργασίας, δέχθηκα την υποστήριξη τους. Είμαι πολύ περήφανος που είχα υπέροχους ανθρώπους γύρω μου. Άνθρωποι οι οποίοι με ενέπνεαν, με ωθούσαν, με καθοδηγούσαν και μου έδιναν κίνητρο.

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου, Δρ Κωνσταντίνο Διαμαντάρα, που με βοήθησε να διαμορφώσω τα θέματα και τις μεθόδους μελέτης με την εμπειρία που έχει στο αντικείμενο. Τα σχόλια του με ενθάρρυναν να σκεφτώ βαθύτερα και να βελτιώσω το τρόπο σκέψης και τη δουλειά μου.

Θα ήθελα να εκφράσω την ευγνωμοσύνη μου σε όλους τους καθηγητές μου που μου δίδαξαν πολύτιμα μαθήματα και με καθοδήγησαν στις σπουδές μου στο τμήμα, παρέχοντάς μου όλες τις γνώσεις και τα εργαλεία που θα χρησιμοποιηθούν ως εφόδια στο μέλλον.

# Περιεχόμενα

Πρόλογος.....	vi
Περίληψη.....	vii
Abstract .....	viii
Ευχαριστίες .....	ix
Περιεχόμενα .....	x
Κατάλογος Εικόνων .....	xii
Κατάλογος Πινάκων.....	xiv
Συντομογραφίες.....	xv
Εισαγωγή.....	1
Κεφάλαιο 1ο: Λογισμικά ευφυούς συνομιλίας.....	2
1.1 Τι είναι τα chatbot; .....	2
1.2 Πορεία των Chatbots .....	2
1.2.1 Ιστορία των Chatbots.....	3
1.2.2 Δοκιμασία Turing.....	7
1.2.3 Βραβείο Loebner .....	8
1.2.4 Πρόσφατες εξελίξεις των Chatbots .....	8
1.3 Άνοδος των Chatbot .....	9
1.3.1 Αύξηση των χρηστών του Διαδικτύου. ....	9
1.3.2 Πρόοδος στην Τεχνολογία .....	9
1.3.3 Οικοσύστημα προγραμματιστών.....	10
1.3.4 Πλατφόρμες ανταλλαγής μηνυμάτων.....	10
Κεφάλαιο 2ο: Rasa Framework.....	11
2.1 Τι είναι το Rasa; .....	11
2.2 Rasa NLU.....	12
2.2.1 Κορμός Rasa.....	12
2.2.2 Μηχανισμοί και εργαλεία.....	14
2.3 Rasa X.....	22
Κεφάλαιο 3ο: Ανάπτυξη VagBot .....	23
3.1 Εγκατάσταση εργαλείων .....	23
3.1.1 Python.....	23

3.1.2	Anaconda .....	23
3.1.3	Rasa Open Source και Rasa X.....	24
3.1.4	MySQL.....	24
3.1.5	Ngrok.....	25
3.1.6	Postman .....	25
3.1.7	Git και GitHub.....	26
3.2	Δημιουργία project .....	26
3.3	Θεματολογία.....	28
3.4	Υλοποίηση.....	28
3.5	Κανόνες.....	32
3.6	Παραμετροποίηση .....	32
3.7	Παραδείγματα.....	33
3.7.1	Απλό παράδειγμα .....	33
3.7.2	Σύνθετο παράδειγμα με διάλογο .....	34
3.7.3	Σύνθετο παράδειγμα με εξόρυξη οντότητας .....	35
3.7.4	Σύνθετο παράδειγμα με εξόρυξη οντότητας και αναζήτηση σε βάση δεδομένων. ....	37
Κεφάλαιο 4ο:	Ανάπτυξη VagFront .....	39
Κεφάλαιο 5ο:	Αποτέλεσμα.....	49
Κεφάλαιο 6ο:	Αξιολόγηση .....	53
6.1	Σύνθεση ερωτηματολόγιου .....	53
6.2	Αποτελέσματα αξιολόγησης.....	54
Κεφάλαιο 7ο:	Συμπεράσματα και προτάσεις βελτίωσης.....	62
ΒΙΒΛΙΟΓΡΑΦΙΑ.....		63

## Κατάλογος Εικόνων

Εικόνα 1.1 Τα Chatbot και οι λειτουργίες τους Πηγή: kmslh.com .....	2
Εικόνα 1.2 Διασυνδεδεμένες εφαρμογές .....	3
Εικόνα 1.3 ELIZA: Το πρώτο chatbot Πηγή: web.njit.edu.....	4
Εικόνα 1.4 Dr.Sbaitso: Το πρώτο chatbot με ομιλία Πηγή: xspblog.com.....	5
Εικόνα 1.5 Το Chatbot ALICE Πηγή: yakbots.com.....	6
Εικόνα 1.6 Η πορεία των Chatbot Πηγή: Building better chatbots 2018.....	7
Εικόνα 1.7 Η δοκιμασία Turing Πηγή: infineon.com .....	8
Εικόνα 2.1 Το Rasa Framework Πηγή: rasa.com.....	11
Εικόνα 2.2 Το Rasa NLU Πηγή: rasa.com.....	12
Εικόνα 2.3 Rasa Stories Πηγή: rasa.com .....	13
Εικόνα 2.4 Rasa Rules Πηγή: rasa.com .....	13
Εικόνα 2.5 Rasa και εκτίμηση πρόθεσης Πηγή: rasa.com .....	14
Εικόνα 2.6 Rasa και εξαγωγή οντοτήτων Πηγή: rasa.com .....	16
Εικόνα 2.7 Οντότητες στο Rasa Framework Πηγή: rasa.com.....	17
Εικόνα 2.8 Rasa και REGular Expressions Πηγή: rasa.com.....	17
Εικόνα 2.9 Τα συνώνυμα στο Rasa Πηγή: rasa.com.....	18
Εικόνα 2.10 Πίνακες Lookup στο Rasa Πηγή: rasa.com .....	18
Εικόνα 2.11 Tokenizers-Lemmatizers Πηγή: rasa.com .....	19
Εικόνα 2.12 Ο Whitespace Tokenizer.....	19
Εικόνα 2.13 Οι Featurizers στο Rasa Πηγή: rasa.com .....	20
Εικόνα 2.14 Παράδειγμα Pipeline 1 Πηγή: rasa.com.....	21
Εικόνα 2.15 Παράδειγμα Pipeline 2 Πηγή: rasa.com.....	21
Εικόνα 2.16 Το Rasa x Πηγή: rasa.com .....	22
Εικόνα 3.1 Το GitHub Repository του VagBot Πηγή: github.com/vagmark30.....	23
Εικόνα 3.2 Το εργαλείο Anaconda Πηγή: anaconda.com.....	24
Εικόνα 3.3 Η βάση δεδομένων του VagBot.....	24
Εικόνα 3.4 Ο πίνακας CourseInfo .....	25
Εικόνα 3.5 Η εφαρμογή Postman.....	26
Εικόνα 3.6 Η εντολή rasa init.....	26
Εικόνα 3.7 Η πρώτη συνομιλία με το VagBot .....	27
Εικόνα 3.8 Η Δομή ενός Rasa Project.....	27
Εικόνα 3.9 Τα stories του VagBot.....	28
Εικόνα 3.10 Παράδειγμα πρόθεσης: Πρόγραμμα εξεταστικής.....	30
Εικόνα 3.11 Παράδειγμα πρόθεσης: Ποια είναι τα στοιχεία του καθηγητή .....	30
Εικόνα 3.12 Συνώνυμα καθηγητών.....	31
Εικόνα 3.13 Οι απαντήσεις και οι ενέργειες του VagBot .....	31
Εικόνα 3.14 Οι κανόνες του VagBot.....	32
Εικόνα 3.15 Αρχείο config.yml.....	33
Εικόνα 3.16 Απλό παράδειγμα.....	34
Εικόνα 3.17 Σύνθετο παράδειγμα με διάλογο.....	35
Εικόνα 3.18 Σύνθετο παράδειγμα με εξόρυξη οντότητας .....	37
Εικόνα 3.19 Σύνθετο παράδειγμα με εξόρυξη οντότητας και αναζήτηση σε βάση δεδομένων.....	38
Εικόνα 4.1 Bootstrap GitHub repository Πηγή: github.com/StartBootstrap.....	39
Εικόνα 4.2 Προκαθορισμένος κώδικας σελίδας Bootstrap. ....	40

Εικόνα 4.3 Προκαθορισμένη Bootstrap σελίδα Πηγή: <a href="https://github.com/StartBootstrap">github.com/StartBootstrap</a> .....	40
Εικόνα 4.4 VagFront πριν την υλοποίηση του VagBot.....	41
Εικόνα 4.5 Rasa webchat Repository Πηγή: <a href="https://github.com/botfront">github.com/botfront</a> .....	42
Εικόνα 4.6 Υλοποίηση του Rasa Webchat.....	43
Εικόνα 4.7 Παραμετροποίηση του Rasa Webchat .....	43
Εικόνα 4.8 Αποδεκτή μορφή αιτημάτων Πηγή: <a href="https://rasa.com">rasa.com</a> .....	44
Εικόνα 4.9 Εντολές mkdir και chmod.....	44
Εικόνα 4.10 SSH tunnel και εντολή git clone .....	45
Εικόνα 4.11 Το ngrok.....	45
Εικόνα 4.12 Αρχείο credentials.yml.....	46
Εικόνα 4.13 Έλεγχος αιτημάτων μέσω Postman 1.....	47
Εικόνα 4.14 Έλεγχος αιτημάτων μέσω Postman 2.....	48
Εικόνα 5.1 VagFront, πλατφόρμα του VagBot .....	49
Εικόνα 5.2 Παράδειγμα συνομιλιών 1 .....	50
Εικόνα 5.3 Παράδειγμα συνομιλιών 2 .....	51
Εικόνα 5.4 Παράδειγμα συνομιλιών 3 .....	52
Εικόνα 6.1 Πρώτη ενότητα ερωτηματολογίου .....	53
Εικόνα 6.2 Δεύτερη ενότητα ερωτηματολογίου.....	54

## Κατάλογος Πινάκων

Πίνακας 3.1: Προθέσεις .....	30
Πίνακας 6.1: Αριθμητικά δεδομένα .....	62

## Συντομογραφίες

TN	Τεχνητή Νοημοσύνη
MM	Μηχανική Μάθηση
AI	Artificial Intelligence
NLU	Natural Language Understanding
NLP	Natural Language Processing
ML	Machine Learning
RegEx	Regular Expressions
JSON	JavaScript Object Notation
AJAX	Asynchronous JavaScript And XML
API	Application Programming Interface



## Εισαγωγή

Στα πλαίσια της παρούσας διπλωματικής εργασίας γίνεται εισαγωγή στην έννοια και την ιστορία των chatbot. Επίσης, γίνεται αναφορά στο Rasa, Framework, το οποίο αποτελεί το εργαλείο που επιλέχθηκε για την κατασκευή του chatbot και των επιμέρους στοιχείων που συμπεριλαμβάνει. Στη συνέχεια, θα παρουσιαστεί ο τρόπος ανάπτυξης του chatbot καθώς και της front-end πλατφόρμας με την οποία θα συνομιλεί ο χρήστης.

Στόχος της παρούσας διπλωματικής εργασίας είναι η σχεδίαση και η ανάπτυξη μιας ολοκληρωμένης εμπειρίας επικοινωνίας με ένα ευφυές διαδραστικό λογισμικό συνομιλίας (chatbot).

Σκοπός είναι οι φοιτητές του τμήματος μηχανικών πληροφορικής και ηλεκτρονικών συστημάτων να μπορούν να διαδίδονται με φυσικό λόγο και να παίρνουν πληροφορίες ή/και να λύνουν θέματα σπουδών τα οποία σε διαφορετική περίπτωση θα έπρεπε να επιλύσουν με τη γραμματεία ή με τους καθηγητές του τμήματος.

Παραδοτέα της παρούσας διπλωματικής είναι:

- η παρούσα εργασία,
- ο κώδικας ανάπτυξης του ευφυούς διαδραστικού λογισμικού συνομιλίας VagBot, <https://github.com/vagmark30/VagBot>,
- ο κώδικας ανάπτυξης του VagFront, της πλατφόρμας που θα επικοινωνεί ο φοιτητής του τμήματος με το VagBot. <https://github.com/vagmark30/VagFront>.

Μια σύντομη περιγραφή των κεφαλαίων είναι:

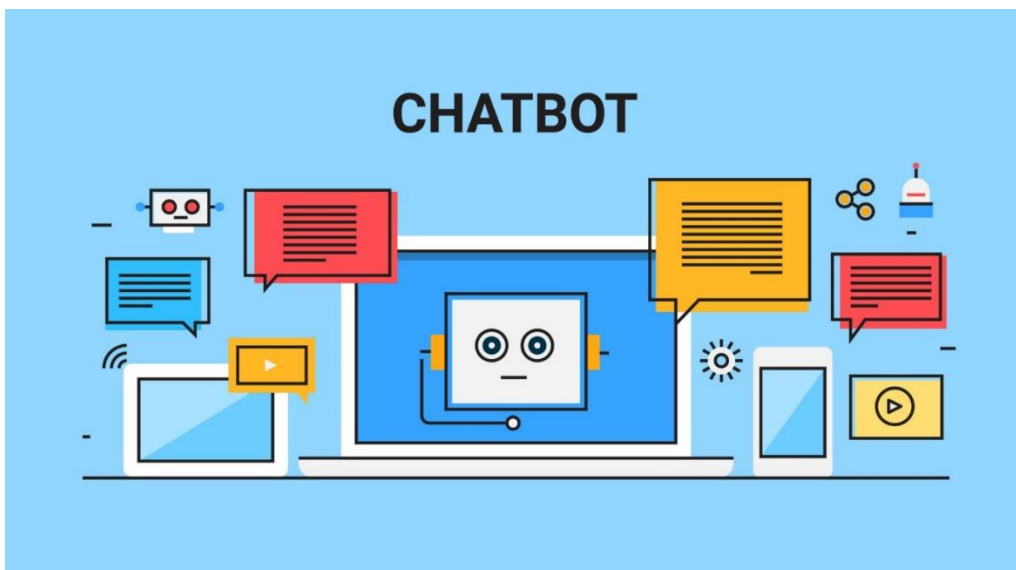
- Το κεφάλαιο 1 περιλαμβάνει την εισαγωγή και τη θεωρητική προσέγγιση στα λογισμικά ευφυούς συνομιλίας, με σκοπό την καλύτερη κατανόηση του θεωρητικού υποβάθρου. Αυτό επιτυγχάνεται μέσα από ιστορική αναδρομή και καταλήγοντας στην κατάσταση του σήμερα.
- Στο κεφάλαιο 2, συνεχίζοντας με την αποσαφήνιση εννοιών και ορισμών, παρουσιάζεται το Rasa Framework καθώς και όλα τα στοιχεία που αυτό περιλαμβάνει. Επίσης, γίνεται αναφορά τόσο στα εργαλεία όσο και στις πιθανές αιτίες των προβλημάτων που μπορεί να προκύψουν.
- Το κεφάλαιο 3 αφορά την ανάπτυξη του συγκεκριμένου chatbot, το VagBot. Στο συγκεκριμένο κεφάλαιο υπάρχει αναλυτική περιγραφή της δημιουργίας της ευφυούς συνομιλίας, ξεκινώντας από την εγκατάσταση των εργαλείων που χρησιμοποιήθηκαν και καταλήγοντας στην αναφορά συγκεκριμένων παραδειγμάτων.
- Στο κεφάλαιο 4 συμπεριλαμβάνεται η διαδικασία κατά την οποία δημιουργήθηκε μια front end πλατφόρμα με σκοπό την επίτευξη της συζήτησης χρήστη-VagBot. Η σελίδα αυτή όφειλε να είναι ενεργή, διαθέσιμη και προσβάσιμη σε όλους τους χρήστες.
- Στο κεφάλαιο 5 παρουσιάζεται, μέσω εικόνων, το ολοκληρωμένο αποτέλεσμα της εργασίας, το οποίο αποτελεί μια λειτουργική σελίδα, στην οποία οποιοσδήποτε χρήστης με σύνδεση στο διαδίκτυο μπορεί να συνομιλήσει με το VagBot.
- Στο κεφάλαιο 6 γίνεται λόγος για τη μεθοδολογία της εργασίας, η οποία συμπεριλαμβάνει τη σύνθεση ερωτηματολογίου, περιέχοντας ερωτήσεις κλειστού τύπου. Τα ερωτηματολόγια απευθύνονται κυρίως σε φοιτητές του τμήματος.
- Στο 7ο και τελευταίο κεφάλαιο εξάγονται τα συμπεράσματα της παρούσας διπλωματικής βάσει της κατασκευής του VagBot και των αρχικών ερωτημάτων. Εξετάζεται η επιτυχία ή μη, του συγκεκριμένου εργαλείου και γίνεται αναφορά σε μελλοντικές μελέτες-προτάσεις βελτίωσης.

## Κεφάλαιο 1ο: Λογισμικά ευφυούς συνομιλίας

Στο κεφάλαιο αυτό, θα καλυφθεί αρχικά το οικοσύστημα chatbot, στη συνέχεια οι ιστορικές εξελίξεις καθώς και βελτιώσεις κατά τη διάρκεια του χρόνου, και τέλος οι διάφορες ανοιχτές πλατφόρμες που υπάρχουν σήμερα, στις οποίες μπορεί ο χρήστης να κοινοποιήσει το chatbot του. Επίσης θα γίνει αναφορά στις τρέχουσες εξελίξεις του κλάδου, ο οποίος διευκολύνει τη διαθεσιμότητα των chatbots σε μεγάλη κλίμακα.

### 1.1 Τι είναι τα chatbot;

Τα chatbots είναι λογισμικά, τα οποία ανταποκρίνονται αυτόματα σε ερωτήματα από χρήστες, χωρίς την ανθρώπινη παρέμβαση [1].



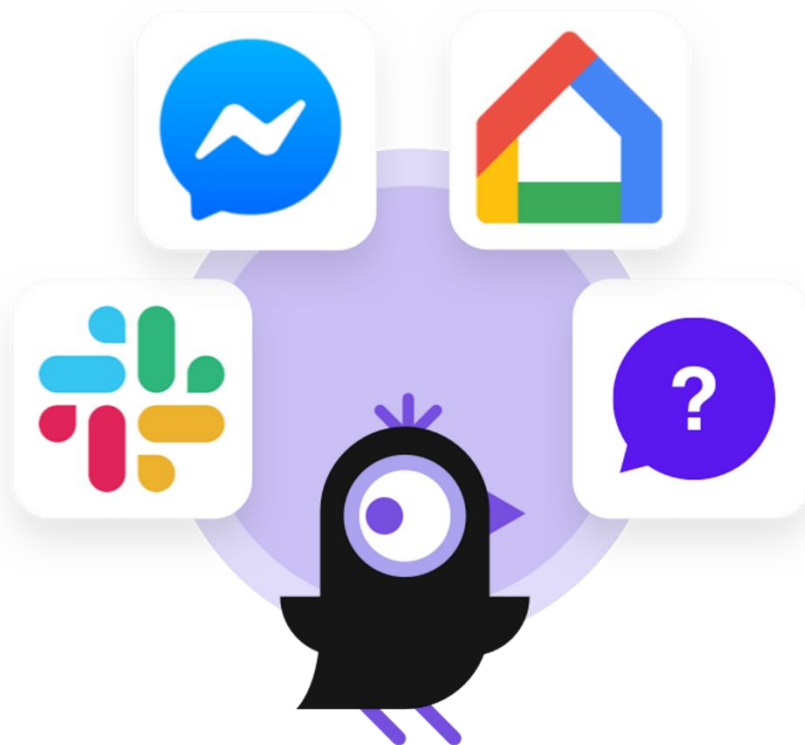
Εικόνα 1.1 Τα Chatbot και οι λειτουργίες τους  
Πηγή: kmslh.com

Ένας από τους κλασικότερους ορισμούς του chatbot χαρακτηρίζει το chatbot ως ένα πρόγραμμα, το οποίο μπορεί να αντιληφθεί τη γλώσσα ως είσοδο στη φυσική της μορφή. Στη συνέχεια, την επεξεργάζεται και παράγει σχετικές απαντήσεις, οι οποίες αποστέλλονται πίσω στο χρήστη. Αναλυτικότερα, τα chatbots τροφοδοτούνται από μηχανές, οι οποίες βασίζονται σε κανόνες, μηχανές Τεχνητής Νοημοσύνης (AI) και Μηχανικής Μάθησης (ML). Οι μηχανές αυτές αλληλεπιδρούν με τους χρήστες μέσω μιας διεπαφής βασισμένης, είτε σε γραπτό, είτε σε προφορικό λόγο. Οι μηχανές αυτές γίνεται να είναι ανεξάρτητα προγράμματα υπολογιστή, που μπορούν να συνδεθούν σε οποιαδήποτε από τις πλατφόρμες ανταλλαγής μηνυμάτων, οι οποίες είναι ανοιχτές στους προγραμματιστές μέσω API όπως το Facebook Messenger, το Slack, Skype, Microsoft Teams και ούτω καθεξής.

### 1.2 Πορεία των Chatbots

Οι τεχνολογικές εξελίξεις έχουν παρουσιάσει ραγδαία αύξηση τα τελευταία χρόνια με αποτέλεσμα εταιρείες όπως η Google, η Apple και η Amazon να πρωταγωνιστούν στην κατασκευή προσωπικών βοηθών με βάση τη φωνή. Συγκεκριμένα η Apple δημιούργησε και κυκλοφόρησε το Siri, το οποίο

διατίθεται σε iPhone, iPad, macOS. Από την άλλη, η Google, ξεκίνησε την δημιουργία του Google Home βασισμένο στο ήδη υπάρχον Google Assistant, ενώ παράλληλα η Amazon κυκλοφόρησε την Alexa. Οι παραπάνω αποτελούν φυσικές συσκευές για το σπίτι ή το γραφείο και παρέχουν βοήθεια σε πληθώρα εργασιών, όπως η αναπαραγωγή τραγουδιών από το Spotify, διαχείριση ημερολογίων και καθημερινών στόχων, έλεγχος οικιακών ηλεκτρονικών συσκευών κ.α.



Εικόνα 1.2 Διασυνδεδεμένες εφαρμογές

Λαμβάνοντας υπόψη την ιστορία των chatbots, γίνεται αντιληπτό πως η συνομιλία, ως πρώτο μέσο, υπάρχει από την εποχή που κατασκευάστηκαν οι υπολογιστές και κατέληξαν να συγκαταλέγονται στα εξέχοντα μέσα επικοινωνίας τις τελευταίες δύο δεκαετίες. Ειδικότερα, οι επιστήμονες είχαν από αρχή στόχο να δημιουργήσουν έναν υπολογιστή που θα έχει την ικανότητα να επικοινωνεί με τον άνθρωπο μέσω φυσικού τρόπου ομιλίας.

### 1.2.1 Ιστορία των Chatbots

Η ιστορία των chatbot φαίνεται να ξεκινάει από τα παλιά ακόμα χρόνια, όταν οι άνθρωποι ξεκίνησαν να αναπτύσσουν έναν τρόπο αλληλεπίδρασης με τους υπολογιστές. Παρόλο που η έννοια του chatbot θεωρείται να είναι μια πρόσφατη λέξη-κλειδί, είναι εντυπωσιακό ότι το πρώτο chatbot παρουσιάστηκε πριν ακόμη αναπτυχθεί ο πρώτος προσωπικός υπολογιστής και ονομάστηκε ELIZA [4]. Η ELIZA αναπτύχθηκε στο Εργαστήριο Τεχνητής Νοημοσύνης του MIT από τον Joseph Weizenbaum το 1966 και υποδύθηκε μια ψυχοθεραπεύτρια. Βασική της λειτουργία αποτελούσε η ανίχνευση λέξεων-κλειδιών, οι οποίες θα ενεργοποιούσαν τους αντίστοιχους κανόνες που θα επηρέαζαν την απάντηση κατά την έξοδο. Η συγκεκριμένη μεθοδολογία για δημιουργία απαντήσεων εξακολουθεί να χρησιμοποιείται ευρέως κατά τη δημιουργία chatbot.

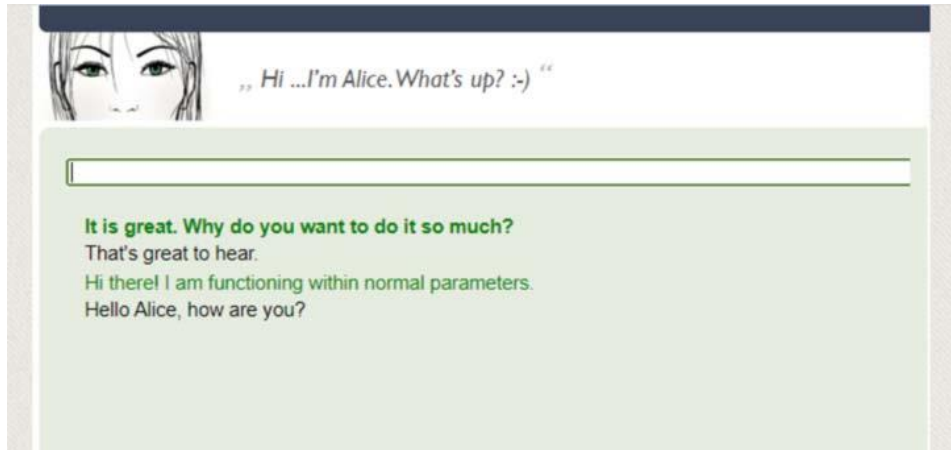


Εικόνα 1.3 ELIZA: Το πρώτο chatbot  
Πηγή: web.njit.edu

Βασικά μειονεκτήματα της ELIZA ήταν ότι η γνώση, η κατανόηση καθώς και η επικοινωνία της ήταν περιορισμένες, και ως εκ τούτου, μπορούσε να συζητήσει μόνο ένα συγκεκριμένο θέμα. Επίσης, αντιμετώπιζε αδυναμία στη συγκράτηση πληροφοριών σε μεγάλες συνομιλίες. Ωστόσο, όλες αυτές οι αστοχίες αποτέλεσαν πηγή έμπνευσης για τη μετέπειτα εξέλιξη των άλλων chatbots.

Μετά την ELIZA, ένα ακόμα chatbot, ο PARRY γράφτηκε από τον ψυχίατρο Kenneth Colby, στο Πανεπιστήμιο του Στάνφορντ, σε μια προσπάθεια προσομοίωσης ενός ατόμου με σχιζοφρένεια [3]. Το PARRY εμφανίστηκε το 1972 και θεωρείται πιο προχωρημένο από το ELIZA διότι παρουσιάζει μια καλύτερη ελεγκτική δομή καθώς είχε την ικανότητα να ορίζει τις απαντήσεις του με βάση ένα σύστημα υποθέσεων και να κάνει αναγνώριση συναισθήματος βάσει των απαντήσεων του χρήστη. Επιπροσθέτως, το PARRY χρησιμοποιήθηκε σε ένα πείραμα το 1979, όταν ψυχίατροι σε ρόλο κριτών συμμετείχαν σε μια συνομιλία, είτε με το PARRY είτε με κάποιον σχιζοφρενή, μέσω γραφομηχανής παλαιάς τεχνολογίας που είχε την ικανότητα να μεταδίδει τυπωμένα μηνύματα από σημείο σε σημείο διαμέσου ενός καναλιού επικοινωνίας. Η πρόκληση ήταν πως έπρεπε να αποφασίσουν αν ήταν ένα πρόγραμμα υπολογιστή ή ένας πραγματικός σχιζοφρενής ασθενής. Το αποτέλεσμα μετά από αυτήν την έρευνα έδειξε πως μόνο στο 48% των περιπτώσεων, οι ψυχίατροι έκριναν ορθά, πως δηλαδή συνομιλούσαν με ένα πρόγραμμα, και όχι με αληθινό άνθρωπο. Βασικά μειονεκτήματα του PARRY θεωρήθηκαν: η χαμηλή δυνατότητα κατανόησης γλώσσας, έκφρασης συναισθημάτων καθώς και η χαμηλή ταχύτητα απόκρισης στη συζήτηση.

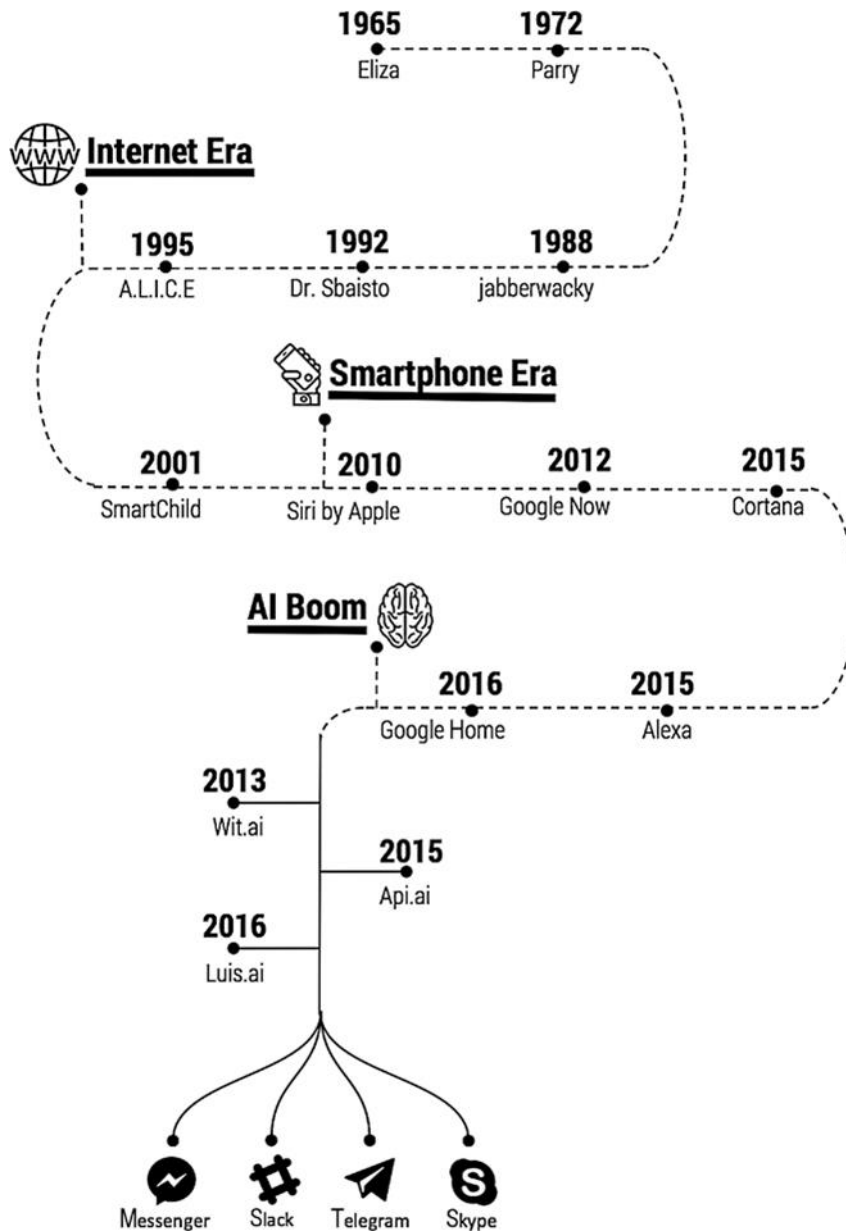




Εικόνα 1.5 Το Chatbot ALICE

Πηγή: yakbots.com

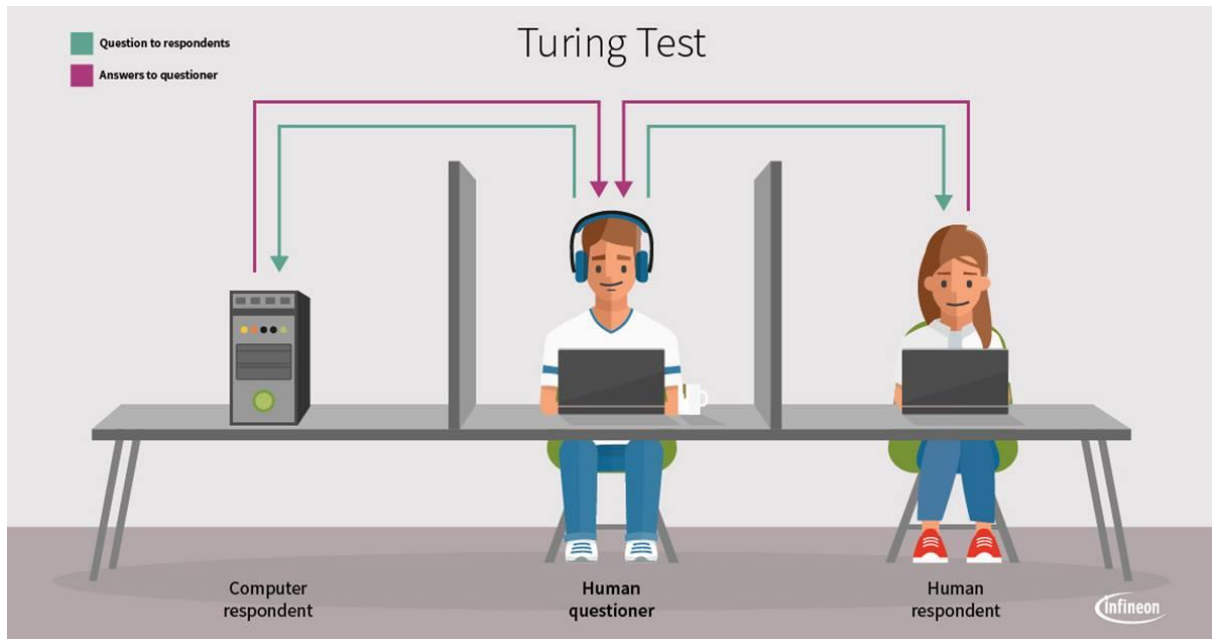
Την πρώτη δεκαετία του 21ου αιώνα, το SmarterChild κατασκευάστηκε από την ActiveBuddy [1]. Η κατασκευή αυτή αποτέλεσε την πρώτη προσπάθεια δημιουργίας ενός chatbot, το οποίο θα μπορούσε όχι μόνο να προσφέρει ψυχαγωγία αλλά και να παρέχει στο χρήστη πιο χρήσιμες πληροφορίες, όπως πληροφορίες μετοχών, αθλητικά αποτελέσματα, αποσπάσματα ταινιών κ.α. Υλοποιήθηκε μέσα στην AOL και στο Live Messenger, με περισσότερους από 30 εκατομμύρια ανθρώπους να το χρησιμοποιούν. Αργότερα όμως εξαγοράστηκε από τη Microsoft το 2007. Το SmarterChild είναι ο πρόδρομος του Siri της Apple, του Bixby της Samsung, της Alexa της Amazon και του Assistant της Google. Η Siri είναι ένας έξυπνος προσωπικός βοηθός που αναπτύχθηκε ως δευτερεύον έργο από την SRI International και αργότερα υιοθετήθηκε από την Apple στο iOS 5 για iPhone. Θεωρείται αναπόσπαστο μέρος του οικοσυστήματος iOS. Το Siri επιτρέπει στους χρήστες να συμμετέχουν σε τυχαίες συνομιλίες ενώ παράλληλα έχει την ικανότητα να παρέχει χρήσιμες πληροφορίες σχετικά με τον καιρό, τα αποθέματα και τα εισιτήρια κινηματογράφου. Εταιρίες τεράστιου βεληνεκούς, όπως η Samsung και η Google έχουν επίσης ακολουθήσει τα βήματα της Apple με την ανάπτυξη των δικών τους βοηθών AI, Bixby και Google Assistant, αντίστοιχα. Υπάρχουν ακόμη, σπιτικοί βοηθοί που λειτουργούν με φωνή όπως το Amazon Alexa και το Google Home, τα οποία είναι μια άλλη αναπαράσταση των chatbot.



Εικόνα 1.6 Η πορεία των Chatbot  
 Πηγή: Building better chatbots 2018.

### 1.2.2 Δοκιμασία Turing

Turing Test είναι ένα τεστ νοημοσύνης για υπολογιστές όπου ένας άνθρωπος (αποστολέας) δεν θα πρέπει να μπορεί να διακρίνει μεταξύ μηχανής (δέκτης) ή άλλου ανθρώπου (δέκτης) όταν οι απαντήσεις και από τους δύο παρουσιάζονται στον αποστολέα [7]. Το τεστ Turing σχεδιάστηκε από τον Alan Turing το 1950 στην εργασία του «Computing Machinery and Intelligence» ενώ εργαζόταν στο πανεπιστήμιο του Μάντσεστερ.



Εικόνα 1.7 Η δοκιμασία Turing  
 Πηγή: infineon.com

### 1.2.3 Βραβείο Loebner

Το βραβείο Loebner ήταν ένας ετήσιος διαγωνισμός τεχνητής νοημοσύνης που απονέμει βραβεία στα προγράμματα υπολογιστών που θεωρούν οι κριτές ως τα πιο ανθρώπινα [8]. Το βραβείο αναφέρεται ότι δεν λειτουργεί από το 2020. Η μορφή του διαγωνισμού ήταν αυτή μιας τυπικής δοκιμής Turing.

### 1.2.4 Πρόσφατες εξελίξεις των Chatbots

Εξετάζοντας την ιστορική εξέλιξη, οι εταιρείες ανέκαθεν κατασκεύαζαν τη δική τους ατομική τεχνητή νοημοσύνη chatbot για να εξυπηρετήσουν το σκοπό των χρηστών τους. Τα τελευταία χρόνια, η τάση αυτή έχει αλλάξει, με την Telegram να ανοίγει την πλατφόρμα της, τον Ιούνιο του 2015, επιτρέποντας έτσι στους προγραμματιστές να κάνουν chatbot που εξυπηρετούν τους χρήστες με πολλές υπηρεσίες όπως δημοσκοπήσεις, ειδήσεις, παιχνίδια και ψυχαγωγία. Επιπλέον, τον Δεκέμβριο του 2015, η Slack, μια εταιρεία ομαδικής συνεργασίας μεταξύ πλατφορμών, ανακοίνωσε πως εισάγει τις δυνατότητες συνομιλίας chatbot με χρήστες. Η Slack συνέβαλε σημαντικά στην ώθηση άλλων εταιρειών να αρχίσουν να επενδύουν σε αυτή τη νέα τάση των chatbot. Ένας από τους σημαντικότερους συντελεστές στην ιστορία των chatbot ήταν το Facebook. Παρόλο που εμφανίστηκε αργοπορημένα, είχε το μεγαλύτερο αντίκτυπο στα chatbot. Βασική είναι κι η αναφορά άλλων σημαντικών παραγόντων όπως το Skype, το Kik και το WeChat, οι οποίοι έχουν αναπτύξει λειτουργίες οι οποίες επιτρέπουν σε προγραμματιστές να δημοσιεύουν chatbot.

Τέλος, λαμβάνοντας υπόψη την εξέλιξη των chatbots από τη δεκαετία του 1960 έως σήμερα, γίνεται αντιληπτή η σημαντικότητα του chatbot. Παρόλο που κάποτε η επικοινωνία με εικονικό και μη ζωντανό δέκτη θεωρούνταν ακατόρθωτη, η ύπαρξή της πλέον είναι εμφανής στην καθημερινότητα.

### 1.3 Άνοδος των Chatbot

Σε αυτή την ενότητα, θα γίνει εξέταση των παραγόντων που προώθησαν την πρόσφατη άνοδο των chatbots. Η δημοτικότητα των chatbots έχει ραγδαία αύξηση και μεγάλο μέρος ατόμων θεωρεί πως το γεγονός αυτό οφείλεται στη διαφημιστική εκστρατεία AI που δημιουργήθηκε από το Facebook, ανοίγοντας την πλατφόρμα του Messenger για τους προγραμματιστές, με σκοπό την κατασκευή chatbots [1][9]. Μπορεί να φαίνεται ότι τα chatbot έγιναν γνωστά σε πολύ σύντομο χρονικό διάστημα, αλλά στην πραγματικότητα, αποτελούν ένα συνδυασμό διαφόρων παραγόντων που συνέβησαν από τις αρχές της δεκαετίας του 2000 έως τώρα. Η σημαντικότητα των chatbots διακρίνεται στο γεγονός πως μεγάλο πλήθος ερωτημάτων υποστήριξης πελατών δεν απαιτεί κάποιο είδος ανθρώπινης παρέμβασης και μπορεί να διεκπεραιωθεί εξ ολοκλήρου από συστήματα τεχνητής νοημοσύνης. Παρακάτω θα αναλυθούν περισσότεροι λόγοι που συντέλεσαν στην άνοδο των chatbot.

#### 1.3.1 Αύξηση των χρηστών του Διαδικτύου.

Η χρήση του Διαδικτύου το 2000 ήταν 300 εκατομμύρια. ενώ το 2017 η αύξηση ήταν τόσο μεγάλη που έφερε ως αποτέλεσμα οι χρήστες να φτάσουν τα 5,1 δισεκατομμύρια άτομα[1][4][23]. Επομένως, η αύξηση της χρήσης του Διαδικτύου ανέρχεται σε ένα πολύ μεγάλο ποσοστό, και καθώς περισσότεροι άνθρωποι συνδέονται στο διαδίκτυο καθημερινά, η δύναμη του Διαδικτύου αυξάνεται. Εντυπωσιακή αύξηση παρατηρείται όχι μόνο στον αριθμό των ατόμων που χρησιμοποιούν το Διαδίκτυο, αλλά και στο χρόνο που αφιερώνουν σε αυτό. Η ενασχόληση των ενηλίκων στο Διαδίκτυο είναι κατά μέσο όρο περίπου 28 ώρες την εβδομάδα για συλλογή πληροφοριών, συζήτηση με φίλους μέσω των μέσων κοινωνικής δικτύωσης ή απλώς κατανάλωση περιεχομένου πολυμέσων. Συνεπώς, με την αύξηση της χρήσης και του αριθμού των ανθρώπων, το Διαδίκτυο υπολογίζεται ότι έχει δημιουργήσει περίπου 1,2 εκατομμύρια terabyte δεδομένων (1 terabyte είναι 1.000 gigabytes). Το έτος 2007 σηματοδότησε την εμφάνιση των Big Data, γεγονός που σημαίνει ότι υπάρχει πολλή πληροφορία, η οποία μπορεί να εξορυχθεί με τη σωστή διαχείριση δεδομένων [10]. Ταυτόχρονα, τα εργαλεία για να γίνει αυτό εξακολουθούν να υπάρχουν καθώς αναπτύσσονται ενεργά από μεγάλες επιχειρήσεις σε όλο τον κόσμο. Για τη δημιουργία ενός επιτυχημένου chatbot, αποτελεί προϋπόθεση η δυνατότητα πρόσβασης σε δεδομένα που μπορούν να επεξεργαστούν, έτσι ώστε να είναι δυνατή η απάντηση σε ερωτήματα που τέθηκαν από χρήστες. Επίσης, ένα από τα βασικά συστατικά για ένα έξυπνο chatbot αποτελεί η δυνατότητα πρόσβασης πολλαπλών χρηστών. Υπάρχουν πολλές πλατφόρμες στο Διαδίκτυο που μπορούν να υποστηρίξουν τέτοιους αριθμούς. Μια από αυτές αποτελεί και το Facebook, που μέσα σε ένα μήνα χρησιμοποιήθηκε πάνω από 1,7 δισεκατομμύρια ανθρώπους και γρήγορα αποδείχθηκαν οι δυνατότητες που παρέχει για επαγγελματικά μηνύματα μέσω των chatbots.

#### 1.3.2 Πρόδος στην Τεχνολογία

Όλα τα δεδομένα που παράγονται καθημερινά από τους χρήστες του Διαδικτύου δεν αποδεικνύονται τόσο χρήσιμα εάν δεν υπάρχουν διαθέσιμα εργαλεία για τη αξιοποίηση των δεδομένων για μαθησιακούς σκοπούς [1][4]. Τη δεκαετία του 2000, το πεδίο της μηχανικής μάθησης εξελίχθηκε, με την προσθήκη της βαθιάς μάθησης, η οποία βοηθά τις μηχανές υπολογιστών να «βλέπουν» και να κατανοούν κείμενο, εικόνες, ήχο, βίντεο κ.α. Η μετάβαση των θεωρητικών προβλημάτων μηχανικής μάθησης στην πρακτική εφαρμογή, έχει βοηθήσει τις εταιρείες και οργανισμούς να αξιοποιήσουν τη μηχανική μάθηση για να αναπτυχθούν. Οι κορυφαίες εταιρείες τεχνολογίας στον κόσμο συνέβαλαν όλες στη δημιουργία των αλγορίθμων μηχανικής μάθησης, οι οποίοι είναι διαθέσιμοι και ανοιχτοί για χρήση και δημιουργία

συναρπαστικών εφαρμογών. Το TensorFlow (open-source) της Google, ως λογισμικό και υπηρεσία cloud, ήταν ορόσημο στη μηχανική μάθηση καθώς παρείχε τη δύναμη της μηχανικής μάθησης προς αξιοποίηση από οποιονδήποτε έχει βασική κατανόηση του προγραμματισμού. Άλλες εταιρείες έχουν πείσει προς την ίδια κατεύθυνση για να κάνουν τη μηχανική μάθηση διαθέσιμη σε όλους. Για παράδειγμα, η Microsoft Azure κυκλοφόρησε μια πλατφόρμα μηχανικής μάθησης στο cloud της και η Amazon πρόσθεσε μοντέλα μηχανικής μάθησης στις υπηρεσίες cloud της, AWS. Το Netflix ξεκίνησε, κάνοντας τους προγραμματιστές να ανταγωνίζονται μεταξύ τους, δημιουργώντας μοντέλα που προσφέρουν καλύτερη απόδοση από τους ήδη υπάρχοντες αλγόριθμους Netflix για προτάσεις ταινιών. Ο Kaggle πήρε την ιδέα από Netflix και μετατράπηκε σε μια πλατφόρμα μηχανικής μάθησης για ανερχόμενους προγραμματιστές, προκειμένου να διαγωνιστούν σε υπάρχοντα μεγάλα σύνολα δεδομένων και να δημιουργήσουν ισχυρά μοντέλα.

### 1.3.3 Οικοσύστημα προγραμματιστών

Στη σημερινή εποχή σημειώνεται σημαντική αύξηση των απαιτήσεων για την εύρεση των θέσεων εργασίας στο χώρο της πληροφορικής [1][4]. Αξίζει να σημειωθεί η ύπαρξη μεγάλης και ενεργής κοινότητας προγραμματιστών, η οποία εκτός από το γεγονός ότι αυξάνεται κι αυτή με τη σειρά της συνεχώς με εκθετικό ρυθμό, αντιμετωπίζει επίσης πίεση από το οικοσύστημα των open-source λογισμικών να αναπτυχθεί και να βελτιώσει υπάρχοντα εργαλεία και framework. Οι εξελίξεις και η εύκολη πρόσβαση σε εργαλεία και framework οδήγησαν στη ραγδαία ανάπτυξη εφαρμογών, μέσα από τις οποίες δόθηκαν οι δυνατότητες να δοκιμαστούν νέες ιδέες. Τα API επιτρέπουν σε μια εφαρμογή να εξάγει πληροφορίες από ένα κομμάτι λογισμικού και να χρησιμοποιήσει αυτές τις πληροφορίες σε μια άλλη εφαρμογή, ή ορισμένες φορές για ανάλυση δεδομένων. Το ανοιχτό οικοσύστημα των API έχει εξελιχθεί τόσο πολύ την τελευταία δεκαετία, που σήμερα είναι πολύ πιθανό να αποκτήσει κανείς πρόσβαση σε ένα API και τα δεδομένα του, για οποιαδήποτε εφαρμογή μπορεί τον ενδιαφέρει. Έτσι, έχοντας οι προγραμματιστές open-source λογισμικά και ανοιχτά API για άντληση δεδομένων, είναι πλέον σε θέση να δημιουργήσουν chatbot που κατανοούν τη φυσική γλώσσα με μεγάλη ευκολία. Μόλις ένα chatbot καταλάβει τι έχει πει ο χρήστης, ανακτά τις απαιτούμενες πληροφορίες με την επίκληση ενός API ή κάνοντας αναζήτηση στη βάση δεδομένων.

### 1.3.4 Πλατφόρμες ανταλλαγής μηνυμάτων

Μία από τις πρωταρχικές εταιρείες που έδωσαν πρόσβαση στην εφαρμογή τους στους προγραμματιστές, παρέχοντας τους τη δυνατότητα να αναπτύξουν τα δικά τους chatbot, ήταν η Telegram. Η πρόσβαση αυτή είχε πολύ σημαντική απήχηση στην γενικότερη εξέλιξη των chatbot, διότι προκάλεσε την αντίδραση από δύο άλλες εταιρείες κολοσσούς, του Facebook και του Slack. Επομένως, το Facebook ανακοίνωσε την εισαγωγή των chatbots στην πλατφόρμα της, ενέργεια η οποία προσέλκυσε το ενδιαφέρον προγραμματιστών σε όλο τον κόσμο. Όλες αυτές οι δημοφιλείς πλατφόρμες ανταλλαγής μηνυμάτων αποτελούν παροχή μιας τεράστιας βάσης δεδομένων, την οποία οι προγραμματιστές μπορούν να αξιοποιήσουν και να δημιουργήσουν καλύτερους και πιο εξελιγμένους πράκτορες συνομιλίας.

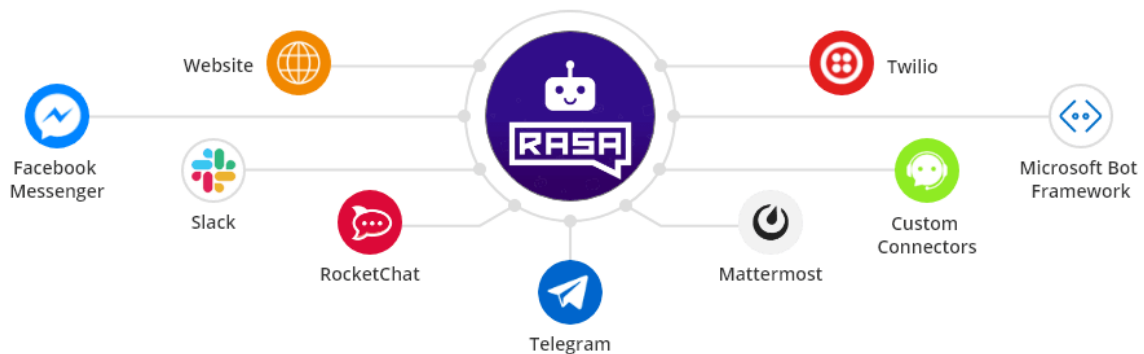
## Κεφάλαιο 2ο: Rasa Framework

### 2.1 Τι είναι το Rasa;

Το Rasa είναι ένα πλαίσιο (framework) μηχανικής μάθησης ανοιχτού κώδικα (open-source) για αυτοματοποιημένες συνομιλίες κειμένου[11]. Είναι το framework που επιλέχθηκε διότι:

- κατανοεί μηνύματα, ταξινομεί προθέσεις, αποτυπώνει το πλαίσιο συζήτησης,
- υποστηρίζει πολλές γλώσσες, εξ αυτών και τα Ελληνικά ,
- είναι εφαρμογή ανοιχτού κώδικα (open-source),
- συνδέεται μέσω API σε πλατφόρμες ανταλλαγής μηνυμάτων,
- υποστηρίζεται και επιλέγεται από μεγάλη κοινότητα προγραμματιστών, πάνω απο10 εκατομμύρια λήψεις από την κυκλοφορία,
- υλοποιείται σε αρθρωτή (modular), επεκτάσιμη αρχιτεκτονική.

Επίσης, αποτελείται από επιμέρους στοιχεία, το Rasa Core, το Rasa NLU, τα οποία πλέον έχουν συγχωνευτεί, και από το Rasa X [24]. Κάθε μέρος εξυπηρετεί ένα συγκεκριμένο και σημαντικό σκοπό. Η κατανόηση των εισαχθέντων δεδομένων, δηλαδή της φυσικής γλώσσας, γίνεται μέσω του Rasa NLU μιας διαδικασίας αλληλεπίδρασης με σαφήνεια και λογική με το χρήστη, καθίσταται το πρώτο και σημαντικότερο στάδιο προκειμένου να λειτουργήσει σωστά το chatbot. Έπειτα, η μηχανή διαλόγου του Rasa μηχανή διαλόγου αποφασίζει τι έπεται σε μια συνομιλία με βάση το περιεχόμενο των μηνυμάτων.



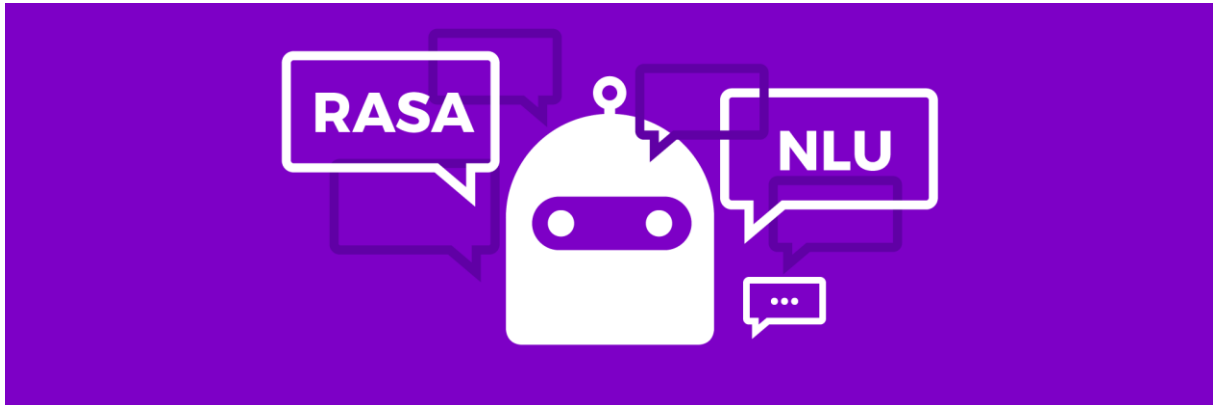
Εικόνα 2.1 Το Rasa Framework

Πηγή: rasa.com

Όσον αφορά μέρος της βιβλιοθήκης ανοιχτού κώδικα Rasa, γίνεται λόγος για μια διαλογική μηχανή, η οποία αποτελεί το κεντρικό μέρος του chatbot. Ειδικότερα, χρησιμοποιεί ένα μοντέλο Μηχανικής Μάθησης, το οποίο βάσει του μοντέλου που έχει εκπαιδευτεί, κατευθύνει το διάλογο. Παρόλο που δεν είναι αναγκαίο για να παραχθεί διαλογικός πράκτορας, το Rasa X, έχει τη δυνατότητα να προσφέρει παραπάνω δυνατότητες, αλλά και διευκολύνσεις, με αποτέλεσμα να αυξηθεί τόσο η ευχρηστία όσο και η αποδοτικότητα της παραγωγής του chatbot.

## 2.2 Rasa NLU

Ο στόχος του NLU (Natural Language Understanding) είναι η εξαγωγή δομημένων πληροφοριών από τα μηνύματα των χρηστών [24]. Αυτό συνήθως περιλαμβάνει την πρόθεση του χρήστη και τυχόν οντότητες που περιέχει το μήνυμά του. Οι επιπλέον πληροφορίες μπορούν να προστεθούν, όπως κανονικές εκφράσεις (Regular Expressions) και πίνακες αναζήτησης (Lookup tables) στα δεδομένα εκπαίδευσης, έτσι ώστε να γίνει ευκολότερη η διαδικασία αναγνώρισης των προθέσεων και των οντοτήτων από το μοντέλο.



Εικόνα 2.2 Το Rasa NLU  
Πηγή: rasa.com

Το Rasa Open Source χρησιμοποιεί αρχεία μορφής YAML ως έναν ενοποιημένο και επεκτάσιμο τρόπο διαχείρισης όλων των δεδομένων εκπαίδευσης, συμπεριλαμβανομένων των δεδομένων NLU, των ιστοριών και των κανόνων. Τα δεδομένα εκπαίδευσης μπορούν να χωριστούν σε οποιονδήποτε αριθμό αρχείων YAML και κάθε αρχείο μπορεί να περιέχει οποιονδήποτε συνδυασμό δεδομένων NLU, ιστοριών και κανόνων

### 2.2.1 Κορμός Rasa

#### 2.2.1.1 Stories

Οι ιστορίες είναι ένας τύπος δεδομένων εκπαίδευσης, οι οποίες χρησιμοποιούνται για την εκπαίδευση του μοντέλου διαχείρισης διαλόγου του chatbot [24]. Οι ιστορίες μπορούν να χρησιμοποιηθούν για την εκπαίδευση των μοντέλων, τα οποία θα είναι σε θέση να γενικεύουν σε συνομιλίες που δεν υπάρχουν στα δεδομένα εκπαίδευσης.

```

- story: happy path
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_great
  - action: utter_happy

- story: sad path 1
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_unhappy
  - action: utter_cheer_up
  - action: utter_did_that_help
  - intent: affirm
  - action: utter_happy

- story: sad path 2
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_unhappy
  - action: utter_cheer_up
  - action: utter_did_that_help
  - intent: deny
  - action: utter_goodbye

```

Εικόνα 2.3 Rasa Stories  
Πηγή: rasa.com

Μια ιστορία είναι μια αναπαράσταση μιας συνομιλίας μεταξύ ενός χρήστη και ενός βοηθού τεχνητής νοημοσύνης, η οποία μετατρέπεται σε μια συγκεκριμένη μορφή όπου τα στοιχεία του χρήστη εκφράζονται ως προθέσεις (και οντότητες όταν είναι απαραίτητο), ενώ οι απαντήσεις και οι ενέργειες του βοηθού εκφράζονται ως ονόματα ενεργειών.

### 2.2.1.2 Responses

Προκειμένου το chatbot να μπορεί να δίνει απαντήσεις στον τελικό χρήστη, θα πρέπει αυτές να έχουν ορισθεί στο domain, εντός του Rasa Core [24]. Οι απαντήσεις αυτές μπορούν να είναι προκαθορισμένα κείμενα, τα οποία έχει γράψει ο developer στο chatbot, αλλά και “custom” απαντήσεις, των οποίων το περιεχόμενο προκύπτει από την ίδια την συζήτηση, με πολλές πιθανές παραλλαγές.

### 2.2.1.3 Actions

Όλες οι ενέργειες που εκτελούνται από το bot, συμπεριλαμβανομένων των απαντήσεων, παρατίθενται σε ιστορίες κάτω από το λέξη κλειδί action [24]. Ο χρήστης μπορεί να χρησιμοποιήσει μια απάντηση από το αρχείο Domain ως action αναφέροντάς την σε ένα story. Ομοίως, μπορεί να υποδείξει ότι μια ιστορία πρέπει να καλεί μια custom action, για την οποία έχει αναπτυχθεί προσαρμοσμένος κώδικας που θα διαχειρίζεται μια ερώτηση ενός χρήστη. Βασική προϋπόθεση αποτελεί να συμπεριληφθεί το όνομα της custom action στη λίστα actions του αρχείου Domain.

### 2.2.1.4 Rules

Οι κανόνες είναι ένας τύπος δεδομένων εκπαίδευσης που χρησιμοποιούνται για την εκπαίδευση του μοντέλου διαχείρισης διαλόγου του βοηθού του χρήστη [24]. Οι κανόνες περιγράφουν σύντομα κομμάτια συνομιλιών που πρέπει πάντα να ακολουθούν την ίδια διαδρομή.

```

- rule: Say goodbye anytime the user says goodbye
  steps:
  - intent: goodbye
  - action: utter_goodbye

- rule: Say 'I am a bot' anytime the user challenges
  steps:
  - intent: bot_challenge
  - action: utter_iamabot

```

Εικόνα 2.4 Rasa Rules  
Πηγή: rasa.com

### 2.2.1.5 Domain

Ο τομέας ορίζει τις λειτουργίες και καθορίζει τις προθέσεις, τις οντότητες, τις υποδοχές, τις απαντήσεις, και τις ενέργειες που πρέπει να γνωρίζει ένα chatbot [24]. Επίσης, συντελεί στη διαμόρφωση της ροής μιας συνομιλίας με το χρήστη. Ο τομέας χρησιμοποιεί την ίδια μορφή YAML με τα δεδομένα εκπαίδευσης και μπορεί να χωριστεί σε πολλά αρχεία ή να συνδυαστεί σε ένα. Τέλος, περιλαμβάνει τους ορισμούς για κάθε είδους απάντηση, είτε προσαρμοσμένης είτε απλής.

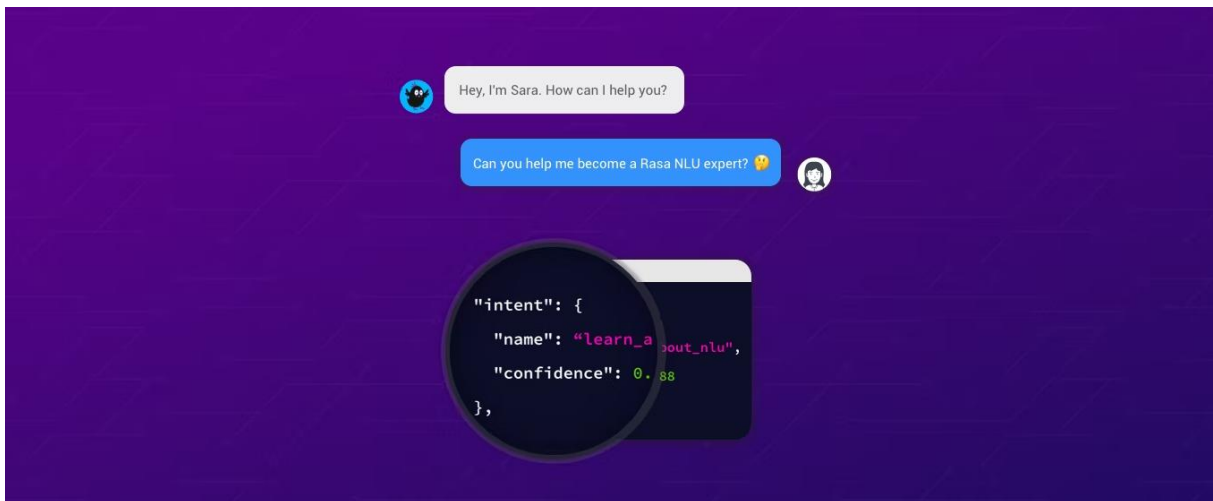
### 2.2.1.6 Config

Το αρχείο διαμόρφωσης ορίζει τα στοιχεία και τις πολιτικές που θα χρησιμοποιήσει το μοντέλο του χρήστη για να κάνει προβλέψεις με βάση τα δεδομένα του [24].

## 2.2.2 Μηχανισμοί και εργαλεία

### 2.2.2.1 Ταξινομητής πρόθεσης (Intent Classifier)

Το Rasa NLU παρέχει την πλήρη δυνατότητα προσαρμογής επεξεργάζοντας τα μηνύματα χρήστη σε μια λεγόμενη διοχέτευση (pipeline) [13][16]. Συγκεκριμένα, μια διοχέτευση ορίζει διαφορετικά στοιχεία, τα οποία επεξεργάζονται ένα μήνυμα χρήστη διαδοχικά και τελικά οδηγούν στην ταξινόμηση των μηνυμάτων χρήστη σε προθέσεις και στην εξαγωγή οντοτήτων.



Εικόνα 2.5 Rasa και εκτίμηση πρόθεσης  
Πηγή: rasa.com

Το Rasa χρησιμοποιεί την έννοια των προθέσεων με σκοπό την περιγραφή του τρόπου κατηγοριοποίησης των μηνυμάτων χρήστη. Το Rasa NLU ταξινομεί τα μηνύματα χρήστη σε μία ή σε πολλαπλές προθέσεις χρήστη.

Ο ταξινομητής του chatbot που υλοποιήθηκε στην παρούσα εργασία χρησιμοποιεί τη βιβλιοθήκη Spacy, ώστε να φορτώσει προ εκπαιδευμένα μοντέλα γλώσσας, τα οποία χρησιμοποιούνται για την αναπαράσταση κάθε λέξης στο μήνυμα χρήστη ως word\_embeddings (ενσωματώσεις λέξεων). Τα word\_embeddings αποτελούν διανυσματικές αναπαραστάσεις λέξεων. Επεξηγηματικά, παρόμοιες λέξεις οφείλουν να αντιπροσωπεύονται από παρόμοια διανύσματα. Οι ενσωματώσεις λέξεων είναι συγκεκριμένες για τη γλώσσα στην οποία εκπαιδεύτηκαν. Ως εκ τούτου, πρέπει ανάλογα με τη γλώσσα που χρησιμοποιείται να επιλεγθούν διαφορετικά μοντέλα.

Το Rasa NLU παίρνει τον μέσο όρο όλων των ενσωματώσεων λέξεων σε ένα μήνυμα και στη συνέχεια, εκτελεί μια αναζήτηση πλέγματος για να βρει τις καλύτερες παραμέτρους για τον ταξινομητή διανυσμάτων υποστήριξης που ταξινομεί τις κατά μέσο όρο ενσωματώσεις στις διαφορετικές προθέσεις. Η αναζήτηση πλέγματος εκπαιδεύει πολλαπλούς ταξινομητές διανυσμάτων υποστήριξης (Support Vectors) με διαφορετικές διαμορφώσεις παραμέτρων και στη συνέχεια επιλέγει την καλύτερη διαμόρφωση με βάση τα αποτελέσματα της δοκιμής.

Αυτός ο ταξινομητής είναι αναγκαίο να χρησιμοποιηθεί ακόμα και αν ο χρήστης έχει μικρές ποσότητες δεδομένων εκπαίδευσης. Αν και αυτό είναι σύνηθες στο συγκεκριμένο σημείο, ο χρήστης θα λάβει τα μέγιστα δυνατά αποτελέσματα ταξινόμησης. Λαμβάνοντας υπόψη πως η εκπαίδευση δεν ξεκινά από το μηδέν, η διαδικασία θα είναι γρήγορη, γεγονός που δίνει σύντομους χρόνους επανάληψης.

Από την άλλη πλευρά, οι καλές ενσωματώσεις λέξεων δεν είναι διαθέσιμες για όλες τις γλώσσες, καθώς εκπαιδεύονται κυρίως σε δημόσια διαθέσιμα σύνολα δεδομένων που είναι κυρίως αγγλικά. Επίσης, δεν καλύπτουν συγκεκριμένες λέξεις, όπως ονόματα προϊόντων ή ακρωνύμια. Σε αυτήν την περίπτωση, συνίσταται η εκπαίδευση προσαρμοσμένων ενσωματώσεων. Καθώς αυτός ο ταξινομητής εκπαιδεύει τις ενσωματώσεις λέξεων από την αρχή, χρειάζεται περισσότερο δεδομένα εκπαίδευσης από τον ταξινομητή που χρησιμοποιεί προεκπαιδευμένες ενσωματώσεις με σκοπό την καλύτερη γενίκευση. Καθώς εκπαιδεύεται στα δεδομένα του χρήστη, προσαρμόζεται στα ειδικά μηνύματα του τομέα. Επίσης, είναι εγγενώς ανεξάρτητο από τη γλώσσα και δεν βασίζεται σε καλές ενσωματώσεις λέξεων για μια συγκεκριμένη γλώσσα. Το εξαιρετικό χαρακτηριστικό αυτού του ταξινομητή είναι ότι υποστηρίζει μηνύματα με πολλαπλές προθέσεις γεγονός που τον καθιστά πολύ ευέλικτο ταξινομητή για περιπτώσεις προηγμένης χρήσης.

Πρέπει να σημειωθεί ότι σε ορισμένες γλώσσες (π.χ. κινέζικα) δεν είναι δυνατή η χρήση της προεπιλεγμένης προσέγγισης του Rasa NLU για να χωρίσει ο χρήστης τις προτάσεις σε λέξεις χρησιμοποιώντας κενά ως διαχωριστικό. Σε αυτή την περίπτωση πρέπει να χρησιμοποιηθεί ένα διαφορετικό στοιχείο tokenizer. Στη συνέχεια, θα παρουσιαστούν μερικά από τα πιο συνηθισμένα προβλήματα που πρέπει να αντιμετωπιστούν προκειμένου να επιτευχθεί η ταξινόμηση προθέσεων

#### **2.2.2.1.1 Πρόβλημα έλλειψης δεδομένων εκπαίδευσης**

Όσο μεγαλύτερη είναι η χρήση των chatbot από τους χρήστες, τόσο περισσότερα δεδομένα συνομιλίας θα υπάρχουν για να επιλεχθούν τα παραδείγματα εκπαίδευσης [16]. Στην αρχή, ο χρήστης έχει ελάχιστα έως καθόλου δεδομένα εκπαίδευσης και η ακρίβεια των ταξινομήσεων των προθέσεων του, είναι χαμηλή. Μια συχνά χρησιμοποιούμενη προσέγγιση για την αντιμετώπιση αυτού του ζητήματος είναι η χρήση του εργαλείου δημιουργίας δεδομένων chatito, που αναπτύχθηκε από τον Rodrigo Pimentel. Η δημιουργία προτάσεων από προκαθορισμένα τμήματα λέξεων, μπορεί να δώσει γρήγορα ένα μεγάλο σύνολο δεδομένων. Είναι σημαντικό ο χρήστης να αποφύγει την χρήση εργαλείων παραγωγής δεδομένων σε υπερβολικό βαθμό, καθώς το μοντέλο κινδυνεύει να κάνει υπερπροσαρμογή στα δεδομένα εκπαίδευσης και να μην γενικεύει (overfitting).

#### **2.2.2.1.2 Πρόβλημα λέξεων εκτός λεξιλογίου**

Αναπόφευκτα, θα χρησιμοποιηθούν λέξεις, τις οποίες το εκπαιδευμένο μοντέλο δεν τις έχει ενσωματωμένες π.χ. κάνοντας τυπογραφικά λάθη ή απλά χρησιμοποιώντας λέξεις που δεν έχουν συμπεριληφθεί από τον δημιουργό του chatbot [16].

### 2.2.2.1.3 Πρόβλημα παρόμοιων προθέσεων

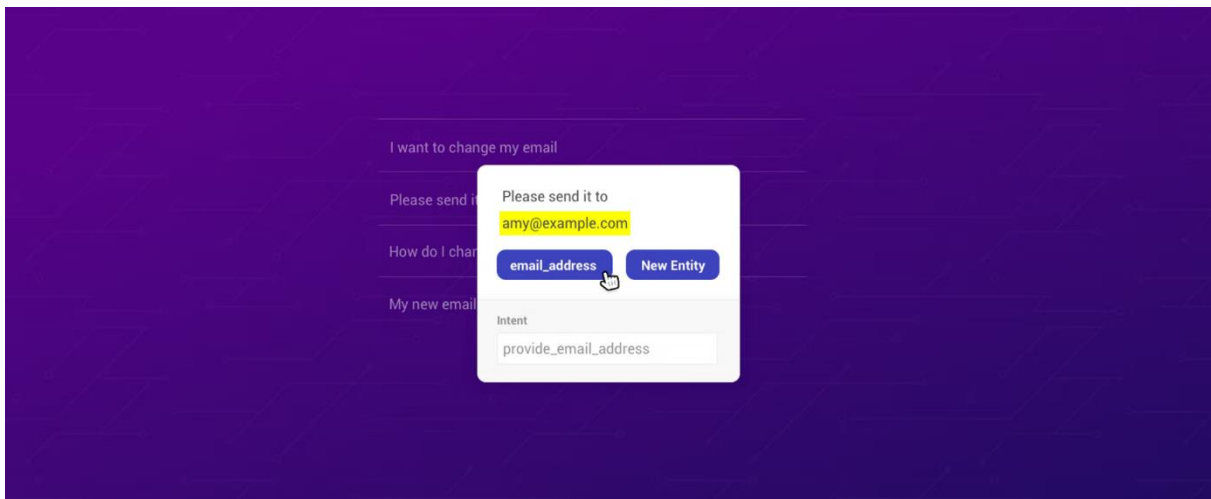
Στην περίπτωση που πρόκειται για πανομοιότυπες προθέσεις, αυξάνεται η δυσκολία διαχωρισμού και αναγνώρισης. Το δεδομένο αυτό θεωρείται προφανές, ωστόσο συχνά παραβλέπεται κατά τη δημιουργία προθέσεων [16].

### 2.2.2.1.4 Πρόβλημα αλλοιωμένων δεδομένων

Στόχο πρέπει να αποτελεί πάντα η διατήρηση μιας προσέγγισης ισορροπίας του αριθμού των παραδειγμάτων ανά πρόθεση [16]. Ωστόσο, μερικές φορές οι προθέσεις μπορεί να ξεπεράσουν τα παραδείγματα εκπαίδευσης άλλων προθέσεων. Ενώ γενικά περισσότερα δεδομένα βοηθούν στην επίτευξη καλύτερης ακρίβειας, η ανισορροπία στο πλήθος των παραδειγμάτων μπορεί να οδηγήσει σε έναν προκατειλημμένο ταξινομητή που με τη σειρά του επηρεάζει αρνητικά την απόδοση του chatbot.

### 2.2.2.2 Εξαγωγή οντοτήτων

Ως πλαίσιο ανοιχτού κώδικα, το Rasa NLU δίνει ιδιαίτερη έμφαση στην εξάλειψη της υπερπροσαρμογής [17]. Το Rasa NLU παρέχει πολλά στοιχεία αναγνώρισης οντοτήτων, τα οποία μπορούν να στοχεύσουν τις προσαρμοσμένες απαιτήσεις του κάθε chatbot.



Εικόνα 2.6 Rasa και εξαγωγή οντοτήτων  
Πηγή: rasa.com

Η βιβλιοθήκη SpaCy προσφέρει προεκπαιδευμένους εξαγωγείς οντοτήτων. Όπως και με τις ενσωματώσεις λέξεων, έτσι και με το Spacy, υποστηρίζονται μόνο ορισμένες γλώσσες.

### 2.2.2.3 Training Data

Τα δεδομένα εκπαίδευσης NLU αποτελούνται από πιθανά παραδείγματα μηνυμάτων του χρήστη που κατηγοριοποιούνται με βάση την πρόθεση [24]. Για την διευκόλυνση της χρήσης των προθέσεων του χρήστη, μπορούν να δοθούν παραδείγματα που σχετίζονται με αυτά που θέλει να ταιριάξει ο χρήστης, αποφεύγοντας κενά και ειδικούς χαρακτήρες.

### 2.2.2.4 Entities

Οι οντότητες είναι δομημένα κομμάτια πληροφοριών μέσα σε ένα μήνυμα χρήστη [24]. Για να λειτουργήσει η εξαγωγή οντοτήτων, πρέπει, είτε να καθοριστούν δεδομένα κατάρτισης για να εκπαιδευτεί ένα μοντέλο ML, είτε να οριστούν κανονικές εκφράσεις για να επιτευχθεί η εξαγωγή

οντοτήτων, χρησιμοποιώντας το `RegexEntityExtractor` μοτίβο που βασίζεται σε ένα μοτίβο χαρακτήρων.

```
nlu:
- intent: check_balance
  examples: |
    - how much do I have on my [savings](account) account
    - how much money is in my [checking>{"entity": "account"} account
    - What's the balance on my [credit card account>{"entity": "account", "value": "credit"}
```

Εικόνα 2.7 Οντότητες στο Rasa Framework  
Πηγή: rasa.com

Όταν αποφασιστεί ποιες οντότητες πρέπει να εξαχθούν, ο χρήστης πρέπει να σκεφτεί ποιες πληροφορίες χρειάζεται το chatbot με σκοπό την επίτευξη των στόχων του χρήστη. Στα δεδομένα εκπαίδευσης ο χρήστης έχει τη δυνατότητα να δώσει πληροφορία σχετικά με τον τρόπο εύρεσης οντοτήτων στα δεδομένα εκπαίδευσης.

### 2.2.2.5 Regular Expressions

Υπάρχει η δυνατότητα χρήσης κανονικών εκφράσεων με σκοπό τη βελτίωση της ταξινόμησης πρόθεσης και την εξαγωγή οντοτήτων σε συνδυασμό με τα στοιχεία `RegexFeaturizer` και `RegexEntityExtractor` στη διοχέτευση [24].

```
nlu:
- regex: account_number
  examples: |
    - \d{10,12}
```

Εικόνα 2.8 Rasa και REGular Expressions  
Πηγή: rasa.com

### 2.2.2.6 Synonyms

Τα συνώνυμα αντιστοιχίζουν τις εξαγόμενες οντότητες σε μια τιμή διαφορετική από το κυριολεκτικό κείμενο που εξάγεται [24]. Ο χρήστης μπορεί να χρησιμοποιήσει συνώνυμα όταν υπάρχουν πολλοί τρόποι με τους οποίους γίνεται αναφορά στο ίδιο πράγμα. Παρατηρώντας τον τελικό στόχο της εξαγωγής μιας οντότητας δίνεται η δυνατότητα υπολογισμού των τιμών που θεωρούνται ισοδύναμες.

```
nlu:
- synonym: credit
  examples: |
    - credit card account
    - credit account
```

Εικόνα 2.9 Τα συνώνυμα στο Rasa  
Πηγή: rasa.com

Για παράδειγμα, αν μια οντότητα `account` που χρησιμοποιείται για την αναζήτηση του υπολοίπου του χρήστη, ένας από τους πιθανούς τύπους λογαριασμού που μπορεί να εμφανιστεί είναι ο «πιστωτικός».

Οι χρήστες αναφέρονται επίσης στον "πιστωτικό" λογαριασμό τους ως "λογαριασμός πίστωσης" και "λογαριασμός πιστωτικής κάρτας".

Σε αυτήν την περίπτωση, θα μπορούσε να οριστεί ως "λογαριασμό πιστωτικής κάρτας" και τον "λογαριασμό πίστωσης" ως συνώνυμα του "πίστωση":

### 2.2.2.7 Lookup Tables

Οι πίνακες αναζήτησης είναι λίστες λέξεων που χρησιμοποιούνται για τη δημιουργία μοτίβων τυπικών εκφράσεων χωρίς διάκριση πεζών-κεφαλαίων [24] [18]. Μπορούν να χρησιμοποιηθούν με τους ίδιους τρόπους που χρησιμοποιούνται οι κανονικές εκφράσεις, σε συνδυασμό με τα στοιχεία `RegexFeaturizer` και `RegexEntityExtractor` στο pipeline.

Εντοπίζεται η δυνατότητα πινάκων αναζήτησης για να γίνει ευκολότερη η εξαγωγή οντοτήτων που έχουν ένα γνωστό σύνολο πιθανών τιμών. Οι πίνακες αναζήτησης πρέπει να διατηρηθούν όσο το δυνατόν πιο συγκεκριμένοι. Για παράδειγμα, για εξαγωγή ονομάτων χωρών, θα μπορούσε να προστεθεί ένας πίνακας αναζήτησης όλων των χωρών στον κόσμο:

```
nlu:
- lookup: country
  examples: |
    - Afghanistan
    - Albania
    - ...
    - Zambia
    - Zimbabwe
```

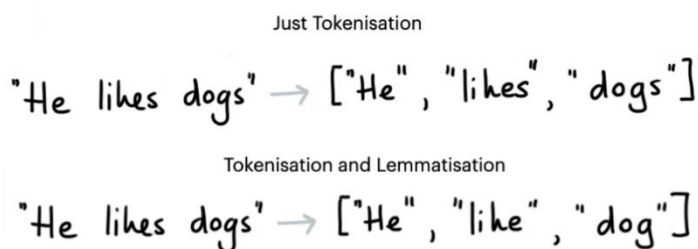
Εικόνα 2.10 Πίνακες Lookup στο Rasa  
Πηγή: rasa.com

### 2.2.2.8 Pipelines

Στο Rasa Open Source, τα εισερχόμενα μηνύματα υποβάλλονται σε επεξεργασία από μια ακολουθία στοιχείων [24]. Τα στοιχεία αυτά εκτελούνται το ένα μετά το άλλο σε μια λεγόμενη διοχέτευση επεξεργασίας, η οποία ορίζεται στο `config.yml` του χρήστη. Η προσαρμογή του μοντέλου και η ακριβής ρύθμιση στο σύνολο των δεδομένων επιτρέπεται μέσω της επιλογής μιας διοχέτευσης NLU.

#### 2.2.2.8.1 Tokenizers

Οι Tokenizers χωρίζουν το κείμενο σε διακριτικά στοιχεία [24]. Στο VagBot χρησιμοποιήθηκε το `WhitespaceTokenizer`, το οποίο δημιουργεί ένα διακριτικό (token) για κάθε ακολουθία χαρακτήρων που χωρίζεται με κενά.



Εικόνα 2.11 Tokenizers-Lemmatizers  
Πηγή: rasa.com

Οποιοσδήποτε χαρακτήρας που δεν εμπεριέχεται στα αλφαριθμητικά και `"_#@&"` θα αντικατασταθεί με κενό διάστημα πριν διαχωριστεί σε κενά, εάν ο χαρακτήρας πληροί οποιαδήποτε από τις ακόλουθες προϋποθέσεις, όπως φαίνονται στην παρακάτω εικόνα.

- the character follows a whitespace: `" !word" → "word"`
- the character precedes a whitespace: `"word! " → "word"`
- the character is at the beginning of the string: `"!word" → "word"`
- the character is at the end of the string: `"word!" → "word"`

Note that:

- `"wo!rd" → "wo!rd"`

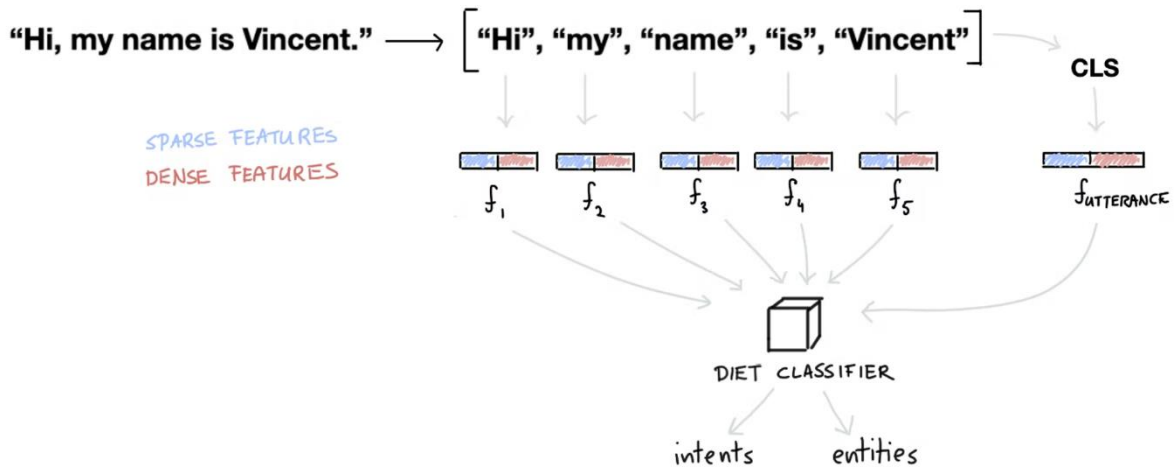
Εικόνα 2.12 Ο Whitespace Tokenizer

Lemmatization είναι η αντιστοίχιση των βασικών μορφών των λέξεων. Εξαλείφει τις καταλήξεις πτώσεων και πληθυντικού αριθμού και επιστρέφει την κάθε λέξη στην βασική της μορφή. Να σημειωθεί πως οι lemmatizers δεν μπορούν να δράσουν σε κάθε προσαρμοσμένο λεξιλόγιο ενός chatbot, όπως ονόματα κ.α. .

#### 2.2.2.8.2 Featurizers

Οι εξαγωγείς χαρακτηριστικών κειμένου χωρίζονται σε δύο διαφορετικές κατηγορίες: στους αραιούς και στους πυκνούς (sparse featurizers and dense featurizers) [24]. Οι αραιοί επιστρέφουν διανύσματα χαρακτηριστικών με πολλές τιμές που λείπουν, π.χ. μηδενικά. Καθώς αυτά τα διανύσματα χαρακτηριστικών θα καταλάμβαναν κανονικά πολλή μνήμη, επιλέγεται η αποθήκευση ως αραιά χαρακτηριστικά. Τα αραιά χαρακτηριστικά αποθηκεύουν μόνο τις τιμές που δεν είναι μηδενικές και τις

θέσεις τους στο διάνυσμα. Έτσι, εξοικονομείται πολλή μνήμη και δίνεται η δυνατότητα εκπαίδευσης σε μεγαλύτερα σύνολα δεδομένων.



Εικόνα 2.13 Οι Featurizers στο Rasa  
Πηγή: rasa.com

Όλοι οι featurizers μπορούν να επιστρέψουν δύο διαφορετικά είδη χαρακτηριστικών: χαρακτηριστικά ακολουθίας και χαρακτηριστικά πρότασης (sequence features and sentence features). Τα χαρακτηριστικά της ακολουθίας είναι ένας πίνακας μεγέθους (αριθμός διακριτικών επι της διάστασης χαρακτηριστικών). Ο πίνακας περιέχει ένα διάνυσμα χαρακτηριστικών για κάθε διακριτικό της ακολουθίας, το οποίο επιτρέπει να εκπαιδευτούν τα μοντέλα ακολουθίας. Τα χαρακτηριστικά πρότασης αντιπροσωπεύονται από έναν πίνακα μεγέθους (1 επι τη διάσταση του χαρακτηριστικού). Περιέχει το διάνυσμα χαρακτηριστικών για την πλήρη έκφραση. Τα χαρακτηριστικά της πρότασης μπορούν να χρησιμοποιηθούν σε οποιοδήποτε μοντέλο «τσάντας λέξεων» (bag-of-words). Ο κάθε ταξινομητής μπορεί επομένως να παραμετροποιηθεί προκειμένου να αποφασισθεί τι είδους χαρακτηριστικά θα χρησιμοποιήσει.

### 2.2.2.8.3 Classifiers

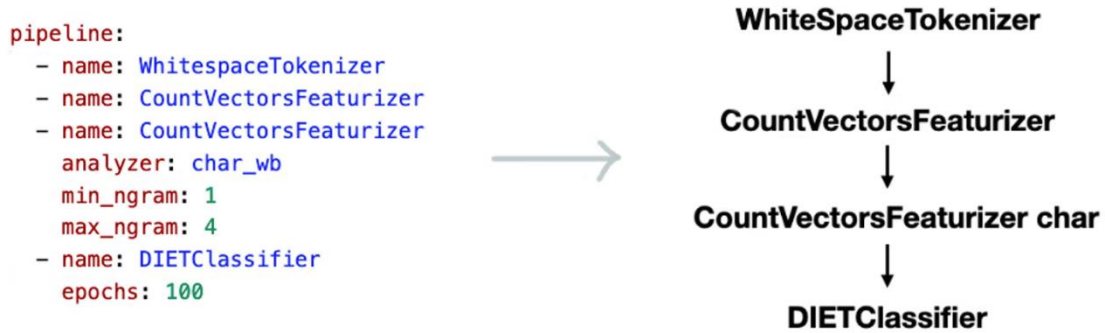
Ο ταξινομητής DIET (Dual Intent and Entity Transformer) είναι μια αρχιτεκτονική για ταξινόμηση πρόθεσης και αναγνώριση οντοτήτων [19]. Η αρχιτεκτονική βασίζεται σε έναν μετασχηματιστή που είναι κοινόχρηστος και για τις δύο εργασίες. Μια ακολουθία ετικετών οντοτήτων προβλέπεται μέσω ενός επιπέδου επισήμανσης τυχαίου πεδίου υπό όρους (Conditional Random Field) πάνω από την ακολουθία εξόδου του μετασχηματιστή που αντιστοιχεί στην ακολουθία εισόδου των διακριτικών. Για τις ετικέτες πρόθεσης, η έξοδος του μετασχηματιστή για την πλήρη έκφραση και οι ετικέτες πρόθεσης, είναι ενσωματωμένες σε έναν ενιαίο σημασιολογικό διανυσματικό χώρο.

Πριν από τον ταξινομητή DIET, το NLU της Rasa χρησιμοποιούσε ένα μοντέλο bag-of-words όπου υπήρχε ένα διάνυσμα χαρακτηριστικών ανά μήνυμα χρήστη[15]. Αν και αυτή είναι μια γρήγορη και πολύ αποδοτική προσέγγιση, το DIET έχει καλύτερη απόδοση.

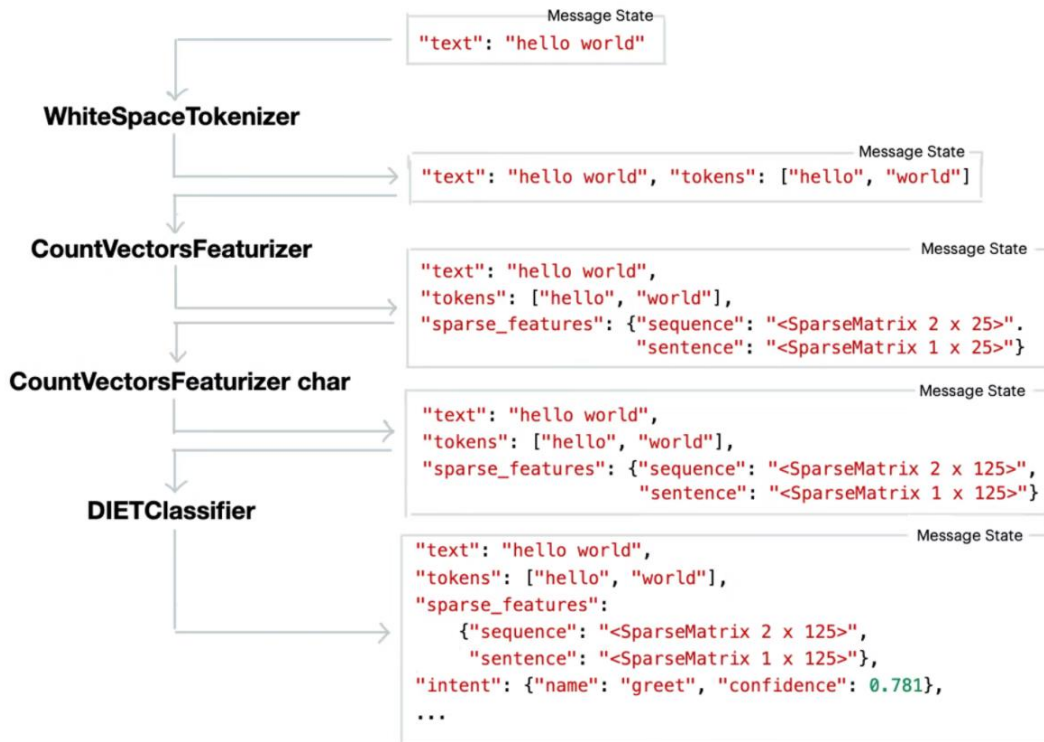
Ο DIET χρησιμοποιεί ένα μοντέλο ακολουθίας που λαμβάνει υπόψη τη σειρά λέξεων, προσφέροντας έτσι καλύτερη απόδοση. Είναι επίσης ένα πιο συμπαγές μοντέλο με plug-and-play αρχιτεκτονική. Για παράδειγμα, μπορεί να χρησιμοποιηθεί ο ταξινομητής DIET για να επιτευχθεί τόσο ταξινόμηση πρόθεσης όσο και εξαγωγή οντοτήτων. Επίσης, με την κατάλληλη παραμετροποίηση γίνεται να

εκτελεστεί μεμονωμένα η κάθε λειτουργία του DIET. Για παράδειγμα, δίνεται η δυνατότητα να απενεργοποιηθεί η ταξινόμηση πρόθεσης και να χρησιμοποιηθεί μόνο για την εξαγωγή οντοτήτων.

Επιπλέον, ο DIET όχι μόνο είναι πολύ πιο γρήγορος στην εκπαίδευση, αλλά και ανταγωνίζεται σε απόδοση μεγάλης κλίμακας προεκπαιδευμένα μοντέλα γλωσσών. Ξεπερνά σε απόδοση μέχρι και ένα Fine-tuned ταξινομητή BERT και βελτιώνει την τρέχουσα κατάσταση σε ένα σύνθετο σύνολο δεδομένων NLU. Γίνεται ευκολά αντιληπτό πως πλέον, η επίτευξη ακρίβειας δεν λειτουργεί εις βάρος της απόδοσης.



Εικόνα 2.14 Παράδειγμα Pipeline 1  
Πηγή: rasa.com

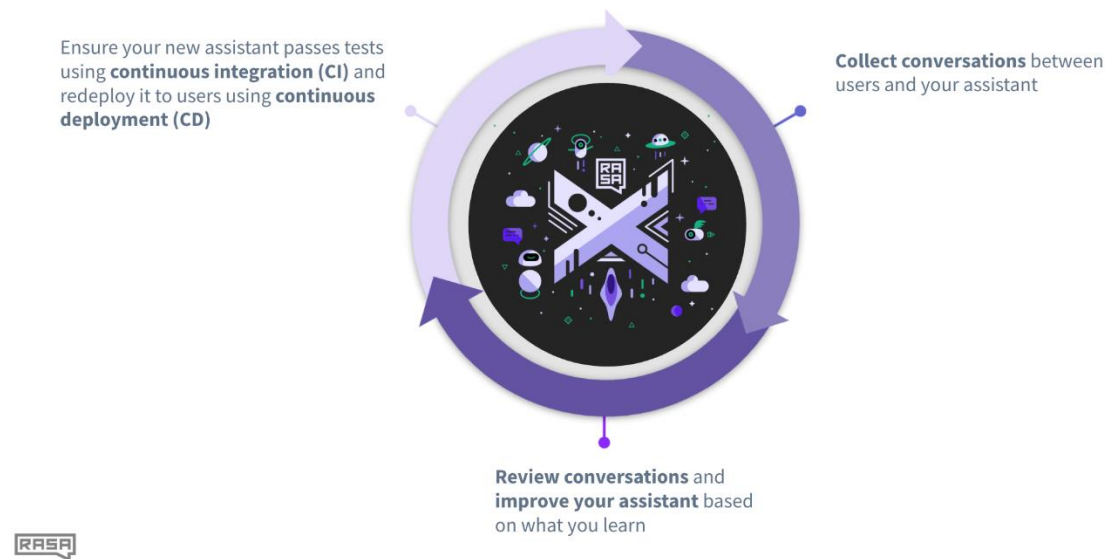


Εικόνα 2.15 Παράδειγμα Pipeline 2  
Πηγή: rasa.com

### 2.3 Rasa X

Το Rasa X είναι ένα εργαλείο για την ανάπτυξη με γνώμονα τη συνομιλία (Conversation-Driven Development CDD), τη διαδικασία ακρόασης των χρηστών και χρήσης αυτών των πληροφοριών για τη βελτίωση του βοηθού AI [20]. Η πρόκληση κατά τη δημιουργία μεγάλων βοηθών τεχνητής νοημοσύνης είναι ότι δεν είναι εφικτό να γίνει πρόβλεψη όλων όσων θα μπορούσαν να ρωτήσουν οι χρήστες. Το πιο πιθανό σε κάθε συνομιλία είναι οι χρήστες να μην στέλνουν μηνύματα ακριβώς όπως έχουν περιγραφεί αυτά στο μοντέλο.

#### Continually improve your assistant using Rasa X

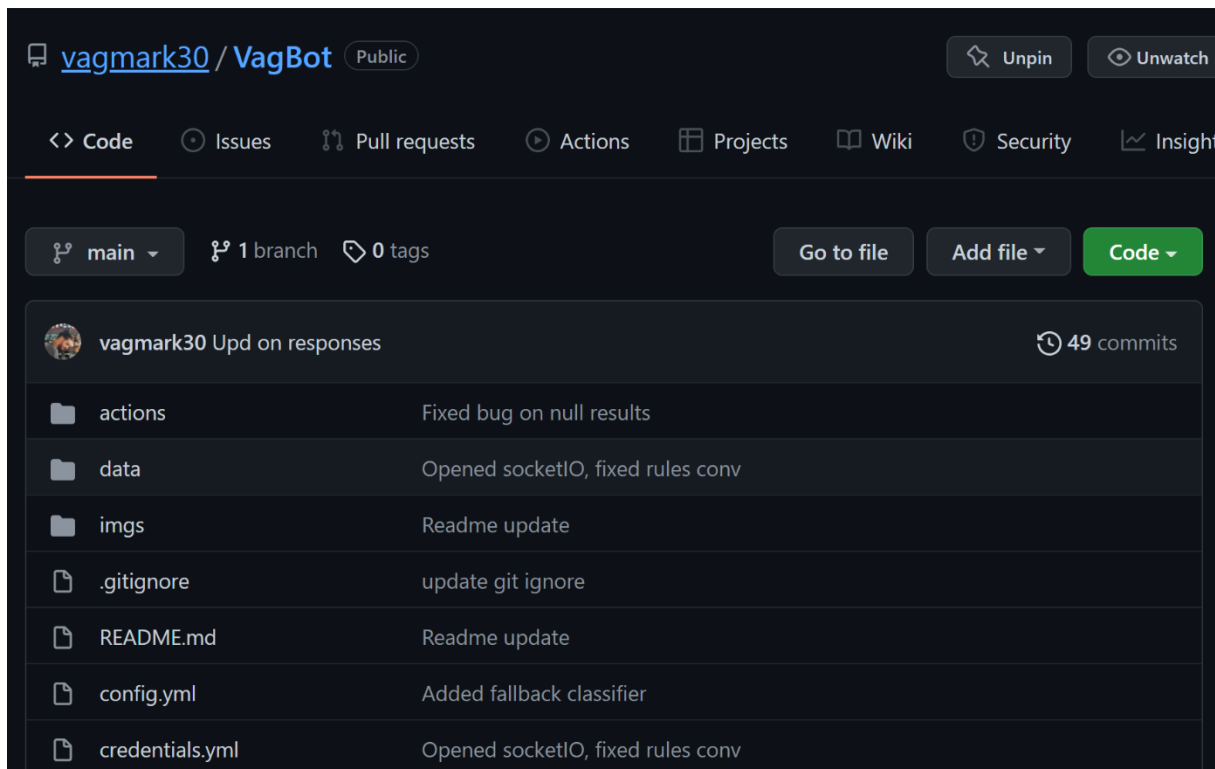


Εικόνα 2.16 Το Rasa x  
Πηγή: rasa.com

Με τη δημιουργία ενός βοηθού που μπορεί να χειριστεί τις πιο σημαντικές ιστορίες, μπορεί να χρησιμοποιηθεί το Rasa X, με σκοπό τη βελτίωση του βοηθού. Είναι ένα εργαλείο που βοηθάει στην καλύτερη οπτικοποίησης ενός chatbot κατά την ανάπτυξη του. Δίνει την δυνατότητα στον δημιουργό να συλλεγεί συνομιλίες που έχει κάνει κάποιο chatbot και ύστερα να τις αναλύει και να τις αξιολογεί.

## Κεφάλαιο 3ο: Ανάπτυξη VagBot

Σε αυτή την ενότητα θα γίνει παρουσίαση των βημάτων που ακολουθήθηκαν προκειμένου να εγκατασταθεί αρχικά το κατάλληλο υπόβαθρο, όπου θα αναπτυχθεί το chatbot. Στη συνέχεια, θα γίνει ανάλυση της σταδιακής ανάπτυξής του.



Εικόνα 3.1 Το GitHub Repository του VagBot  
Πηγή: [github.com/vagmark30](https://github.com/vagmark30)

### 3.1 Εγκατάσταση εργαλείων

#### 3.1.1 Python

Αρχικά, είναι αναγκαίο να εγκατασταθεί η Python και ταυτόχρονα ο package manager pip. Αυτό γίνεται από την κεντρική της ιστοσελίδα και ύστερα επιλέγοντας την τελευταία έκδοση για το λειτουργικό σύστημα του υπολογιστή που πρόκειται να εγκατασταθεί.

Το pip είναι ένας διαχειριστής πακέτων για την Python, που σημαίνει πως είναι ένα εργαλείο που επιτρέπει την εγκατάσταση και τη διαχείριση βιβλιοθηκών, οι οποίες δεν είναι προεγκατεστημένες στη βασική έκδοση της Python.

#### 3.1.2 Anaconda

Το Anaconda είναι η πιο δημοφιλής πλατφόρμα διανομής (distribution platform) Python στον κόσμο με περισσότερους από 25 εκατομμύρια χρήστες. Είναι ένα open-source οικοσύστημα για επιστημονικούς κυρίως σκοπούς, όπως η επιστήμη δεδομένων (data science), εφαρμογές μηχανικής μάθησης (machine learning), κ.λπ. που στοχεύει στην απλοποίηση της διαχείρισης πακέτων κατά την ανάπτυξη

εφαρμογών. Μερικά από τα πιο διαδεδομένα εργαλεία που προσφέρονται είναι τα εξής, Jupiter, Matplotlib, SciPi, Spyder, ScikitLearn κ.α.



Εικόνα 3.2 Το εργαλείο Anaconda  
Πηγή: anaconda.com

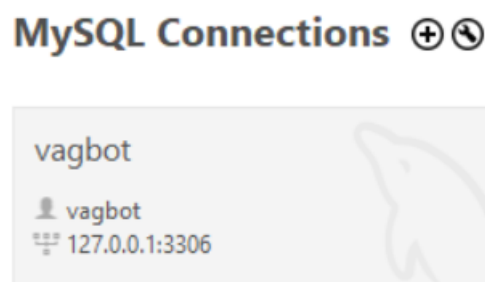
Το Anaconda χρησιμοποιήθηκε για τη δημιουργία περιβάλλοντος, μέσα στο οποίο θα εγκατασταθούν τα επόμενα εργαλεία για την ανάπτυξη του chatbot. Η εξής εντολή `conda create --name rasa` δημιουργεί ένα περιβάλλον ονόματος `rasa`.

### 3.1.3 Rasa Open Source και Rasa X

Στη συνέχεια, θα γίνει εγκατάσταση του Rasa Open Source μέσα σε ένα περιβάλλον (environment) του Anaconda. Χρησιμοποιώντας τη γραμμή εντολών (terminal), πρέπει πρώτα να ενεργοποιηθεί το περιβάλλον που δημιουργήθηκε στην προηγούμενη ενότητα, εκτελώντας την παρακάτω εντολή: `conda activate rasa`. Έπειτα, εκτελώντας τις εντολές `conda install rasa` και `conda install rasa-x`, θα γίνει εγκατάσταση του Rasa Open Source και Rasa X.

### 3.1.4 MySQL

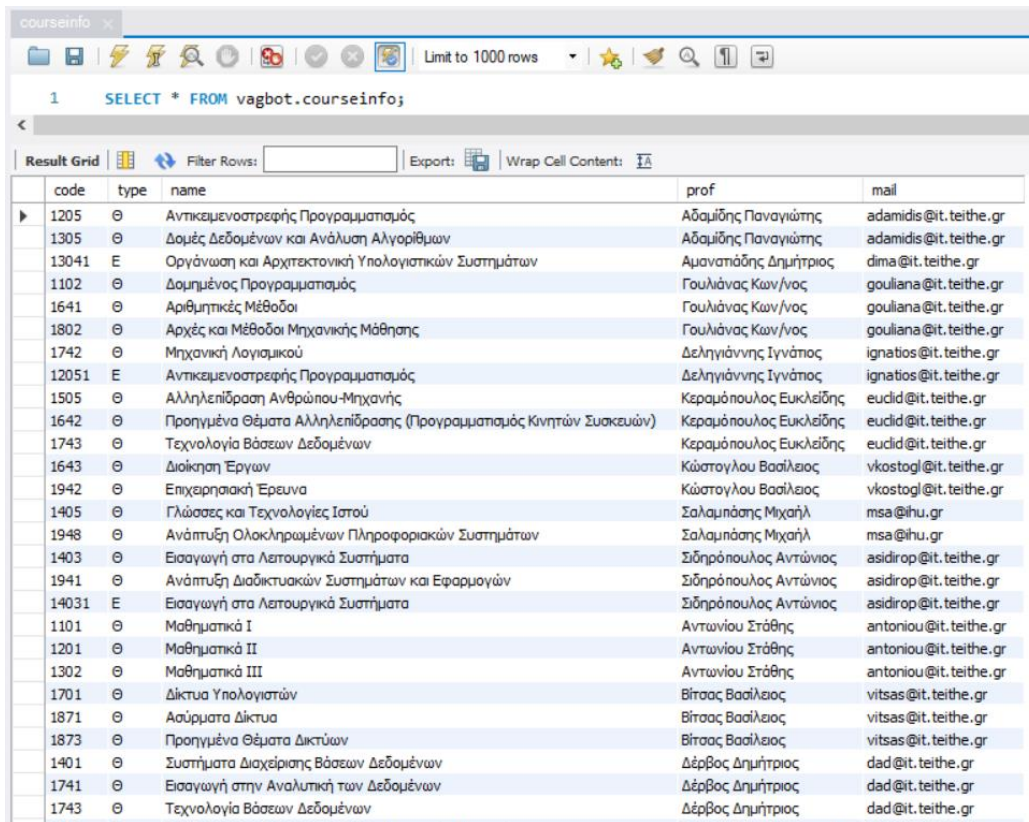
Επιλέχθηκε η MySQL ως κύριο μέσο αποθήκευσης των δεδομένων, τα οποία ήταν απαραίτητα για τις συνομιλίες του VagBot. Η λήψη έγινε από την κεντρική ιστοσελίδα της MySQL μαζί με το εργαλείο MySQL Workbench, για την καλύτερη οπτικοποίηση των δεδομένων. Δημιουργήθηκε η βάση που φαίνεται παρακάτω στον τοπικά εγκαταστημένο εξυπηρετητή της MySQL στον Η/Υ του VagBot, ώστε να δέχεται συνδέσεις στην θύρα 3306 σύμφωνα με την [21].



Εικόνα 3.3 Η βάση δεδομένων του VagBot

Η βάση δεδομένων χρησιμοποιήθηκε για την αποθήκευση των οντοτήτων που φαίνονται στην παρακάτω εικόνα. Στον πίνακα `courseinfo` συνδέεται κάθε μάθημα που προσφέρεται από τη σχολή με τον καθηγητή ο οποίος το διδάσκει, συμπεριλαμβανομένου και το πανεπιστημιακό/προσωπικό του

email. Στη συνέχεια, καθώς παρουσιάζεται το VagBot, θα γίνει κατανοητός ο λόγος ο οποίος ήταν αναγκαία η σύνδεση μεταξύ των οντοτήτων.



code	type	name	prof	mail
1205	Θ	Αντικειμενοστρεφής Προγραμματισμός	Αδαμίδης Παναγιώτης	adamidis@it.teithe.gr
1305	Θ	Δομές Δεδομένων και Ανάλυση Αλγορίθμων	Αδαμίδης Παναγιώτης	adamidis@it.teithe.gr
13041	E	Οργάνωση και Αρχιτεκτονική Υπολογιστικών Συστημάτων	Αμανασίδης Δημήτριος	dima@it.teithe.gr
1102	Θ	Δομημένος Προγραμματισμός	Γουλιάνας Κων/νος	gouliana@it.teithe.gr
1641	Θ	Αριθμητικές Μέθοδοι	Γουλιάνας Κων/νος	gouliana@it.teithe.gr
1802	Θ	Αρχές και Μέθοδοι Μηχανικής Μάθησης	Γουλιάνας Κων/νος	gouliana@it.teithe.gr
1742	Θ	Μηχανική Λογισμικού	Δεληγιάννης Ιγνάπος	ignatios@it.teithe.gr
12051	E	Αντικειμενοστρεφής Προγραμματισμός	Δεληγιάννης Ιγνάπος	ignatios@it.teithe.gr
1505	Θ	Αλληλεπίδραση Ανθρώπου-Μηχανής	Κεραμόπουλος Ευκλείδης	euclid@it.teithe.gr
1642	Θ	Προηγμένα Θέματα Αλληλεπίδρασης (Προγραμματισμός Κινητών Συσκευών)	Κεραμόπουλος Ευκλείδης	euclid@it.teithe.gr
1743	Θ	Τεχνολογία Βάσεων Δεδομένων	Κεραμόπουλος Ευκλείδης	euclid@it.teithe.gr
1643	Θ	Διοίκηση Έργων	Κώστογλου Βασίλειος	vkostogl@it.teithe.gr
1942	Θ	Επιχειρησιακή Έρευνα	Κώστογλου Βασίλειος	vkostogl@it.teithe.gr
1405	Θ	Γλώσσες και Τεχνολογίες Ιστού	Σαλαμπόσης Μιχαήλ	msa@ihu.gr
1948	Θ	Ανάπτυξη Ολοκληρωμένων Πληροφοριακών Συστημάτων	Σαλαμπόσης Μιχαήλ	msa@ihu.gr
1403	Θ	Εισαγωγή στα Λειτουργικά Συστήματα	Σιδιρόπουλος Αντώνιος	asidirop@it.teithe.gr
1941	Θ	Ανάπτυξη Διοδικτυακών Συστημάτων και Εφαρμογών	Σιδιρόπουλος Αντώνιος	asidirop@it.teithe.gr
14031	E	Εισαγωγή στα Λειτουργικά Συστήματα	Σιδιρόπουλος Αντώνιος	asidirop@it.teithe.gr
1101	Θ	Μαθηματικά I	Αντωνίου Στάθης	antoniou@it.teithe.gr
1201	Θ	Μαθηματικά II	Αντωνίου Στάθης	antoniou@it.teithe.gr
1302	Θ	Μαθηματικά III	Αντωνίου Στάθης	antoniou@it.teithe.gr
1701	Θ	Δίκτυα Υπολογιστών	Βίτσας Βασίλειος	vitsas@it.teithe.gr
1871	Θ	Ασύρματα Δίκτυα	Βίτσας Βασίλειος	vitsas@it.teithe.gr
1873	Θ	Προηγμένα Θέματα Δικτύων	Βίτσας Βασίλειος	vitsas@it.teithe.gr
1401	Θ	Συστήματα Διαχείρισης Βάσεων Δεδομένων	Δέρβος Δημήτριος	dad@it.teithe.gr
1741	Θ	Εισαγωγή στην Αναλυτική των Δεδομένων	Δέρβος Δημήτριος	dad@it.teithe.gr
1743	Θ	Τεχνολογία Βάσεων Δεδομένων	Δέρβος Δημήτριος	dad@it.teithe.gr

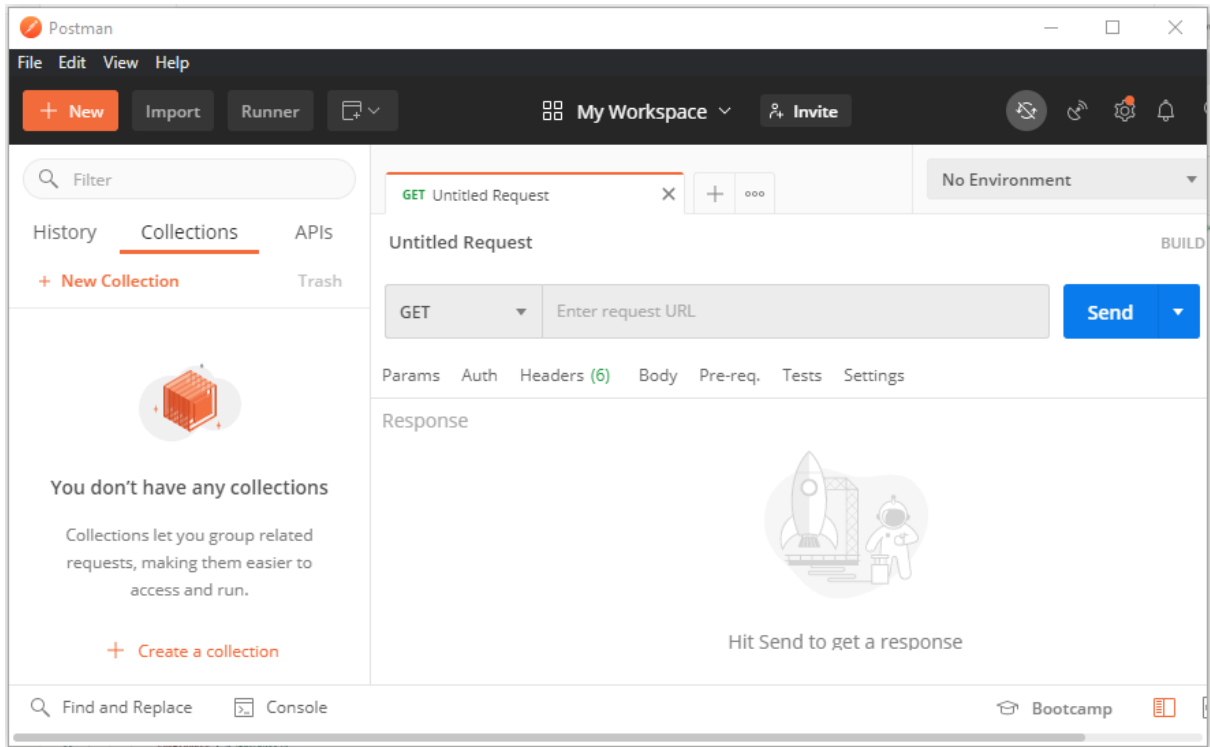
Εικόνα 3.4 Ο πίνακας CourseInfo

### 3.1.5 Ngrok

Το ngrok είναι ένα εργαλείο, το οποίο δίνει τη δυνατότητα να εκθέτει μια εικονική θύρα δικτύου του υπολογιστή στον παγκόσμιο ιστό. Αυτό πραγματοποιείται συνδέοντας το τοπικό πρόγραμμα ngrok με δημόσιες διευθύνσεις, στις οποίες θα αναμεταδίδεται κάθε κλήση που γίνεται σε αυτή τη διεύθυνση, πρώτα στο τοπικό ngrok και στη συνέχεια στην τοπική διεύθυνση και θύρα που ορίστηκε. Η λήψη πραγματοποιήθηκε από την κεντρική ιστοσελίδα <https://ngrok.com/>.

### 3.1.6 Postman

Η εφαρμογή Postman, είναι ένα εργαλείο το οποίο χρησιμοποιήθηκε προκειμένου να δοκιμαστεί η επικοινωνία μεταξύ της πλατφόρμας που θα στέλνει τα μηνύματα ο χρήστης και του chatbot. Δίνει τη δυνατότητα να εκτελεστούν αιτήματα διαφόρων πρωτοκόλλων όπως REST, SOAP κ.λπ. σε κάποιο API [22]. Έπειτα, εντοπίζει αυτόματα τη γλώσσα της απάντησης και τη μορφή του κειμένου, ώστε να διευκολύνει την επιθεώρηση των αποτελεσμάτων. Σε επόμενη ενότητα θα διαπιστωθεί η χρησιμότητα αυτού του εργαλείου, δίνοντας παραδείγματα χρήσης του.



Εικόνα 3.5 Η εφαρμογή Postman

### 3.1.7 Git και GitHub

Τόσο το VagBot όσο και το VagFront project που υλοποιήθηκαν, χρησιμοποίησαν το εργαλείο git (version control system) για την καλύτερη και αποδοτικότερη ανάπτυξη των εφαρμογών και την πλατφόρμα GitHub, η οποία φιλοξενεί διαδικτυακά τα repository.

## 3.2 Δημιουργία project

Εκτελώντας την εντολή `rasa init`, δημιουργεί ένα νέο ολοκληρωμένο project με παραδείγματα εκπαιδευτικών δεδομένων (training data), ενεργειών (actions) και αρχείων παραμετροποίησης (configuration files).

```
(rasa) C:\Users\vagmark\diplomatest>rasa init
Welcome to Rasa! 🤖

To get started quickly, an initial project will be created.
If you need some help, check out the documentation at https://rasa.com/docs/rasa.
Now let's start! 🤖📁

? Please enter a path where the project will be created [default: current directory]
Created project directory at 'C:\Users\vagmark\diplomatest'.
Finished creating project structure.
? Do you want to train an initial model? 🤖📁 Yes
Training an initial model...
Core model training completed.
Your Rasa model is trained and saved at 'C:\Users\vagmark\diplomatest\models\20220121-183807.tar.gz'.
? Do you want to speak to the trained assistant on the command line? 🤖📁 Yes
```

Εικόνα 3.6 Η εντολή `rasa init`

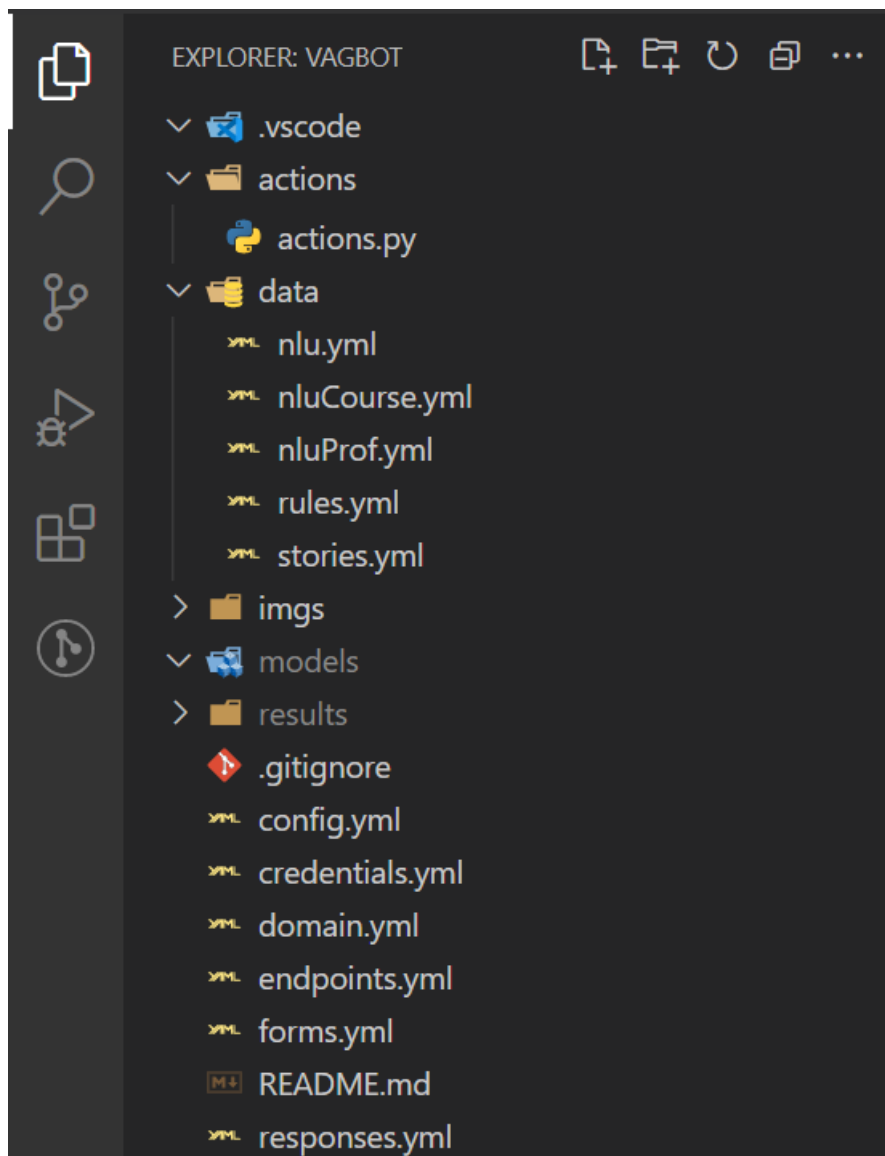
```

Bot loaded. Type a message and press enter (use '/stop' to exit):
Your input -> hey
Hey! How are you?
Your input -> great
Great, carry on!
Your input -> thanks
Bye
Your input -> 

```

Εικόνα 3.7 Η πρώτη συνομιλία με το VagBot

Μετά την αρχικοποίηση του VagBot το εργαλείο επιλογής ανάπτυξης του project ήταν το Visual Studio Code.



Εικόνα 3.8 Η Δομή ενός Rasa Project

### 3.3 Θεματολογία

Σε αυτήν την ενότητα θα γίνει αναφορά της θεματολογίας, η οποία επιλέχθηκε με σκοπό την επιλογή του εύρους των ζητημάτων που καλύπτεται από το VagBot. Ο στόχος της επιλογής των θεμάτων [14] έγινε με γνώμονα των παρακάτω χαρακτηριστικών:

- να καθοδηγήσει και να προσελκύσει τους χρήστες
- για τη βελτίωση της παραγωγικότητας της συνομιλίας
- να διατηρήσει τη συνομιλία ζωντανή
- για την παροχή πρόσθετων, χρήσιμων πληροφοριών.

Στο σημείο αυτό, πρέπει να αναφερθεί πως υπήρξε επικοινωνία με καθηγητές του τμήματος, οι οποίοι εξέφρασαν την γνώμη τους, προκειμένου να υπάρχει μια σφαιρική κάλυψη των ζητημάτων που καλύπτονται. Συγκεκριμένα τα θέματα είναι:

- Στοιχεία επικοινωνίας με τους καθηγητές του τμήματος.
- Τα μαθήματα που διδάσκει κάθε καθηγητής.
- Προβλήματα σύνδεσης στις υπηρεσίες του τμήματος όπως Apps, Moodle, Thesis, Webmail, Pithia, Eudoxos, AcademicID.
- Προϋποθέσεις ανάληψης πτυχιακής/διπλωματικής εργασίας.
- Πού πρέπει να απευθυνθεί κανείς για απορίες ή προβλήματα.
- Προϋποθέσεις συμμετοχής στην εμβόλιμη εξεταστική και ποια μαθήματα προσφέρονται.
- Πρόγραμμα εξεταστικής.

Έχοντας την βασική δομή ενός rasa project έτοιμη, πρέπει να αναπτυχθεί το υλικό, με βάση το οποίο θα εκπαιδευτεί το VagBot.

### 3.4 Υλοποίηση

Μετά την επιλογή της θεματολογίας, το επόμενο βήμα είναι η δημιουργία των stories, τα οποία βασίζονται στα παραπάνω θέματα. Ενδεικτικά τα story που δημιουργήθηκαν είναι τα εξής:

```
version: "2.0"
stories:
- story: greet and ask
- story: are you a bot
- story: covered topics
- story: ptyxiakh able
- story: ptyxiakh unable
- story: diplwmatikh able
- story: diplwmatikh unable
- story: how many dm have i
- story: emvolimh able
- story: emvolimh courses
- story: problem with service
- story: who is in charge of a service
- story: password recovery
- story: problem with apps as registred
- story: problem with apps as non registred
- story: professor info
- story: professor courses
- story: exams schedule
```

Εικόνα 3.9 Τα stories του VagBot

Στη συνέχεια, είναι αναγκαία η δημιουργία των δεδομένων εκπαίδευσης (training data), τα οποία οφείλουν να ανταπεξέλθουν σε κάθε πιθανή ερώτηση που μπορεί να θέσει ο χρήστης. Τα training data ομαδοποιούνται σε κατηγορίες από προθέσεις (intents) και κάθε πρόθεση μπορεί να έχει όσα παραδείγματα από training data κι αν χρειαστούν. Τα intents που δημιουργήθηκαν για το VagBot παραθέτονται στην πρώτη στήλη του παρακάτω πίνακα. Στη δεύτερη στήλη δίνεται ένα πιθανό μήνυμα χρήστη το οποίο θα ταξινομηθεί αντίστοιχη πρόθεση.

Πίνακας 3.1 Προθέσεις

Πρόθεση (Intent)	Παράδειγμα
greet	Γεια
Bot_challenge	Μιλάω με άνθρωπο;
which_topics_are_covered	Σε τι μπορείς να με βοηθήσεις?
affirm	Ναι
deny	Όχι
am_i_able_ptyxiakh	Μπορώ να πάρω Πτυχιακή?
how_many_dm_have_i	Πόσες διδακτικές μονάδες έχω;
am_i_able_diplwmatikh	Μπορώ να πάρω Διπλωματική?
am_i_able_emvolimh	Ποια είναι τα κριτήρια για να συμμετάσχω στην εμβόλιμη εξεταστική;
which_courses_are_offered_in_emvolimh	Που μπορώ να δω ποια μαθήματα είναι στην εμβόλιμη;
having_issue_with_a_service	Έχω πρόβλημα με το πυθία/thesis/webmail/eudoxos/moodle
in_charge_of_service	Από ποιον διαχειρίζεται το apps/pythia/mail/...
password_recovery	Έχω ξεχάσει τον κωδικό μου στο apps/moodle/pythia/...
problem_with_apps	Δεν μπορώ να συνδεθώ στο apps
exams_schedule	Ποιο είναι το πρόγραμμα εξεταστικής;
which_courses_by_professor	Ποια μαθήματα κάνει ο Διαμαντάρας;
whats_the_professors_info	Ποιο είναι το mail της Ασδρέ

Στο VagBot το intent exams\_schedule έχει 4 παραδείγματα, πράγμα που το κάνει ένα πολύ απλό intent προκειμένου να ταξινομηθεί σωστά από το μοντέλο.

```
- intent: exams_schedule
examples: |
  - Ποιο είναι το πρόγραμμα εξεταστικής;
  - Που μπορώ να βρω το πρόγραμμα εξεταστικής;
  - Που υπάρχει το πρόγραμμα εξεταστικής;
  - Πρόγραμμα εξεταστικής;
```

Εικόνα 3.10 Παράδειγμα πρόθεσης: Πρόγραμμα εξεταστικής

Απεναντίας το intent: whats\_the\_professors\_info έχει 225 παραδείγματα διότι έχει πολλές διαφορετικές οντότητες, στην προκειμένη περίπτωση 53 καθηγητές, από τις οποίες πρέπει να εκπαιδευτεί.

```
- intent: whats_the_professors_info
examples: |
  - ποιο είναι το [webmail](yphresia) του [Αδαμίδης Παναγιώτης](professor)
  - [email](yphresia) [Αμανατιάδης Δημήτριος](professor)
  - ποιο είναι το [email](yphresia) του [Αμανατιάδης Δημήτριος](professor)
  - ποιο είναι το [μειλ](yphresia) [Αδαμίδης Παναγιώτης](professor)
  - [μειλ](yphresia) [Αδαμίδης Παναγιώτης](professor)
  - ποιο είναι το [μειλ](yphresia) [Αδαμίδης Παναγιώτης](professor)
  - [μειλ](yphresia) [Αμανατιάδης Δημήτριος](professor)
  - [εμαιλ](yphresia) [Γουλιάνας Κων/νος](professor)
  - ποιο είναι το [εμαιλ](yphresia) [Αδαμίδης Παναγιώτης](professor)
  - [εμαιλ](yphresia) [Γουλιάνας Κων/νος](professor)
  - ποιο είναι το [εμαιλ](yphresia) [Αδαμίδης Παναγιώτης](professor)
  - [mail](yphresia) [Δεληγιάννης Ιγνάτιος](professor)
  - ποιο είναι το [mail](yphresia) [Αδαμίδης Παναγιώτης](professor)
  - [webmail](yphresia) [Κεραμόπουλος Ευκλείδης](professor)
  - [webmail](yphresia) [Κώστογλου Βασίλειος](professor)
  ...
  ...
  ...
  - [webmail](yphresia) [Τσιρογιαννη Ανδreas](professor)
  - [webmail](yphresia) [Μπαλτατζη Δημητριος](professor)
  - [webmail](yphresia) [Κωτσακη Ρηγας](professor)
```

Εικόνα 3.11 Παράδειγμα πρόθεσης: Ποια είναι τα στοιχεία του καθηγητή

Για να βοηθηθεί ακόμη περισσότερο το μοντέλο στην εξαγωγή οντοτήτων, δόθηκαν συνώνυμα για κάθε καθηγητή, όπως φαίνονται παρακάτω.

```

- synonym: Δέρβος Δημήτριος
examples: |
  - Δέρβος
  - ΔΕΡΒΟΣ
  - δέρβο
- synonym: Διαμαντάρας Κωνσταντίνος
examples: |
  - Διαμαντάρας
  - ΔΙΑΜΑΝΤΑΡΑΣ
  - διαμαντάρα

```

Εικόνα 3.12 Συνώνυμα καθηγητών

Τέλος, μετά τη συμπλήρωση των training data, το VagBot θα είναι σε θέση να καταλάβει τα ζητούμενα που έχουν τεθεί. Εφόσον ολοκληρωθεί αυτή η διαδικασία, το VagBot έχει τη δυνατότητα να απαντήσει στις ερωτήσεις του χρήστη, βάσει των training data. Οι απαντήσεις που πρόκειται να δώσει το chatbot, πρέπει δηλωθούν και αυτές αντίστοιχα.

```

actions:
- '...'
- utter_greet_and_ask
- utter_i_am_bot
- utter_out_of_scope
- utter_please_rephrase
- utter_ask_dm_ptyxiakh
- utter_ask_dm_diplwmatikh
- utter_able_diplwmatikh
- utter_unable_diplwmatikh
- utter_able_ptyxiakh
- utter_unable_ptyxiakh
- utter_how_many_dm_have_i
- utter_ability_for_emvolimh
- utter_courses_offered_in_emvolimh
- utter_ask_if_registred_on_apps
- utter_problem_as_registred_on_apps
- utter_problem_as_non_registred_on_apps
- action_resolve_issue
- action_in_charge_of_service
- action_password_recovery
- action_get_professor_info
- action_get_professor_courses

```

Εικόνα 3.13 Οι απαντήσεις και οι ενέργειες του VagBot

Σε περίπτωση που τα training data είναι ελλιπή, τότε το VagBot θα αντιμετωπίσει δυσκολία με την αναγνώριση του αντίστοιχου story. Αντίθετα, στην περίπτωση που δοθούν πολλά training data, τότε το μοντέλο που θα δημιουργηθεί θα τείνει να κάνει υπερπροσαρμογή (overfitting) πάνω σε αυτά. Ως εκ τούτου, δε θα μπορεί να γενικεύει και θα υπάρχει δυσκολία κατανόησης των ερωτήσεων που θα απουσιάζουν από τα training data. Επομένως, υπάρχει ανάγκη εύρεσης της κατάλληλης ισορροπίας του πλήθους των training data για κάθε story.

### 3.5 Κανόνες

Όπως προαναφέρθηκε, οι κανόνες περιγράφουν σύντομα κομμάτια συνομιλιών που πρέπει πάντα να ακολουθούν την ίδια διαδρομή [24]. Στο VagBot χρειάστηκαν να χρησιμοποιηθούν τρεις κανόνες. Ο πρώτος κανόνας ενεργοποιείται όταν η αναγνώριση της πρόθεσης (intent) του μηνύματος του χρήστη είναι κάτω από ένα νούμερο, το οποίο έχει επιλεγθεί στο αρχείο config. Στο VagBot δηλαδή, όταν ο ταξινομητής δεν έχει κατανοήσει με πιθανότητα 70% και πάνω, την πρόθεση του μηνύματος.

```
pipeline:
- name: FallbackClassifier
  threshold: 0.7
rules:
- rule: Ask the user to rephrase on low NLU confidence
  steps:
  - intent: nlu_fallback
  - action: utter_please_rephrase
- rule: stop-the-conversation
  steps:
  - action: utter_ask_anything_else
  - intent: deny
  - action: utter_goodbye
- rule: continue-the-conversation
  steps:
  - action: utter_ask_anything_else
  - intent: affirm
  - action: utter_ask_else
```

Εικόνα 3.14 Οι κανόνες του VagBot

Ο δεύτερος και ο τρίτος κανόνας χρειάστηκαν για την σωστή ροή της συζήτησης με το VagBot. Δηλαδή, μετά από το πέρας κάθε ερώτησης, το chatbot θα στέλνει ένα μήνυμα το οποίο θα ρωτάει τον χρήστη αν θα ήθελε κάτι άλλο. Στην περίπτωση που ο χρήστης απαντήσει ναι, τότε το chatbot θα συνεχίσει την συζήτηση, ρωτώντας τον τι άλλο θα ήθελε. Στην περίπτωση που ο χρήστης πει όχι, τότε θα απαντάει με μήνυμα αποχαιρετισμού στον χρήστη.

### 3.6 Παραμετροποίηση

Στο αρχείο config.yml συμβαίνει η παραμετροποίηση του μοντέλου που πρόκειται να εκπαιδευτεί [24]. Στο συγκεκριμένο αρχείο προσδιορίζονται τα pipelines, δηλαδή οι μηχανισμοί με τους οποίους το VagBot πρώτα θα επεξεργαστεί και στη συνέχεια θα καταλάβει τα μηνύματα εισόδου του χρήστη. Επιπροσθέτως, επιλέγονται οι πολιτικές του μοντέλου που καθορίζουν την επιλογή των απαντήσεων. Το VagBot αναπτύχθηκε πάνω στο SpaCy pipeline, το οποίο δίνει την δυνατότητα στη χρήση της ελληνικής γλώσσας.

```

1  language: el
2  pipeline:
3    - name: WhitespaceTokenizer
4    - name: RegexFeaturizer
5    - name: LexicalSyntacticFeaturizer
6    - name: CountVectorsFeaturizer
7    - name: CountVectorsFeaturizer
8      analyzer: "char_wb"
9      min_ngram: 1
10     max_ngram: 4
11    - name: DIETClassifier
12      epochs: 100
13    - name: EntitySynonymMapper
14    - name: ResponseSelector
15      epochs: 100
16    - name: FallbackClassifier
17      threshold: 0.7
18
19  policies:
20    - name: MemoizationPolicy
21    - name: RulePolicy
22    - name: TEDPolicy
23      max_history: 10
24      epochs: 100
25

```

Εικόνα 3.15 Αρχείο config.yml

### 3.7 Παραδείγματα

Παρακάτω θα παρουσιαστούν τέσσερις περιπτώσεις ολοκληρωμένων παραδειγμάτων (story, training data, response). Ο λόγος αναφοράς των τεσσάρων παραδειγμάτων, είναι η διαφορά στη σύνθεση του διαλόγου και στη λήψη της απόφασης αναγνώρισης του story.

#### 3.7.1 Απλό παράδειγμα

Αρχικά, ακολουθεί το πρώτο παράδειγμα που αντιστοιχεί σε απλό story. Ως ένα απλό story θεωρείται μια ερώτηση από ένα χρήστη, οπότε το VagBot πρέπει μόνο να καταλάβει το intent της ερώτησης και να απαντήσει πάνω σε αυτό. Η ταξινόμηση της πρόθεσης στο συγκεκριμένο παράδειγμα, θα επιτευχθεί αν ο χρήστης ρωτήσει κάτι παρεμφερές των παραδειγμάτων που δόθηκαν στα δεδομένα εκπαίδευσης. Πιο συγκεκριμένα, θα μπορούσε ο χρήστης να θέσει μια ερώτηση που δεν εντάσσεται μέσα στο intent: exams\_schedule, όπως “Θέλω να μάθω το πρόγραμμα εξεταστικής”, και το DietClassifier χρησιμοποιώντας τις τεχνικές που προαναφέρθηκαν θα μπορέσει να το ταξινομήσει στο ανάλογο intent. Εφόσον τον αναγνωρίσει με 70% και πάνω confidence, τότε θα δώσει την απάντηση που ορίζεται στο story, δηλαδή: utter\_exams\_schedule.

```

- story: exams schedule
  steps:
  - intent: exams_schedule
  - action: utter_exams_schedule

- intent: exams_schedule
  examples: |
  - Ποιο είναι το πρόγραμμα εξεταστικής;
  - Που μπορώ να βρω το πρόγραμμα εξεταστικής;
  - Που υπάρχει το πρόγραμμα εξεταστικής;
  - Πρόγραμμα εξεταστικής;

  utter_exams_schedule:
  - text: |
    Το πρόγραμμα εξεταστικής μπορείτε να το βρείτε σε αυτό το σύνδεσμο:
    - https://www.iee.ihu.gr/exams-program/

```

Εικόνα 3.16 Απλό παράδειγμα

### 3.7.2 Σύνθετο παράδειγμα με διάλογο

Στη συνέχεια γίνεται αναφορά σε ένα πιο σύνθετο παράδειγμα, στο οποίο το VagBot, μετά την αναγνώριση της ερώτησης, προχωράει στη δημιουργία δικιάς του ερώτησης με σκοπό την πλήρη κατανόηση των ζητημάτων του χρήστη. Η λογική που χρησιμοποιείται είναι παρόμοια με το προηγούμενο παράδειγμα, η μόνη διαφορά είναι πως γίνεται επικοινωνία 2 σταδίων προκειμένου να δοθεί η τελική απάντηση στο χρήστη. Ανάλογα με την απάντηση που θα δώσει ο χρήστης, επιβεβαίωσης ή άρνησης, θα ενεργοποιηθεί το κατάλληλο story και θα δοθεί η ανάλογη απάντηση utter\_able\_diplwmatikh ή utter\_unable\_diplwmatikh.

```

- story: diplwmatikh able
  steps:
  - intent: am_i_able_diplwmatikh
  - action: utter_ask_dm_diplwmatikh
  - intent: affirm
  - action: utter_able_diplwmatikh
- story: diplwmatikh unable
  steps:
  - intent: am_i_able_diplwmatikh
  - action: utter_ask_dm_diplwmatikh
  - intent: deny
  - action: utter_unable_diplwmatikh
- intent: am_i_able_diplwmatikh
  examples: |
  - Μπορώ να παρω Διπλωματική;
  - Μπορώ να αναλάβω διπλωματική
  - μπορώ να παρω Διπλωματική εργασία;
  - μπορώ να παρω διπλωματική;
  - Μπορώ να παρώ διπλωματική
  - Μπορώ να παρώ διπλωματική?
  utter_ask_dm_diplwmatikh:

```

```

- text: |
  Έχεις συμπληρώσει 210 Δ.Μ;
  - Αν δεν γνωρίζεις, τότε μπορείς να μάθεις σε αυτό το σύνδεσμο
  - https://www.iee.ihu.gr/ects160/

- intent: affirm
examples: |
  - ναι
  - ν
  - Ναι!
  - Προφανως
  - Εχω
  - Σιγουρα
  - Φυσικα
- intent: deny
examples: |
  - οχι!
  - ο
  - οχι
  - Μπα
  - Δυσκολα
  - Δεν εχω
utter_able_diplwmatikh:
- text: |
  Ναι! Μπορείς να αναλάβεις Διπλωματική.
utter_unable_diplwmatikh:
- text: |
  Δυστυχώς όχι! Πρέπει να έχεις συμπληρώσει τουλάχιστον 210 Δ.Μ για να
  μπορείς αναλάβεις Διπλωματική εργασία

```

Εικόνα 3.17 Σύνθετο παράδειγμα με διάλογο

### 3.7.3 Σύνθετο παράδειγμα με εξόρυξη οντότητας

Το τρίτο παράδειγμα είναι σύνθετο καθώς το VagBot οφείλει, πέρα από την αναγνώριση της πρόθεσης του μηνύματος του χρήστη, να εξορύξει την οντότητα στην οποία αναφέρεται ο χρήστης. Εφόσον γίνουν οι δυο αυτές λειτουργίες επιτυχώς, τότε το VagBot δεν θα ανατρέξει να δώσει μια απάντηση από τις ήδη υπάρχουσες “utter\_...” αλλά θα προκαλέσει μια custom action, του αρχείου actions.py. Εκεί, ανάλογα με την οντότητα που εξήγαγε, θα γίνει η επιλογή της απάντησης.

```

- story: problem with service
  steps:
- intent: having_issue_with_a_service
examples: |
  - προβλήματα με το [thesis](yphresia)

```

```

- Έχω πρόβλημα με το [θεσης>{"entity": "yphresia", "value": "thesis"}
- προβλήματα με [apps](yphresia)
- πρόβλημα με το [αππς>{"entity": "yphresia", "value": "apps"}
- προβλήματα με [webmail](yphresia)
- προβλήματα με [email>{"entity": "yphresia", "value": "webmail"}
- Έχω πρόβλημα με το [mail>{"entity": "yphresia", "value": "webmail"}
- προβλήματα με [pithia](yphresia)
- Έχω πρόβλημα με το [πυθια>{"entity": "yphresia", "value": "pithia"}
- προβλήματα με [eudoxos](yphresia)
- Έχω πρόβλημα με τον [Ευδοξο>{"entity": "yphresia", "value": "eudoxos"}
- προβλήματα με [moodle](yphresia)
- Έχω πρόβλημα με το [μοοδλε>{"entity": "yphresia", "value": "moodle"}
- εχω πρόβλημα με το [apps](yphresia)
- action: action_resolve_issue

class ResolveIssue(Action):

    def name(self) -> Text:
        return "action_resolve_issue"

    def run(self, dispatcher: CollectingDispatcher, tracker: Tracker, domain:
Dict[Text, Any]) -> List[Dict[Text, Any]]:

        service_type = next(tracker.get_latest_entity_values('yphresia'),
None)

        if str(service_type) == 'apps':
            strg = 'Για προβλήματα σχετικά με το Apps επικοινωνήστε με το
noc@it.teithe.gr'
        elif str(service_type) == 'webmail':
            strg = 'Για προβλήματα σχετικά με το Webmail επικοινωνήστε με το
noc@it.teithe.gr ή στο https://helpdesk.the.ihu.gr'
        elif str(service_type) == 'thesis':
            strg = 'Για προβλήματα σχετικά με το Thesis επικοινωνήστε με τον
Κυριάκο Τσιακμάκη στο ktsiak@physics.auth.gr'
        elif str(service_type) == 'pithia':
            strg = 'Για προβλήματα σχετικά με το Πυθία επικοινωνήστε με την
γραμματεία του τμήματος info@iee.ihu.gr'
        elif str(service_type) == 'eudoxos':
            strg = 'Για πληροφορίες σχετικά με τον Εύδοξο: γραμματεία τμήματος
info@iee.ihu.gr, /n για τεχνικά θέματα: https://helpdesk.the.ihu.gr/,
noc@the.ihu.gr'
        elif str(service_type) == 'moodle':
            strg = 'Για προβλήματα πρόσβασης στα περιεχόμενα ενός μαθήματος
στο Moodle επικοινωνήστε με τον εκάστοτε υπεύθυνο καθηγητή'

```

```

else:
    strg = 'Συγγνώμη δεν κατάλαβα με ποια υπηρεσία υπάρχει πρόβλημα.
Μπορείτε να αναδιατυπώσετε την ερώτησή σας ή να επικοινωνήσετε με την
γραμματεία του τμήματος info@iee.ihu.gr'
    dispatcher.utter_message(strg)
return []

```

Εικόνα 3.18 Σύνθετο παράδειγμα με εξόρυξη οντότητας

### 3.7.4 Σύνθετο παράδειγμα με εξόρυξη οντότητας και αναζήτηση σε βάση δεδομένων.

Το τελευταίο παράδειγμα αποτελεί το πιο σύνθετο από τα παραπάνω, καθώς το VagBot μετά την αναγνώριση της προθέσεως της ερώτησης, οφείλει να κάνει αναγνώριση δύο οντοτήτων (yphresia, professor) και να ανατρέξει στη βάση δεδομένων, όπου θα αναζητήσει βάσει της οντότητας καθηγητή που εξόρυξε. Η σύνδεση πραγματοποιείται με την εντολή `mysql.connector.connect`, δίνοντας σαν παράμετρο τα διαπιστευτήρια σύνδεσης της βάσης δεδομένων. Στη συνέχεια, πραγματοποιείται μια προσαρμογή στην οντότητα καθηγητή που έχει εξαχθεί, προκειμένου να αποφευχθεί το ενδεχόμενο ορθογραφικών λαθών και των αλλαγών στις καταλήξεις των πτώσεων. Έτσι, για παράδειγμα, αν ο χρήστης ρωτήσει “ποιο είναι το mail του αδαμηδη”, ο entity extractor θα εξάγει δυο οντότητες, την `yphresia:webmail` και το `professor:αδαμηδη`. Την οντότητα `professor` θα την περάσει από την προσαρμογή όπου θα μετατραπεί από “αδαμηδη” => “αδαμ\_δ”. Έπειτα, θα αναζητήσει στην βάση δεδομένων εκτελώντας το query `"SELECT prof,mail FROM courseinfo where UPPER(prof) LIKE UPPER('%"+ “αδαμ_δ” +"%")`. Αξίζει να σημειωθεί πως για την αποφυγή διαφοροποιήσεων σε χαρακτήρες όσον αφορά κεφαλαία ή πεζά χρησιμοποιείται η συνάρτηση `UPPER` όπου προσαρμόζει σε κεφάλαια το πεδίο της βάσης όπου θα ψάξει αλλά και την οντότητα. Αν γίνουν όλα τα προαναφερθέντα βήματα σωστά και υπάρχει ο καθηγητής που ρώτησε ο χρήστης, τότε το VagBot θα απαντήσει με τα στοιχεία του καθηγητή, αλλιώς θα βγάλει ανάλογο μήνυμα αποτυχίας.

```

- story: professor info
  steps:
  - intent: whats_the_professors_info
  - action: action_get_professor_info

- intent: whats_the_professors_info
  examples: |
    - ποιο είναι το [webmail] (yphresia) του [Αδαμίδης Παναγιώτης] (professor)
    .

class GetProfInfo(Action):
    def name(self) -> Text:
        return "action_get_professor_info"
    def run(self, dispatcher: CollectingDispatcher, tracker: Tracker, domain: Dict[Text, Any]) -> List[Dict[Text, Any]]:
        mydb = mysql.connector.connect(user='VagBot',
                                       password='VagBot123',
                                       host='localhost',
                                       database='VagBot')

```

```

mycursor = mydb.cursor()
professor = next(tracker.get_latest_entity_values('professor'),
None)

profName=str(professor)
profNameCut=profName[:-1]
# \ = line continuation
profNameSqlReady=profNameCut.replace("ι", "_")\
    .replace("η", "_")\
    .replace("υ", "_")\
    .replace("ο", "_")\
    .replace("ω", "_")\
    .replace("νν", "ν%")\
    .replace("λλ", "λ%")\
    .replace("μμ", "μ%")\
    .replace("ππ", "π%")\
    .replace("κκ", "κ%")\
    .replace("σσ", "σ%")\
    .replace("ττ", "τ%")\
    .replace("γγ", "γ%")\
    .replace("γκ", "γ%")

query = "SELECT prof,mail FROM courseinfo where UPPER(prof) LIKE
UPPER('%"+profNameSqlReady+"%')"
mycursor.execute(query)
myresult = mycursor.fetchone()
if(myresult is not None):
    strg = 'Ο/Η κ.' +str(myresult[0])+' δέχεται email στο: '
+str(myresult[1])
else:
    strg = 'Δεν βρέθηκαν αποτελέσματα για το όνομα που δώσατε.'

dispatcher.utter_message(strg)
mydb.close()
return []

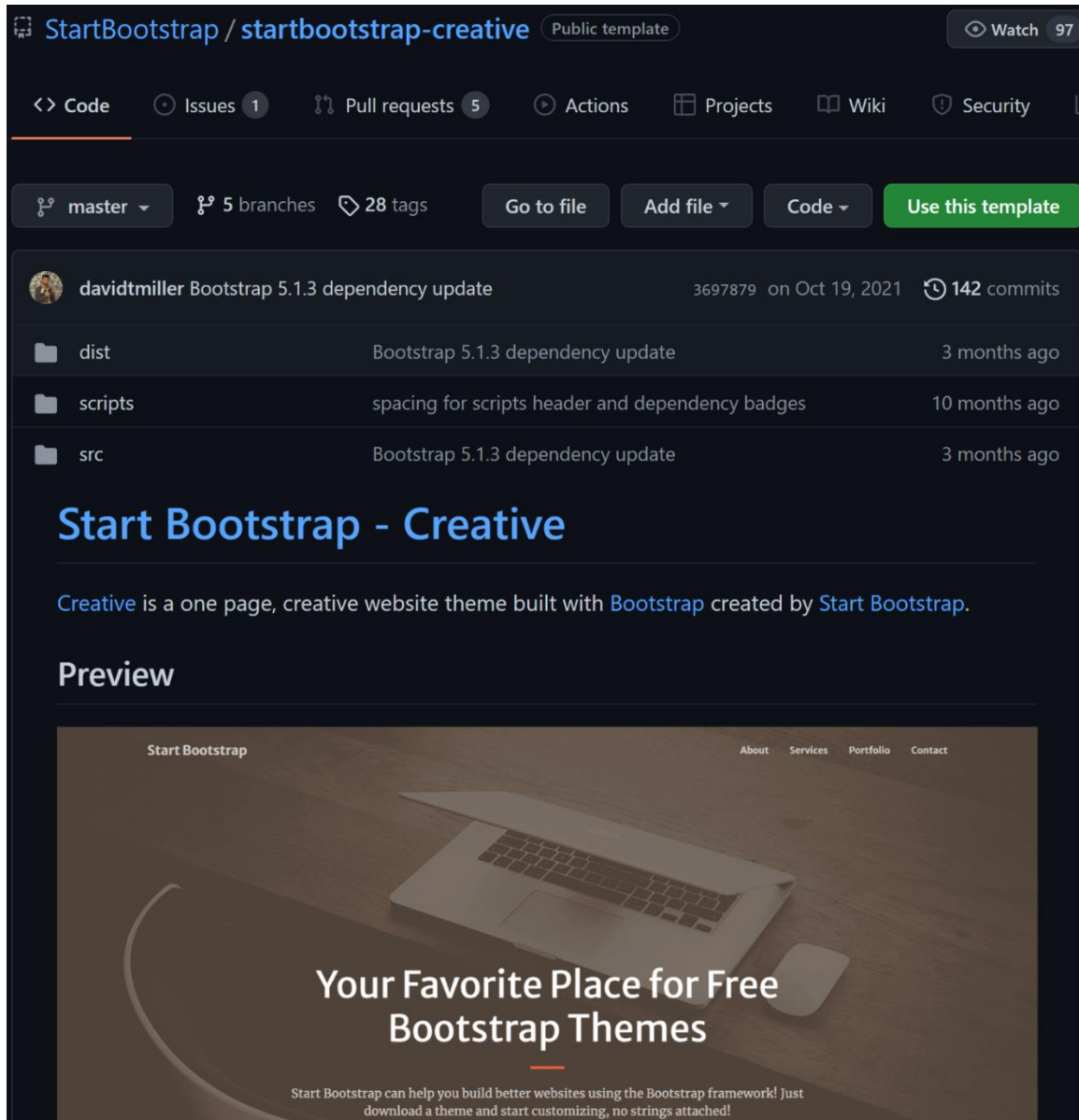
```

Εικόνα 3.19 Σύνθετο παράδειγμα με εξόρυξη οντότητας και αναζήτηση σε βάση δεδομένων.

Τέλος, έχοντας συμπληρώσει όλα τα προαναφερθέντα στοιχεία όπως stories, intents, rules, pipelines κ.λπ. μπορεί να ξεκινήσει η εκπαίδευση του μοντέλου. Η εκπαίδευση θα πραγματοποιηθεί μέσω των συναρτήσεων εκπαίδευσης σύμφωνα με την παραμετροποίηση που δόθηκε στο αρχείο config.yml. Κάθε αλλαγή που γίνεται, είτε στα δεδομένα, είτε στην παραμετροποίηση θα πρέπει να γίνει επανεκπαίδευση του μοντέλου εκτελώντας την εντολή: `rasa train`.

## Κεφάλαιο 4ο: Ανάπτυξη VagFront

Προκειμένου να συζητήσει κάποιος χρήστης με το VagBot ήταν αναγκαίο πρώτα να φτιαχτεί μια front end πλατφόρμα. Για την υλοποίηση της διεπαφής, όπου θα αλληλεπιδρά ο χρήστης χρησιμοποιήθηκε μια σελίδα bootstrap. Η λήψη έγινε από το βασικό λογαριασμό που διατίθεται στο GitHub (<https://github.com/startbootstrap/startbootstrap-creative>). Ο λόγος, για τον οποίο επιλέχθηκε μια έτοιμη σελίδα, είναι επειδή η συγκεκριμένη χρησιμοποιεί μόνο HTML, Javascript και CSS, που σημαίνει πως την κάνει να είναι εξαιρετικά απλή στην ανάπτυξη και υλοποίηση της.



Εικόνα 4.1 Bootstrap GitHub repository  
Πηγή: [github.com/StartBootstrap](https://github.com/StartBootstrap)

Εκτελώντας την εντολή `git clone https://github.com/StartBootstrap/startbootstrap-creative.git` γίνεται κλωνοποίηση της υλοποιημένης σελίδας στον Η/Υ όπου πρόκειται να γίνει η ανάπτυξη του VagFront. Για την επεξεργασία του project χρησιμοποιήθηκε το πρόγραμμα Visual Studio Code και όπως φαίνεται στην παρακάτω εικόνα, το αρχείο `index.html` περιέχει τον κώδικα της σελίδας Bootstrap και όλα τα απαραίτητα στοιχεία που δομούν έναν ιστότοπο.

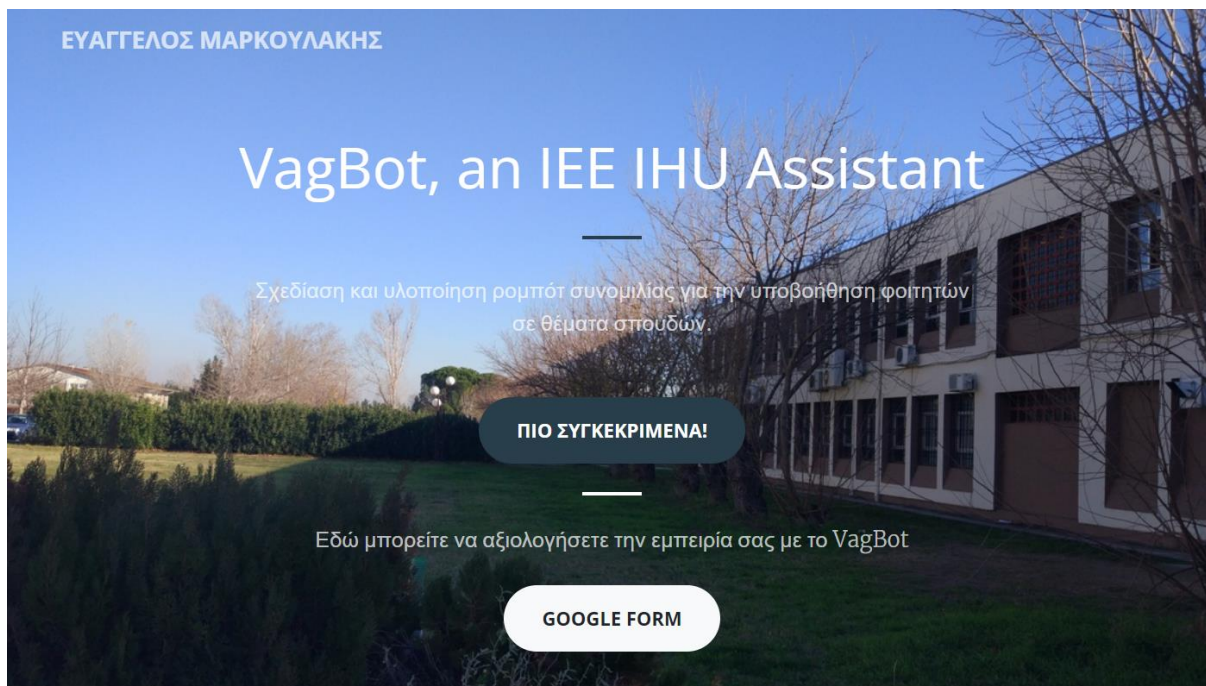
Εικόνα 4.2 Προκαθορισμένος κώδικας σελίδας Bootstrap.

Το αποτέλεσμα του παραπάνω κώδικα φαίνεται στην επόμενη εικόνα, όπου γίνεται εύκολα αντιληπτή η δόμηση της ιστοσελίδας. Στην κορυφή υπάρχει μια μπάρα πλοήγησης, στη συνέχεια η αρχική σελίδα και ύστερα εμφωλευμένα στην ίδια σελίδα σε μορφή (SPA-Single page application) ακόμα τέσσερα τμήματα.



Εικόνα 4.3 Προκαθορισμένη Bootstrap σελίδα  
Πηγή: [github.com/StartBootstrap](https://github.com/StartBootstrap)

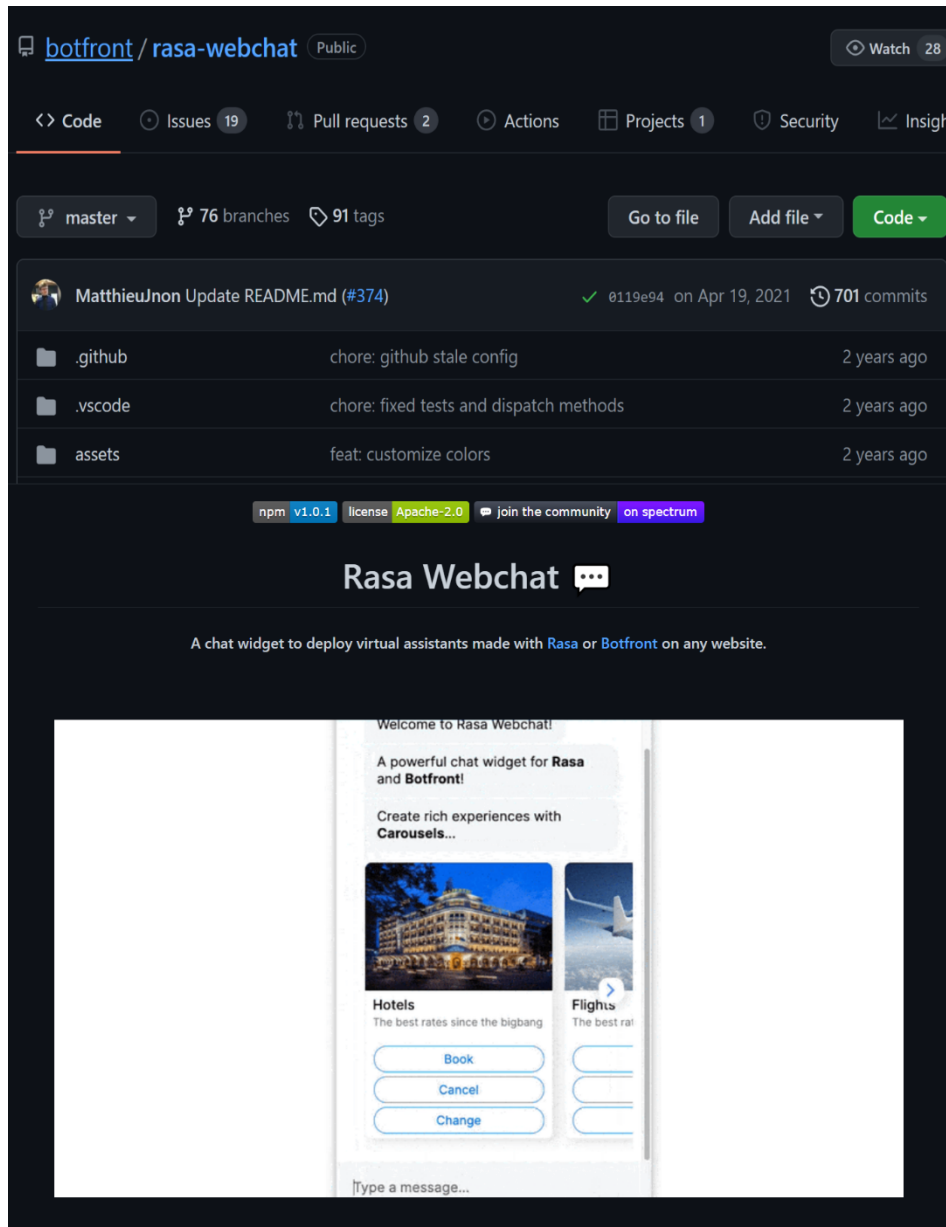
Έχοντας την σελίδα έτοιμη προς επεξεργασία, χρειάστηκε να πραγματοποιηθούν αλλαγές σε ορισμένα στοιχεία, όπως τίτλος, κείμενα, χρώματα, φωτογραφίες κ.λπ. προκειμένου να φτάσει η σελίδα στην επιθυμητή μορφή και να είναι έτοιμη να φιλοξενήσει, σε επόμενα βήματα, το chatbot. Έτσι, το αποτέλεσμα, σύμφωνα με τις αλλαγές που πραγματοποιήθηκαν, φαίνεται στην παρακάτω εικόνα.



Εικόνα 4.4 VagFront πριν την υλοποίηση του VagBot

Πέρα από την άμεση υλοποίηση της σελίδας, μπορεί κανείς να διαπιστώσει περεταίρω θετικά χαρακτηριστικά υλοποίησης μια σελίδας Bootstrap, όπως animations, γεωμετρία και τα οποία δίνουν ένα αξιόλογο οπτικό αποτέλεσμα προς το χρήστη. Έχοντας υλοποιημένη τη σελίδα που θα φιλοξενήσει το VagBot, μένει να αναπτυχθεί το γραφικό στοιχείο, με το οποίο θα μπορεί κάποιος χρήστης να ανταλλάσσει μηνύματα.

Έπειτα από έρευνα, ο καλύτερος τρόπος για να ενσωματωθεί το VagBot στη σελίδα, ήταν το module `rasa-webchat` (<https://github.com/botfront/rasa-webchat>). Το Rasa WebChat είναι ένα γραφικό στοιχείο συνομιλίας (widget), το οποίο βοηθάει στην ανάπτυξη εικονικών βοηθών σε οποιονδήποτε ιστότοπο.



Εικόνα 4.5 Rasa webchat Repository

Πηγή: [github.com/botfront](https://github.com/botfront)

Η ενσωμάτωση αυτού του γραφικού στοιχείου στην σελίδα πραγματοποιήθηκε χρησιμοποιώντας τη βιβλιοθήκη jQuery. Η jQuery βοηθά στην απλοποίηση και την τυποποίηση των αλληλεπιδράσεων μεταξύ κώδικα JavaScript και στοιχείων HTML. Η JavaScript επιτρέπει στους ιστότοπους να είναι διαδραστικοί και δυναμικοί και η jQuery είναι ένα εργαλείο, που απλοποιεί αυτή τη διαδικασία. Επίσης η jQuery βοηθάει σε πολλές περίπλοκες λειτουργίες της JavaScript, όπως κλήσεις AJAX (Asynchronous JavaScript And XML) και χειρισμό του DOM(Document Object Model), δηλαδή των στοιχείων μιας σελίδας.

```

33 <body id="page-top">
34
35 <div id="webchat">
36 </div>
37 <script src="https://cdn.jsdelivr.net/npm/rasa-webchat/lib/index.min.js"></script>
38 <link rel="stylesheet" type="text/css" href="https://">
39 <script>!function () {
40     let e = document.createElement("script"),
41         t = document.head || document.getElementsByTagName("head")[0];
42     (e.src =
43         "https://cdn.jsdelivr.net/npm/rasa-webchat@1.0.1/lib/index.js"),
44     (e.async = !0),
45     (e.onload = () => {
46         window.WebChat.default(
47             { ...
85             null
86         });
87     }),
88     t.insertBefore(e, t.firstChild);
89 });
90 </script>
91

```

Εικόνα 4.6 Υλοποίηση του Rasa Webchat

Όπως φαίνεται στην παραπάνω εικόνα, εντός του σώματος (body) της σελίδας δημιουργήθηκε ένα καινούργιο τμήμα div (Division tag) με κλειδί “webchat”. Αυτό το τμήμα, κατά τη φόρτωση της ιστοσελίδας θα είναι κενό, μέχρι να πραγματοποιηθεί σύνδεση με τον εξυπηρετητή του VagBot. Αφότου πραγματοποιηθεί η σύνδεση, το τμήμα “webchat” θα συμπληρωθεί με το γραφικό στοιχείο του chatbot. Με σκοπό την επίτευξη επικοινωνίας με τον εξυπηρετητή του VagBot, πρέπει πρώτα να γίνει κατάλληλη παραμετροποίηση αυτού του στοιχείου. Υπάρχουν περισσότερες από είκοσι ιδιότητες, οι οποίες επηρεάζουν την λειτουργία και την οπτική εμφάνιση του γραφικού στοιχείου, μερικές από τις οποίες φαίνονται στην παρακάτω εικόνα.

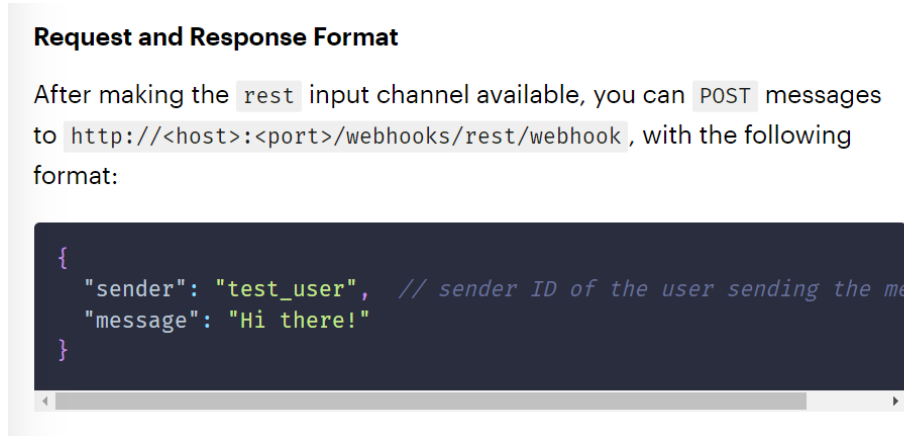
```

window.WebChat.default(
  {
    selector: "#webchat",
    initPayload: "/get_started",
    customData: { "language": "el" },
    socketUrl: "https://8588-185-44-144-54.eu.ngrok.io",
    socketPath: "/socket.io/",
    title: "VagBot",
    tooltip: true,
    subtitle: "Powered by IEE IHU",
    connectingText: "waiting for server...",
    profileAvatar: "img/IEE_logo_v2.svg",
    openLauncherImage: "img/chatbot.svg",
    params: { "storage": "session" },
    mainColor: "#0095A4",
    userBackgroundColor: "#283e4a",
    userTextColor: "#cde9ce",
    showMessageDate: false,
    inputTextFieldHint: "Πληκτρολογήστε μια ερώτηση...",
    displayUnreadCount: true,
    showCloseButton: true,
    showFullScreenButton: false,
  },
  null
);
}),
t.insertBefore(e, t.firstChild);

```

Εικόνα 4.7 Παραμετροποίηση του Rasa Webchat

Η σημαντικότερη εξ αυτών, είναι η “socketUrl”, η οποία ορίζει τη διεύθυνση, στην οποία θα αποστέλλει αιτήματα η σελίδα VagFront. Τα αιτήματα είναι μεθόδου Post του πρωτοκόλλου REST και η δομή τους θα είναι ένα μήνυμα τύπου Json, όπως φαίνεται στην επόμενη εικόνα. Η μεταβλητή της διεύθυνσης συμπληρώθηκε με το URL, το οποίο προέκυψε από το εργαλείο ngrok και εξηγείται στην επόμενη παράγραφο.



Εικόνα 4.8 Αποδεκτή μορφή αιτημάτων  
Πηγή: rasa.com

Στην περίπτωση του VagBot, η φυσική τοποθεσία της σελίδας έπρεπε να βρίσκεται κάπου που θα επιτρέπεται η πρόσβαση σε κάθε επισκέπτη. Έτσι, επόμενο και σημαντικότερο βήμα ήταν να δημοσιευτεί η σελίδα σε κάποιον εξυπηρετητή, ώστε να είναι ενεργή, διαθέσιμη και προσβάσιμη για όλους τους φοιτητές του τμήματος. Η ανάγκη αυτή καλύφθηκε χρησιμοποιώντας τον ιστότοπο της προσωπικής σελίδας που διαθέτει κάθε φοιτητής στους server της τμήματος (<https://users.iee.ihu.gr/~it174966>). Η πρόσβαση στο προσωπικό φάκελο του τμήματος πραγματοποιήθηκε χρησιμοποιώντας το εργαλείο Putty και τα διαπιστευτήρια της πλατφόρμας APPS. Το Putty, είναι μια εφαρμογή με ιδιότητες SSH (Secure Shell), δηλαδή ένα ασφαλές δικτυακό πρωτόκολλο, το οποίο επιτρέπει τη μεταφορά δεδομένων μεταξύ δύο υπολογιστών. Η σελίδα που μόλις αναπτύχθηκε τοπικά, έπρεπε να μεταφερθεί στον προσωπικό φάκελο `public_html` του εξυπηρετητή του τμήματος, ο οποίος πρώτα έπρεπε να δημιουργηθεί. Η δημιουργία αυτού του φακέλου που θα περιέχει το project VagFront είναι η πρώτη εντολή στην παρακάτω εικόνα.

```
it174966@users:~$ mkdir public_html
it174966@users:~$ chmod 711 public_html
```

Εικόνα 4.9 Εντολές `mkdir` και `chmod`

Η δεύτερη εντολή, `chmod`, ορίζει τα δικαιώματα πρόσβασης στον αντίστοιχο φάκελο. Στη περίπτωση του VagBot χρειάστηκε να δοθούν τα δικαιώματα 711, το οποίο αποκωδικοποιείται ως εξής:

- Ο κάτοχος μπορεί να διαβάζει, να γράφει και να εκτελεί,
- Η ομάδα στην οποία ανήκει δεν μπορεί να διαβάσει, να γράφει και να εκτελέσει,
- Όλοι οι υπόλοιποι δεν μπορούν να διαβάσουν, να γράψουν και να εκτελέσουν.

Με στόχο το project VagFront να μεταφερθεί στο φάκελο `public_html`, χρησιμοποιήθηκε η εντολή `git clone`, η οποία φαίνεται στην παρακάτω εικόνα, καθώς το project ήταν ήδη κοινοποιημένο στη πλατφόρμα GitHub. Με αυτό τον τρόπο η σελίδα που βρισκόταν τοπικά, βρίσκεται πλέον σε εξυπηρετητή και είναι προσβάσιμη από οπουδήποτε.

```

it174966@users: ~
Using username "it174966".
it174966@users.it.teithe.gr's password:
Linux users.iee.ihu.gr 4.19.0-18-amd64 #1 SMP Debian 4.19.208-1 (2021-09-29) x86_64

Welcome to

International Hellenic University
Department of Information and Electronic Engineering

We are running Debian GNU/Linux

users.iee.ihu.gr
(195.251.123.31)

**** NOTE: Please use users.iee.ihu.gr instead of users.it.teithe.gr from now on ****

Last login:
it174966@users:~$ git clone https://github.com/vagmark30/RasaFront.git

```

Εικόνα 4.10 SSH tunnel και εντολή git clone

Επίσης, ο rasa server που έτρεχε το VagBot, έπρεπε να είναι και αυτός ενεργός, διαθέσιμος και προσβάσιμος στο διαδίκτυο. Προκειμένου να επιτευχθεί αυτό, χρησιμοποιήθηκε το εργαλείο ngrok, το οποίο έδωσε πρόσβαση στην θύρα 5005, δρομολογώντας κάθε αίτημα προς την διεύθυνση (<http://8588-185-44-144-54.eu.ngrok.io>) στο τοπικό μηχανήμα, όπου έτρεχε ο εξυπηρετητής του VagBot.

```

PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE
ngrok by @inconshreveable

Session Status      online
Account             Evangelos Markoulakis (Plan: Free)
Version             2.3.40
Region              Europe (eu)
Web Interface       http://127.0.0.1:4040
Forwarding           http://8588-185-44-144-54.eu.ngrok.io -> http://localhost:5005
Forwarding          https://8588-185-44-144-54.eu.ngrok.io -> http://localhost:5005

Connections
  ttl    opn    rt1    rt5    p50    p90
  813    10     0.06  0.05  5.99  54.85

HTTP Requests
-----
POST /socket.io/      200 OK
POST /socket.io/      200 OK
GET  /socket.io/       400 Bad Request

```

Εικόνα 4.11 Το ngrok

Χρησιμοποιώντας την δημόσια διεύθυνση Forwarding IP του ngrok, η front end πλατφόρμα μπορεί πλέον να ανταλλάσσει ερωτήσεις και απαντήσεις με το VagBot. Ένα ακόμη βήμα προκειμένου να επιτευχθεί επικοινωνία, ήταν να γίνουν προσβάσιμες οι υποδοχές του καναλιού socket.io. Το socket.io υλοποιείται χρησιμοποιώντας websockets σε πραγματικό χρόνο. Προκειμένου να χρησιμοποιηθεί αυτό το κανάλι επικοινωνίας, πρέπει να προστεθεί στο αρχείο credentials.yml η εξής παραμετροποίηση.

```

1 # This file contains the credentials for the voice & chat platforms
2 # which your bot is using.
3 # https://rasa.com/docs/rasa/messaging-and-voice-channels
4
5 socketio:
6   user_message_evt: user_uttered
7   bot_message_evt: bot_uttered
8   session_persistence: true

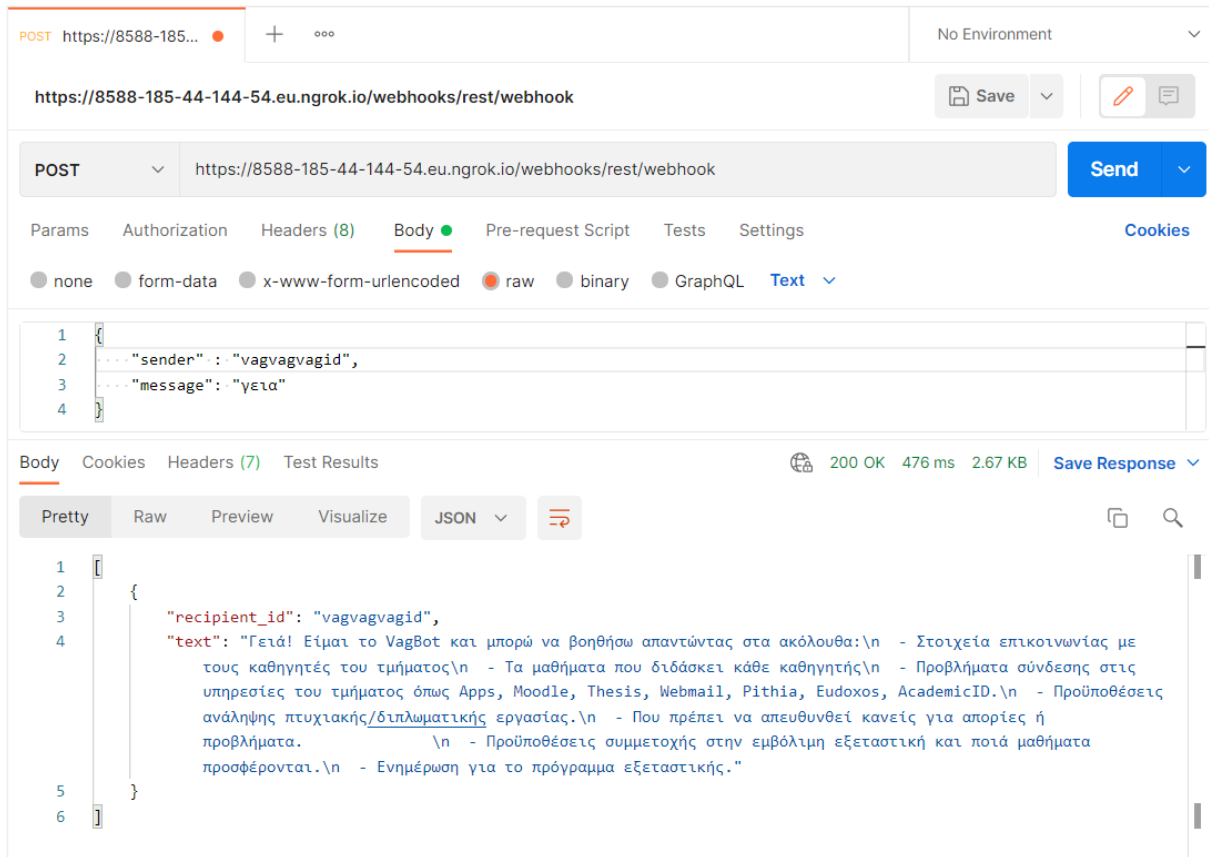
```

Εικόνα 4.12 Αρχείο credentials.yml

Οι μεταβλητές `user_message_evt: user_uttered` και `bot_message_evt: bot_uttered` ρυθμίζουν τα γεγονότα που χρησιμοποιούνται από το Framework της Rasa, όταν το chatbot ανταλλάσσει μηνύματα μέσω του socketio. Η τελευταία μεταβλητή, αφορά την διατήρηση ή μη των μηνυμάτων κατά το κλείσιμο της πλατφόρμας.

Προκειμένου ο εξυπηρετητής να απαντάει σε ερωτήσεις, για τις οποίες έχει γραφτεί ειδικός κώδικας στον αρχείο `Actions.py`, πρέπει να εκτελεστεί η εντολή `rasa run actions`. Αυτή η εντολή επιτρέπει να απαντηθούν ορισμένες ερωτήσεις, για τις οποίες απαιτείται προγραμματιστική λογική, που μόνο σε κώδικα να υλοποιηθεί. Επίσης, είναι υπεύθυνη για την επικοινωνία μεταξύ του VagBot και της MySQL βάσης δεδομένων.

Τέλος, με στόχο ο εξυπηρετητής rasa να δέχεται αιτήματα rest από οποιαδήποτε προέλευση, χρειάστηκε κατά την εκκίνησή του, δηλαδή με την εκτέλεση της εντολής `rasa run -models` να του δοθούν τα γνώρισμα «enable-api» και «cors "\*"» (Cross-Origin Resource Sharing). Ως προεπιλογή, ο εξυπηρετητής rasa δεν δέχεται αιτήσεις και κλήσεις από οποιαδήποτε διεύθυνση. Το πρώτο γνώρισμα εκκινεί το API του εξυπηρετητή, ώστε να δέχεται αιτήματα. Σε αυτό το σημείο όμως υπάρχει ένα επίπεδο ασφαλείας το οποίο αν δεν δοθεί το δεύτερο γνώρισμα, ο server δεν θα δέχεται αιτήσεις από καμία διεύθυνση πέρα της τοπικής (localhost). Για αυτό το λόγο, χρησιμοποιήθηκε αυτό το γνώρισμα με παράμετρο «\*» προκειμένου να επιτρέψει την επικοινωνία από οποιαδήποτε προέλευση. Έχοντας υλοποιήσει όλα τα προηγούμενα βήματα που αναλύθηκαν, έγιναν αρκετές δοκιμές προκειμένου να ελεγχθεί αν υπάρχει το επιθυμητό αποτέλεσμα. Οι δοκιμές εκτελέστηκαν μέσω της εφαρμογής Postman και δυο εκ των οποίων φαίνονται στις παρακάτω εικόνες. Το πρωτόκολλο όπως προαναφέρθηκε ήταν το REST και η μέθοδος ήταν Post, ενσωματώνοντας τα ανάλογα μηνύματα στο σώμα των αιτήσεων.



Εικόνα 4.13 Έλεγχος αιτημάτων μέσω Postman 1

Τα συμπεράσματα στα οποία μπορεί να καταλήξει κανείς από το πρώτο αίτημα της εικόνας 5.13 είναι πως:

- Το URL του ngrok δρομολογεί καταλλήλως τα μηνύματα
- Ο δρομολογητής δέχεται μηνύματα στην θύρα 5005
- Ο ταξινομητής καταλαβαίνει ορθός την πρόθεση του μηνύματος του χρήστη και του απαντάει αναλόγως

Στο δεύτερο αίτημα μπορεί κανείς να εξάγει επιπλέον συμπεράσματα διότι η απάντηση του δρομολογητή για το συγκεκριμένο αίτημα περιέχει επιπρόσθετη πληροφορία. Πρώτον, ο ταξινομητής απάντησε πως η πρόθεση του μηνύματος του χρήστη, με 99.9% βεβαιότητα, ταξινομείται στο intent “whats\_the\_professors\_info”. Έπειτα, έχει εξάγει δυο οντότητες, με 99.9% βεβαιότητα και τις δυο, την “yghresia” και την “professor”. Την πρώτη μάλιστα την έχει αντιστοιχίσει με το συνώνυμο της, επειδή ο χρήστης έγραψε “μαιλ” αντί για το ορθό στο VagBot “webmail”.

The screenshot shows the Postman interface for a POST request to the endpoint `https://8588-185-44-144-54.eu.ngrok.io/model/parse`. The request body is a JSON object with a `text` field containing the Greek phrase "μιαλ αδαμιδη". The response is also a JSON object, which includes an `intent` object with `id`, `name`, and `confidence` fields, and an `entities` array. The first entity is `"yphresia"` (webmail) and the second is `"professor"` (αδαμιδη). The `intent_ranking` array also contains the same intent information.

```

1 {
2   "text": "μιαλ αδαμιδη"
3 }

Body Cookies Headers (7) Test Results
Pretty Raw Preview Visualize JSON
1 {
2   "text": "μιαλ αδαμιδη",
3   "intent": {
4     "id": 6593137652476477762,
5     "name": "whats_the_professors_info",
6     "confidence": 0.9999912381
7   },
8   "entities": [
9     {
10    "entity": "yphresia",
11    "start": 0,
12    "end": 4,
13    "confidence_entity": 0.9996720552,
14    "value": "webmail",
15    "extractor": "DIETClassifier",
16    "processors": [
17      "EntitySynonymMapper"
18    ]
19    },
20    {
21    "entity": "professor",
22    "start": 5,
23    "end": 12,
24    "confidence_entity": 0.999253571,
25    "value": "αδαμιδη",
26    "extractor": "DIETClassifier"
27    }
28  ],
29  "intent_ranking": [
30    {
31    "id": 6593137652476477762,
32    "name": "whats_the_professors_info",

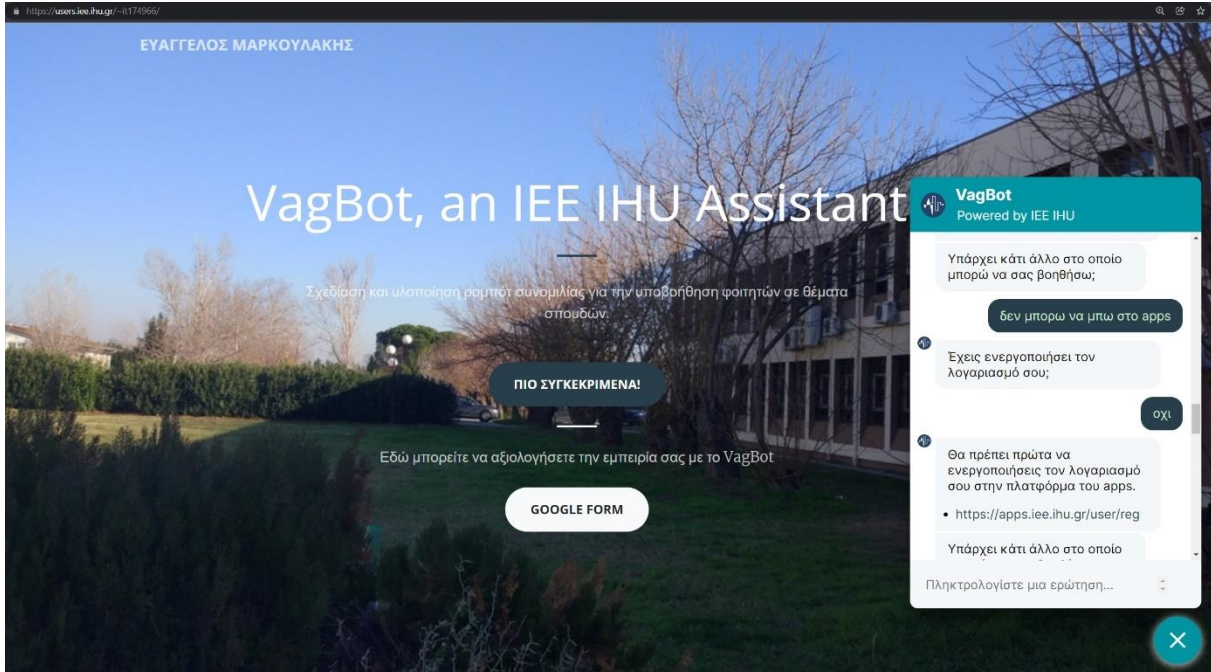
```

Εικόνα 4.14 Έλεγχος αιτημάτων μέσω Postman 2

Χρησιμοποιώντας όλα τα παραπάνω εργαλεία αναπτύχθηκε η σελίδα με την οποία θα αλληλεπιδράει ο χρήστης. Το γραφικό στοιχείο του VagBot έχει τοποθετηθεί κάτω δεξιά όπως είναι το πιο σύνηθες για κάθε chatbot. Το αποτέλεσμα αυτής της προσπάθειας φαίνεται στην επόμενη ενότητα.

## Κεφάλαιο 5ο: Αποτέλεσμα

Το τελικό αποτέλεσμα είναι μια πλήρως λειτουργική σελίδα με δυνατότητες επικοινωνίας μέσω διαλόγου με το VagBot. Το URL της πλατφόρμας όπως φαίνεται είναι το <https://users.iee.ihu.gr/~it174966/>.



Εικόνα 5.1 VagFront, πλατφόρμα του VagBot

Παρακάτω θα παρουσιαστούν με εικόνες όλες οι πιθανές ιστορίες που πρόκειται να συναντήσει κάποιος χρήστης συνομιλώντας με το VagBot.



Εικόνα 5.2 Παράδειγμα συνομιλιών 1

**VagBot**  
Powered by IEE IHU

**εχω προβλημα με το thesis**

Για προβλήματα σχετικά με το Thesis επικοινωνήστε με τον Κυριάκο Τσιακμάκη στο [ktsiak@physics.auth.gr](mailto:ktsiak@physics.auth.gr)

Υπάρχει κάτι άλλο στο οποίο μπορώ να σας βοηθήσω;

**ποιος είναι υπευθυνος για την πυθια**

Το ριθία και οι συνδεδεμένες εφαρμογές τους διαχειρίζονται από το Κέντρο Δικτύου της πανεπιστημιούπολης Σίνδου (<http://www.noc.teithe.gr/>).

Υπάρχει κάτι άλλο στο οποίο μπορώ να σας βοηθήσω;

**ποιο είναι το προγραμμα εξεταστικής;**

Το πρόγραμμα εξεταστικής μπορείτε να το βρείτε σε αυτό το σύνδεσμο:

- <https://www.iee.ihu.gr/exams-program/>

**VagBot**  
Powered by IEE IHU

**εχω ξεχασει τον κωδικο μου στο απις**

Για ανάκτηση του κωδικού του apps μπορείτε να το κάνετε αυτόματα από την πλατφόρμα <https://apps.iee.ihu.gr/user/reset>

Υπάρχει κάτι άλλο στο οποίο μπορώ να σας βοηθήσω;

**δεν μπορω να μπω στο apps**

Έχεις ενεργοποιήσει τον λογαριασμό σου;

**οχι**

Θα πρέπει πρώτα να ενεργοποιήσεις τον λογαριασμό σου στην πλατφόρμα του apps.

- <https://apps.iee.ihu.gr/user/reg>

Υπάρχει κάτι άλλο στο οποίο μπορώ να σας βοηθήσω;

Εικόνα 5.3 Παράδειγμα συνομιλιών 2

**VagBot**  
Powered by IEE IHU

ποια ειναι τα κριτηρια για να δωσω εμβολιμη;

• Προϋπόθεση για την συμμετάσχει κανείς στην εμβόλιμη εξέταση ενός μαθήματος είναι να το έχει δηλώσει τουλάχιστον μια φορά στο παρελθόν και να έχει συμπληρώσει τουλάχιστον 160 ΠΜ.

Υπάρχει κάτι άλλο στο οποίο μπορώ να σας βοηθήσω;

ποσες δμ εχω;

• Μπορείς να μάθεις πόσες διδακτικές μονάδες εχεις σε αυτό το σύνδεσμο

- <https://www.iee.ihu.gr/ects160/>

Υπάρχει κάτι άλλο στο οποίο μπορώ να σας βοηθήσω;

**VagBot**  
Powered by IEE IHU

ποια μαθηματα προσφeronται στην εμβολιμη;

• Τα μαθήματα που προσφέρονται στην εμβόλιμη εξεταστική μπορείτε να τα βρείτε σε αυτό το σύνδεσμο

- <https://www.iee.ihu.gr/ects160/courses.html> Κατά κανόνα στις εμβόλιμες εξεταστικές προσφέρονται όλα τα μαθήματα κορμού και όλα τα μαθήματα που είναι υποχρεωτικά στις ομάδες μαθημάτων.

Υπάρχει κάτι άλλο στο οποίο μπορώ να σας βοηθήσω;

Εικόνα 5.4 Παράδειγμα συνομιλιών 3

## Κεφάλαιο 6ο: Αξιολόγηση

### 6.1 Σύνθεση ερωτηματολογίου

Το ερωτηματολόγιο χωρίστηκε σε δυο ενότητες. Οι ερωτήσεις της 1ης Ενότητας του ερωτηματολογίου είχαν ως σκοπό να σχηματισθεί μια εικόνα του χρήστη που το χρησιμοποίησε [15].

Σε ποιά κατηγορία φοιτητή ανοίκετε;

Προπτυχιακός

Μεταπτυχιακός

---

Είχατε ξανά εμπειρία επικοινωνίας με Chatbot;

Ναι

Όχι

---

Πιστεύετε πως είναι χρήσιμα τα Chatbot;

1      2      3      4      5

Καθόλου                                    Πάρα πολύ

---

Στο πλαίσιο της σχολής, πιστεύετε θα βοηθούσε μια πλατφόρμα chatbot για την απάντηση ερωτημάτων και επίλυση προβλημάτων;

1      2      3      4      5

Καθόλου                                    Πάρα πολύ

---

Θα επιλέγατε την επικοινωνία με chatbot πρώτου προβείτε στα κοινωνικά δίκτυα;

Ναι

Όχι

Ίσως

Εικόνα 6.1 Πρώτη ενότητα ερωτηματολογίου

Σκοπός των ερωτήσεων της 2ης Ενότητας του ερωτηματολογίου, ήταν η σφαιρική προσέγγιση αξιολόγησης ενός chatbot [15]. Προκειμένου να αξιολογηθεί επιτυχώς ένα chatbot πρέπει να τεθούν ερωτήματα βάσει των σημαντικότερων παραγόντων, οι οποίοι καθορίζουν την ποιότητά του. Οι παράγοντες που επιλέχθηκαν να εξεταστούν είναι οι εξής:

- Μέσος χρόνος συνομιλίας
- Δείκτης ικανοποίησης
- Δείκτης Επίδοσης
- Δείκτης μη απαντημένων ερωτήσεων
- Δείκτης αυτοεξυπηρέτησης

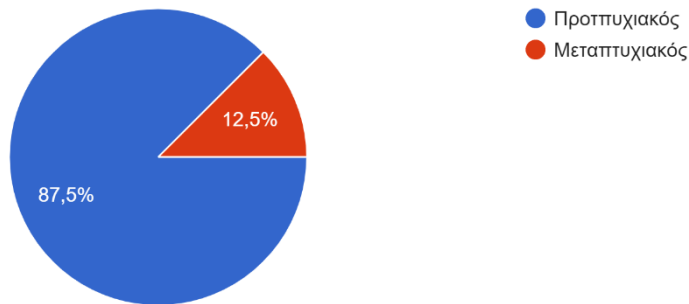
Η εμπειρία σας με το VagBot	
<p>Πόσες ερωτήσεις θέσατε στο VagBot;</p> <p><input type="radio"/> 1</p> <p><input type="radio"/> 2</p> <p><input type="radio"/> 3</p> <p><input type="radio"/> Πάνω απο 3</p>	<p>Ήταν απαραίτητη η επικοινωνία με ανθρώπινο παράγοντα προκειμένου να λυθεί το πρόβλημά σας.</p> <p><input type="radio"/> Ναι</p> <p><input type="radio"/> Όχι</p> <p><input type="radio"/> Ίσως</p>
<p>Πόση ώρα διαθέσατε συνομιλώντας με το VagBot.</p> <p><input type="radio"/> Εώς 5 λεπτά</p> <p><input type="radio"/> 5 εώς 10 λεπτά</p> <p><input type="radio"/> Πάνω απο 10 λεπτά</p>	<p>Υπήρξαν φορές που το VagBot δεν σας εδωσε καμια απαντηση;</p> <p><input type="radio"/> Ναι</p> <p><input type="radio"/> Όχι</p>
<p>Οι απαντήσεις που σας δώθηκαν ήταν ικανοποιητικές;</p> <p><input type="radio"/> Ναι</p> <p><input type="radio"/> Όχι</p> <p><input type="radio"/> Ίσως</p>	<p>Ποια είναι η γνώμη σας για την επίδοση του VagBot;</p> <p><input type="radio"/> Κακή</p> <p><input type="radio"/> Μέτρια</p> <p><input type="radio"/> Καλή</p> <p><input type="radio"/> Πολύ καλή</p>
<p>Κατάφερε το VagBot να λύσει κάποιο πρόβλημά σας;</p> <p><input type="radio"/> Ναι</p> <p><input type="radio"/> Όχι</p>	<p>Ποιές απο τις παρακάτω δυνατότητες θα επιθυμούσατε να συναντήσετε στο VagBot;</p> <p><input type="checkbox"/> Επιλογή αγγλικών ή άλλων γλωσσών</p> <p><input type="checkbox"/> Επιλογή φωνητικής συνομιλίας</p> <p><input type="checkbox"/> Περισσότερες πληροφορίες σχετικά με τα μαθήματα.</p> <p><input type="checkbox"/> Προσωποποιημένες πληροφορίες μετά απο σύνδεση με το apps</p> <p><input type="checkbox"/> Άλλο: _____</p>

Εικόνα 6.2 Δεύτερη ενότητα ερωτηματολογίου

## 6.2 Αποτελέσματα αξιολόγησης

Η πρώτη ερώτηση του ερωτηματολογίου αποτελεί ερώτημα αναγνώρισης της ιδιότητας του χρήστη. Από τις απαντήσεις που δόθηκαν είναι εμφανής η ανταπόκριση των φοιτητών προπτυχιακού επιπέδου, καθώς είναι αυτοί οι οποίοι χρειάζονται ένα εργαλείο όπως το chatbot.

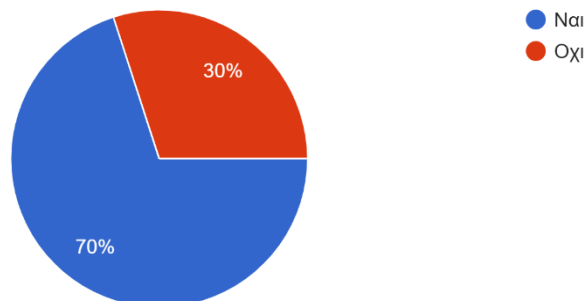
Σε ποιά κατηγορία φοιτητή ανοίκετε;  
40 απαντήσεις



Γράφημα 1: Κατηγορία φοιτητών

Η δεύτερη ερώτηση αφορά την εμπειρία των χρηστών με άλλα εργαλεία chatbot. Από τα παρακάτω αποτελέσματα γίνεται αντιληπτό πως οι περισσότεροι χρήστες είχαν χρησιμοποιήσει και άλλη φορά chatbot πριν από το VagBot. Είναι λογικό να υπάρχει τέτοιου είδους εμπειρία καθώς οι ευφείς συνομιλίες είναι μέρος της καθημερινότητας.

Είχατε ξανά εμπειρία επικοινωνίας με Chatbot;  
40 απαντήσεις

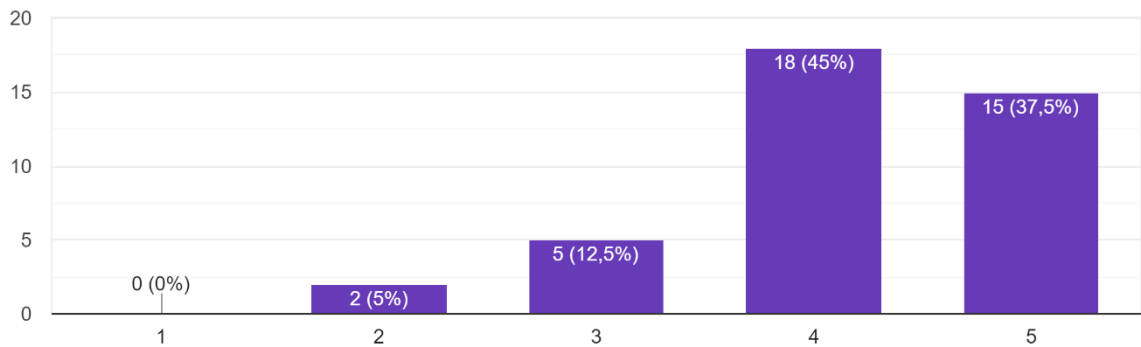


Γράφημα 2: Εμπειρία με chatbot

Σχετικά με τη χρησιμότητα των chatbot, οι περισσότεροι ερωτηθέντες φαίνεται να θεωρούν τη χρήση τέτοιου είδους εργαλείου αρκετά σημαντική καθώς σχεδόν οι μισοί βαθμολόγησαν το chatbot με τον αριθμό 4.

Πιστεύετε πως είναι χρήσιμα τα Chatbot;

40 απαντήσεις

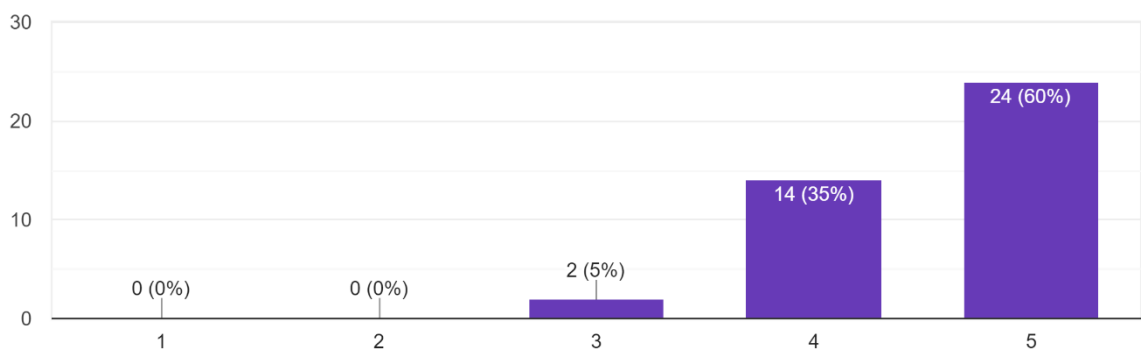


Γράφημα 3: Χρησιμότητα chatbot

Στην τέταρτη ερώτηση οι χρήστες έπρεπε να απαντήσουν σχετικά με τη βοήθεια που μπορεί να προσφέρει ένα chatbot για ερωτήματα και επίλυση αυτών που αφορούν τη σχολή. Οι απαντήσεις που δόθηκαν είναι πολύ ενθαρρυντικές καθώς το 95% θεώρησε πως η βοήθεια του chatbot είναι από αρκετά έως πολύ σημαντική.

Στο πλαίσιο της σχολής, πιστεύετε θα βοηθούσε μια πλατφόρμα chatbot για την απάντηση ερωτημάτων και επίλυση προβλημάτων;

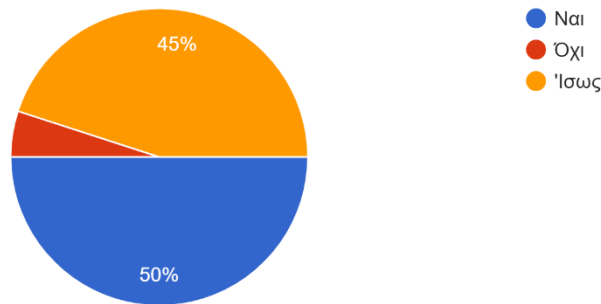
40 απαντήσεις



Γράφημα 4: Βοήθεια chatbot

Στην ερώτηση 5 οι χρήστες ρωτήθηκαν αν θα επέλεγαν το chatbot πριν χρησιμοποιούσαν κάποιο άλλο είδος κοινωνικού δικτύου. Από τα ποσοστά των απαντήσεων γίνεται αντιληπτή η προθυμία των χρηστών για την επιλογή του chatbot. Παρόλα αυτά, μεγάλο ποσοστό φαίνεται να είναι διστακτικό. Πολύ πιθανό το γεγονός αυτό να οφείλεται στο περιεχόμενο της ερώτησης.

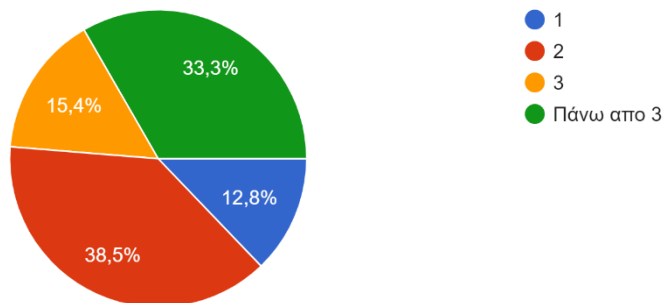
Θα επιλέγατε την επικοινωνία με chatbot πρώτου προβείτε στα κοινωνικά δίκτυα;  
40 απαντήσεις



Γράφημα 5: Επιλογή chatbot

Το ερώτημα 6 σχετίζεται με το πλήθος των ερωτήσεων που τέθηκαν στο VagBot. Από τα αποτελέσματα του ερωτηματολογίου φαίνεται πως οι χρήστες είχαν αλληλεπίδραση με το VagBot, καθώς το μεγαλύτερο ποσοστό ρώτησε από δύο ερωτήσεις και πάνω.

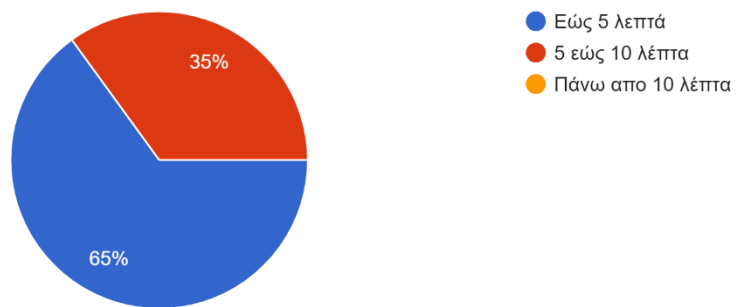
Πόσες ερωτήσεις θέσατε στο VagBot;  
39 απαντήσεις



Γράφημα 6: Πλήθος ερωτήσεων VagBot

Στην ερώτηση 7 οι χρήστες απάντησαν σχετικά με το χρόνο που αφιέρωσαν στο VagBot. Οι περισσότεροι διέθεσαν μέχρι πέντε λεπτά, ενώ κανένας χρήστης δεν συνομίλησε πάνω από 10 λεπτά. Το γεγονός αυτό φανερώνει την αποτελεσματικότητα και την αποδοτικότητα του VagBot.

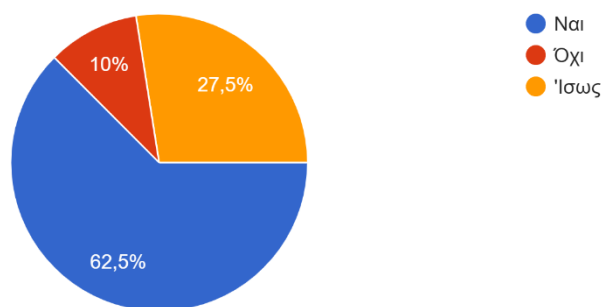
Πόση ώρα διαθέσατε συνομιλώντας με το VagBot.  
40 απαντήσεις



Γράφημα 7: Χρόνος διάθεσης στο VagBot

Σχετικά με την ερώτηση 8, φαίνεται πως οι χρήστες είναι αρκετά ευχαριστημένοι από τις απαντήσεις που δόθηκαν από το VagBot. Μόνο το 10% απάντησε αρνητικά σχετικά με την αποδοτικότητα του VagBot.

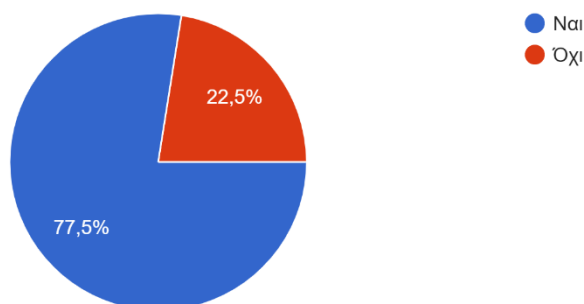
Οι απαντήσεις που σας δώθηκαν ήταν ικανοποιητικές;  
40 απαντήσεις



Γράφημα 8: Ικανοποίηση απαντήσεων

Στην ερώτηση 9 οι χρήστες απάντησαν στο αν το VagBot κατάφερε να λύσει το πρόβλημα τους. Τα αποτελέσματα είναι θετικά, καθώς το 77.5% απάντησε πως το VagBot τους βοήθησε.

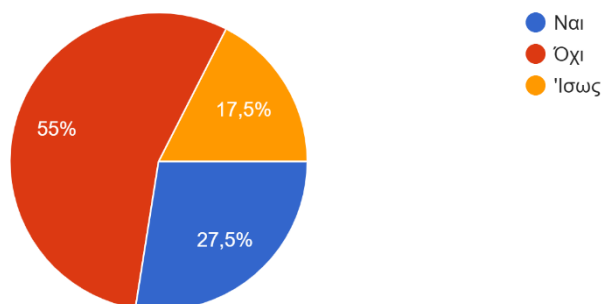
Κατάφερε το VagBot να λύσει κάποιο πρόβλημά σας;  
40 απαντήσεις



Γράφημα 9: Λύση προβλημάτων από VagBot

Η ερώτηση 10 σχετίζεται με την αναγκαιότητα παρέμβασης ανθρώπινου παράγοντα με σκοπό την επίλυση του προβλήματος. Μέσα από τα αποτελέσματα, φαίνεται πως πάνω από τους μισούς ερωτηθέντες δε χρειάστηκαν τη βοήθεια άλλου ατόμου προκειμένου να λυθεί το πρόβλημά τους.

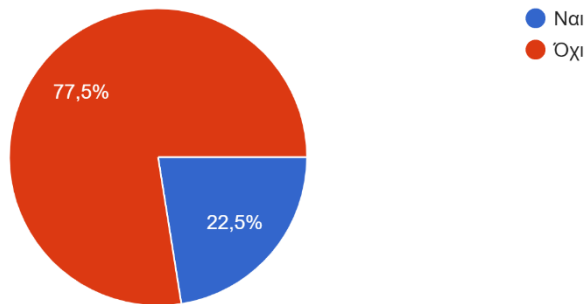
Ήταν απαραίτητη η επικοινωνία με ανθρώπινο παράγοντα προκειμένου να λυθεί το πρόβλημά σας.  
40 απαντήσεις



Γράφημα 10: Επέμβαση ανθρώπινου παράγοντα

Στην ενδέκατη ερώτηση γίνεται αντιληπτή η σωστή λειτουργικότητα του VagBot, αφού μόνο το 22.5 % δεν πήρε απάντηση από το εργαλείο.

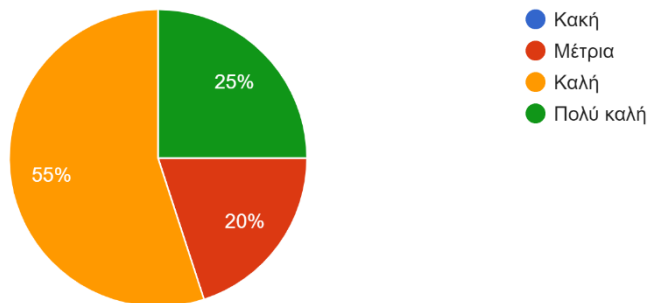
Υπήρξαν φορές που το VagBot δεν σας έδωσε καμία απάντηση;  
40 απαντήσεις



Γράφημα 11: Λειτουργικότητα VagBot

Όσον αφορά τη γνώμη των χρηστών για την επίδοση του VagBot (ερώτηση 12), οι ερωτηθέντες φαίνεται να είναι πολύ θετικοί απέναντι στο συγκεκριμένο chatbot, καθώς κανένας δε θεωρεί την επίδοση του κακή, ενώ το 25% την αξιολόγησε ως «πολύ καλή».

Ποια είναι η γνώμη σας για την επίδοση του VagBot;  
40 απαντήσεις



Γράφημα 12: Επίδοση VagBot

Καταλήγοντας στην τελευταία ερώτηση του ερωτηματολογίου, οι χρήστες είχαν τη δυνατότητα επιλογής των απαντήσεων τους μέσα από ένα πλήθος προεπιλεγμένων απαντήσεων, αλλά ταυτόχρονα μπορούσε κάποιος να γράψει και μια δική του.

Ποιές απο τις παρακάτω δυνατότητες θα επιθυμούσατε να συναντήσετε στο VagBot;

38 απαντήσεις



Γράφημα 13: Δυνατότητες VagBot

Συμπερασματικά, παρόλο που οι ερωτηθέντες είναι ευχαριστημένοι από το VagBot, υπάρχουν πάντα περιθώρια εξέλιξης. Αναφορικά, πολλοί χρήστες θα επιθυμούσαν το VagBot να συμπεριλαμβάνει περισσότερες πληροφορίες σχετικά με τα μαθήματα της σχολής.

Πίνακας 7.1 Απαντήσεις

Προεπιλεγμένες απαντήσεις	
Επιλογή αγγλικών ή άλλων γλωσσών	13
Επιλογή φωνητικής συνομιλίας	10
Περισσότερες πληροφορίες σχετικά με τα μαθήματα.	20
Προσωποποιημένες πληροφορίες μετά από σύνδεση με το app.	18
Απαντήσεις χρηστών	
Το παράδειγμα που έχει το Skroutz όταν μπαίνεις σε κάποια συγκεκριμένη αναζήτηση (πχ λάπτοπ), έχει πιο personalized προτάσεις και η χρήση βίντεο το κάνει πιο φιλικό. Αντίστοιχα κάτι τέτοιο θα ήταν πολύ χρήσιμο να μπει στην σελίδα των πρωτοετών που θα τους βοηθάει να κατατοπιστούν και να καταλάβουν τι δουλειά της κάθε πλατφόρμας που χρησιμοποιούμε στη σχολή. :)	
Mobile friendly design and more correct and details answers.	
Να δίνεται η δυνατότητα να υπάρχει σύνδεση με apps ώστε π.χ. για πρόγραμμα εξετάσεων να βγάζει αυτά που έχει δηλώσει	
Να απαντάει στις ερωτήσεις τύπου πόσα χρόνια είναι η σχολή	
Θα ήταν χρήσιμο αν δεν μπορεί να απαντηθεί μια ερώτηση μετά από την 1η η την 2η φορά και πέφτει σε plu fallback να παραπέμπει στο τηλέφωνο ή στο e-mail της Γραμματείας.	
Πιο "ευαίσθητο" σχετικά τις λέξεις κλειδιά του χρήστη στην ανάλογη ενότητα που αναζητά. Κατά τα άλλα είναι πολύ χρήσιμο και καλοφτιαγμένο. Μπράβο μάγκα!	

## Κεφάλαιο 7ο: Συμπεράσματα και προτάσεις βελτίωσης

Συνοψίζοντας, η δημιουργία μιας ευφυούς συνομιλίας είναι μια πρόκληση, καθώς υπάρχουν πολλές πτυχές που πρέπει να λάβει κάποιος υπόψη. Το chatbot είναι χρήσιμο και εύκολο στην αλληλεπίδραση του με τους χρήστες, καθώς οι τελευταίοι πληκτρολογούν το ερώτημα τους στη φυσική γλώσσα και έχουν τη δυνατότητα να ανακτήσουν πληροφορίες. Στη συγκεκριμένη εργασία, έγινε παρουσίαση πληροφοριών σχετικών με το σχεδιασμό και την υλοποίηση των εργαλείων αυτών, αλλά ταυτόχρονα έγινε εκτενής ανάλυση στη διαδικασία ανάπτυξης ενός συγκεκριμένου λογισμικού ευφυούς συνομιλίας, το VagBot. Το εργαλείο αυτό απευθύνεται σε φοιτητές του τμήματος μηχανικών πληροφορικής και ηλεκτρονικών υπολογιστών. Μέσα από την διεξαγωγή του ερωτηματολογίου, όπου συμμετείχαν 40 άτομα, γίνεται αντιληπτή η ανταπόκριση των συμμετεχόντων καθώς και η προθυμία τους να χρησιμοποιήσουν νέες τεχνολογίες, προσιτές, ευχάριστες και αποδοτικές. Λαμβάνοντας υπόψη τις απαντήσεις που δόθηκαν γίνεται αντιληπτό ότι πρόγραμμα έχει περιθώρια εξέλιξης. Αρχικά, θα μπορούσε να αυξηθεί το πλήθος των θεμάτων που καλύπτονται, καθώς και να εκπαιδευτεί καταλλήλως ώστε να το VagBot να κατανοεί τους λατινικούς χαρακτήρες. Το σημαντικότερο επίτευγμα είναι η ενσωμάτωση του εργαλείου στην κεντρική πλατφόρμα apps της σχολής των Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων. Ταυτόχρονα, μπορούν να απαντηθούν τα αρχικά ερωτήματα που αφορούν την αναγκαιότητα και τη χρησιμότητα ενός τέτοιου λογισμικού στα ακαδημαϊκά πλαίσια. Η ύπαρξη του VagBot, αλλά και παρόμοιων εργαλείων σε σχολές θεωρείται μια διαδικασία φιλική προς το χρήστη, η οποία εξοικονομεί χρόνο. Είναι εύκολο να ειπωθεί πως η ανάπτυξη και η βελτίωση του σχεδιασμού chatbot αυξάνεται με απρόβλεπτο ρυθμό λόγω της ποικιλίας των μεθόδων και προσεγγίσεων που χρησιμοποιούνται για το σχεδιασμό του. Επιπλέον, οι απαιτήσεις των χρηστών αλλάζουν και αυξάνονται συνεχώς και για αυτό το λόγο τα chatbot οφείλουν να εκπαιδεύονται ώστε να καλύψουν τα νέα δεδομένα. Καταλήγοντας, παρόλο που η ζήτηση για νέες τεχνολογίες διαρκώς αυξάνεται και η εξέλιξη των λογισμικών είναι συνεχόμενη, υπάρχουν περιθώρια βελτίωσης με σκοπό την εύρεση κατάλληλης προσέγγισης για το σχεδιασμό ενός chatbot.

## ΒΙΒΛΙΟΓΡΑΦΙΑ

### Βιβλία

- [1] R. Khan and D. Anik, *Introduction to chatbots in Build Better Chatbots*, Berkeley, CA:Apress, 2018.
- [2] S. Raj, *The Beloved Chatbots. In: Building Chatbots with Python*. Apress, Berkeley, CA.2019.

### Άρθρα Περιοδικών

- [3] T. Zemčík, A Brief History of Chatbots. DEStech Transactions on Computer Science and Engineering. 2019.
- [4] V. Sharma, et al. "An Intelligent Behaviour Shown by Chatbot System." *International Journal of New Technology and Research*, vol. 3, no. 4, Apr. 2017.
- [5] E. Adamopoulou and L. Moussiades, Chatbots: History, technology, and applications. *Machine Learning with Applications*.2020.
- [6] B. Shawar and E. Atwell, ALICE Chatbot: Trials and Outputs. *Computación y Sistemas*, 19(4), 625-632. 2015.
- [7] B. Shawar and E. Atwell, Different measurements metrics to evaluate a chatbot system. 89-96. 2007.
- [8] L. Bradeško and D. Mladeníc, A Survey of Chatbot Systems through a Loebner Prize Competition. *C*. 34. 2012.
- [9] J. Seering, M. Luria, G.Kaufman, and J. Hammer. Beyond Dyadic Interactions: Considering Chatbots as Community Members. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, Paper 450, 1–13. 2019
- [10]S. Sagiroglu and D. Sinanc, "Big data: A review," *International Conference on Collaboration Technologies and Systems (CTS)*, 2013, pp. 42-47. 2013
- [11] R. Sharma, An Analytical Study and Review of open source Chatbot framework, Rasa. *International Journal of Engineering Research and*. V9. 2020.
- [12] S. Perez-Soler, S. Juarez-Puerta, E. Guerra and J. de Lara, "Choosing a Chatbot Development Tool," in *IEEE Software*, vol. 38, no. 4, pp. 94-103. 2021
- [13] J, Anran, An Intelligent Chatbot System Based on Entity Extraction Using RASA NLU and Neural Network. *Journal of Physics: Conference Series*. 1487. 2020
- [14] C. Steinmacher, P. Ana and M. Gerosa, How should my chatbot interact? A survey on human-chatbot interaction design. 2019.

## **Ιστότοποι**

- [15] In-Depth Study Of RASA's DIET Architecture, Διαθέσιμο: [www.towardsdatascience.com](http://www.towardsdatascience.com)
- [16] Rasa NLU in Depth: Part 1 – Intent Classification, Διαθέσιμο: [www.rasa.com/blog/](http://www.rasa.com/blog/)
- [17] Rasa NLU in Depth: Part 2 – Entity Extraction, Διαθέσιμο: [www.rasa.com/blog/](http://www.rasa.com/blog/)
- [18] Improving entity extraction, Διαθέσιμο: [www.rasa.com/blog/](http://www.rasa.com/blog/)
- [19] Introducing DIET: outperforms BERT and is 6X faster to train, Διαθέσιμο: [www.rasa.com/blog/](http://www.rasa.com/blog/)
- [20] Introduction to Rasa X, Διαθέσιμο: [www.rasa.com/docs/rasa-x/](http://www.rasa.com/docs/rasa-x/)
- [21] MySQL workbench General Information, Διαθέσιμο: [www.dev.mysql.com](http://www.dev.mysql.com)
- [22] Introduction, Making requests, Testing APIs, Διαθέσιμο: [www.learning.postman.com](http://www.learning.postman.com)
- [23] Internet Growth Statistics, [www.internetworldstats.com/emarketing.htm](http://www.internetworldstats.com/emarketing.htm)
- [24] Rasa Documentation, Διαθέσιμο: [www.rasa.com/docs](http://www.rasa.com/docs)
- [25] 10 Key Metrics to Evaluate your AI Chatbot Performance, Διαθέσιμο: [www.inbenta.com](http://www.inbenta.com)