

ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

Τμήμα Ηλεκτρονικών Μηχανικών και Μηχανικών
Ηλεκτρονικών Υπολογιστών

ΠΟΛΥΜΟΡΦΙΚΑ ΣΥΣΤΗΜΑΤΑ ΕΠΙΚΟΙΝΩΝΙΑΣ ΓΙΑ
ΠΑΙΔΙΑ

Διπλωματική εργασία του

Αθανάσιου Τέγου

Επιτροπή καθηγητών

Αλέξανδρος Ποταμιάνος (Επιβλέπων)
Βασίλης Διγαλάκης
Ευριπίδης Πετράκης

Χανία 2005

Περιεχόμενα

1	Εισαγωγή	8
2	Ορίζοντας το Πολυμορφικό Σύστημα	9
2.1	Εισαγωγή	9
2.2	Ορίζοντας το Πολυμορφικό Σύστημα	10
2.3	Η Ιστορία των Πολυμορφικών Συστημάτων Επικοινωνίας	13
2.4	Τα Πλεονεκτήματα του Πολυμορφικού Τρόπου Επικοινωνίας	15
3	Αρχές Σχεδιασμού Πολυμορφικών Διαλογικών Συστημάτων	16
3.1	Εισαγωγή	16
3.2	Σχεδιάζοντας ένα Πολυμορφικό Σύστημα Διαλόγου	18
3.2.1	Η Αρχιτεκτονική ενός Πολυμορφικού Συστήματος	18
3.2.2	Συλλογή Δεδομένων και τα Πειράματα WoZ	19
3.2.3	Σχεδιασμός των Εφαρμογών και του Διαχειριστή Επικοινωνίας	21
3.2.4	Σχεδιασμός της Επιφάνειας Αλληλεπίδρασης	23
3.2.5	Η Ανταπόκριση του Χρήστη και η Αποτίμηση του Συστήματος	24

4	Υλοποιώντας ένα Πολυμορφικό Σύστημα Επικοινωνίας για Παιδιά	26
4.1	Εισαγωγή	26
4.2	Η Δομή του Συστήματος	26
4.3	Τρόπος Λειτουργίας του Συστήματος	30
4.4	Οι Εφαρμογές του Συστήματος	31
4.4.1	Η Λειτουργικότητα του Συστήματος	32
4.4.2	Τα Διαλογικά Χαρακτηριστικά του Συστήματος	35
4.5	Η Αρχιτεκτονική του Συστήματος	36
4.6	Σύστημα Αναγνώρισης Φωνής (audio client - ASR)	37
4.7	Οι Διαλογικές Καταστάσεις του Συστήματος	40
4.8	Dialog Manager	43
4.9	Αποτίμηση του Συστήματος	45
5	Τρόποι Βελτίωσης και Πιθανής Μελλοντικής Επέκτασης του Συστήματος	48
	Βελτίωση της Απόδοσης του Αναγνωριστή	48
	Μείωση του Χρόνου Απόκρισης	48
	Προσθήκη Επιπλέον Λειτουργικότητας	49
	Βελτίωση της Διαδραστικότητας μεταξύ Χρήστη και Συστήματος	49
	Παράρτημα	50
	Γραμματικές Χωρίς Συμφραζόμενα	50

Η Γραμματική του Dialog Manager σε μορφή Backus Naur	51
Το Εγχειρίδιο Χρήσης του Συστήματος	55
Το Σενάριο της Αποτίμησης του Συστήματος	59
Το Ερωτηματολόγιο της Αποτίμησης του Συστήματος	60
Τα Αποτελέσματα της Αποτίμησης του Συστήματος	62
 Βιβλιογραφία	 65

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέπων καθηγητή Αλέξανδρο Ποταμιάνο για την πολύτιμη βοήθεια και καθοδήγησή του καθ' όλη τη διάρκεια της εκπόνησης αυτής της διπλωματικής εργασίας.

Ακόμη, θα ήθελα να ευχαριστήσω την οικογένειά μου καθώς και όλους τους φίλους μου για την αμέριστη συνεισφορά τους.

Περίληψη

Τα πολυμορφικά συστήματα επικοινωνίας είναι συστήματα που συνδυάζουν αρκετές μορφές εισόδου και έχουν τη δυνατότητα να παρουσιάζουν την πληροφορία στο χρήστη συνδυάζοντας αρκετές διαφορετικές μορφές εξόδου.

Σκοπός της διπλωματικής αυτής εργασίας είναι η υλοποίηση μιας νέας έκδοσης του πρωτότυπου πολυμορφικού συστήματος επικοινωνίας για παιδιά CHIMP (Children's Interactive Multimedia Project). Από το πρωτότυπο σύστημα για το σκοπό αυτό χρησιμοποιήσαμε το γραφικό του περιβάλλον στο οποίο προσθέσαμε επιπλέον λειτουργικότητα καθώς και το σύστημα ελέγχου, το οποίο επίσης μεταποιήσαμε σε ένα βαθμό. Ενώ υλοποιήθηκαν από την αρχή το σύστημα διαλόγου καθώς και η φωνητική αλληλεπίδραση με το σύστημα.

Αρχικά ορίζουμε τις έννοιες της πολυμορφικότητας καθώς και αυτής των πολυμορφικών συστημάτων επικοινωνίας, παρουσιάζουμε την ιστορία των πολυμορφικών συστημάτων επικοινωνίας, όπως επίσης εκθέτουμε και τα πλεονεκτήματα του πολυμορφικού τρόπου επικοινωνίας.

Στη συνέχεια παρουσιάζουμε κάποιες βασικές αρχές σχεδιασμού των πολυμορφικών διαλογικών συστημάτων. Πιο συγκεκριμένα, γίνεται λόγος για την αρχιτεκτονική που συνήθως έχουν αυτά τα συστήματα καθώς και για τον τρόπο συλλογής δεδομένων μέσα από πειράματα WoZ. Αναφέρονται τρόποι σχεδιασμού των εφαρμογών του συστήματος, του διαχειριστή επικοινωνίας (Dialog Manager) και της επιφάνειας αλληλεπίδρασης, όπως επίσης και τεχνικές αποτίμησης του συστήματος.

Έπειτα, περιγράφεται αναλυτικά η υλοποίηση του συστήματος. Παρουσιάζεται η δομή και ο τρόπος λειτουργίας του. Οι εφαρμογές του συστήματος υλοποιήθηκαν με τη χρήση εξυπηρετητών (agents) και συγκεκριμένα έχει δύο εφαρμογές, έναν communication agent (ο οποίος κάνει ανάκτηση πληροφορίας από βάση δεδομένων, τηλεφωνεί, στέλνει e-mails, καταχωρεί δεδομένα στη βάση και δίνει πρόσβαση στο διαδίκτυο) και ένα παιχνίδι λέξεων (spelling bee). Η διαλογική υλοποίηση του πρώτου έγινε με τη χρήση 19 διαλογικών καταστάσεων ενώ του δεύτερου με τη χρήση μιας fsm. Εν' συνεχεία περιγράφεται η αρχιτεκτονική υλοποίησης του συστήματος αλλά και των επιμέρους υποσυστημάτων. Γίνεται εκτενής αναφορά του τρόπου υλοποίησης της φωνητικής αλληλεπίδρασης του συστήματος όπως επίσης και του τρόπου υλοποίησης του διαχειριστή επικοινωνίας (Dialog Manager). Ο οποίος κάνοντας εξόρυξη σημασιολογικής ερμηνείας (με τη χρήση γραμματικής) από την εντολή του χρήστη αποφασίζει για την ενέργεια που θα εκτελέσει το σύστημα και την απόκριση που θα δώσει στο χρήστη.

Τέλος, παρουσιάζονται τα αποτελέσματα και τα συμπεράσματα που προέκυψαν από την αποτίμηση του συστήματος όπως και τρόποι βελτίωσης και πιθανής μελλοντικής επέκτασης του συστήματος.

Κεφάλαιο 1

Εισαγωγή

Η ανάγκη του ανθρώπου για έναν πιο φυσικό και άμεσο τρόπο επικοινωνίας με ένα υπολογιστικό σύστημα, γέννησε τα πολυμορφικά συστήματα επικοινωνίας. Τα οποία, είναι συστήματα που συνδυάζουν αρκετές μορφές εισόδου και έχουν τη δυνατότητα να παρουσιάζουν την πληροφορία στο χρήστη συνδυάζοντας αρκετές διαφορετικές μορφές εξόδου.

Σκοπός της διπλωματικής αυτής είναι η υλοποίηση μιας νέας έκδοση του πρωτότυπου πολυμορφικού συστήματος επικοινωνίας για παιδιά CHIMP (Children's Interactive Multimedia Project). Από το πρωτότυπο σύστημα για το σκοπό αυτό χρησιμοποιήσαμε το γραφικό του περιβάλλον στο οποίο προσθέσαμε επιπλέον λειτουργικότητα καθώς και το σύστημα ελέγχου, το οποίο επίσης μεταποιήσαμε σε ένα βαθμό. Ενώ υλοποιήθηκαν από την αρχή το σύστημα διαλόγου καθώς και η φωνητική αλληλεπίδραση με το σύστημα.

Στο Κεφάλαιο 2, θα ορίσουμε τις έννοιες της πολυμορφικότητας καθώς και αυτής των πολυμορφικών συστημάτων επικοινωνίας, όπως επίσης θα παρουσιάσουμε και τα πλεονεκτήματα του πολυμορφικού τρόπου επικοινωνίας.

Στο κεφάλαιο 3, θα αναφερθούμε εκτενώς στις βασικές αρχές σχεδιασμού των πολυμορφικών συστημάτων επικοινωνίας με έμφαση στη φωνή σαν την κύρια μορφή επικοινωνίας.

Στο Κεφάλαιο 4, παρουσιάζεται η ανάπτυξη του συστήματος, περιγράφοντας αναλυτικά τα διάφορα στάδια υλοποίησής του καθώς και τα αποτελεσμάτων μιας αποτίμησης που κάναμε στο σύστημα που υλοποιήσαμε.

Τέλος, στο Κεφάλαιο 5, παρουσιάζονται ιδέες βελτίωσης και πιθανής μελλοντικής επέκτασης του συστήματος.

Κεφάλαιο 2

Ορίζοντας το Πολυμορφικό Σύστημα

2.1 Εισαγωγή

Στις αρχές της δεκαετίας του 80 εμφανίστηκε το γραφικό περιβάλλον επικοινωνίας (Graphical user interface), το οποίο, εκτός του ότι ήταν ένα μεγάλο βήμα προς τη δημιουργία εύχρηστων υπολογιστικών συστημάτων, αποτέλεσε και την απαρχή για την ευρεία εξάπλωση των υπολογιστών. Το γραφικό περιβάλλον επικοινωνίας παρείχε στο χρήστη οπτική αναπαράσταση των δεδομένων καθώς του έδινε και τη δυνατότητα για άμεσο έλεγχο του υπολογιστή χρησιμοποιώντας μαζί με το προγενέστερο πληκτρολόγιο και μια καινούρια συσκευή που λεγόταν ποντίκι. Η εξάπλωση των προσωπικών υπολογιστών αλλά και η καθολική αποδοχή από τους χρήστες αυτού του μέσου αλληλεπίδρασης, ανάγκασε τις εταιρίες να στραφούν προς αυτή την κατεύθυνση και να το εξελίσσουν συνεχώς (Apple MacOS, Microsoft Window X Window System).

Τη φράση “direct manipulation” την εισήγαγε ο Ben Shneiderman [1]. Σύμφωνα με τον ορισμό που έδωσε ένα direct manipulation interface θα πρέπει:

- να παρουσιάζει τα αντικείμενα οπτικά σε μια κατανοητή μορφή μεταφοράς
- να έχει γρήγορα ολοκληρώσιμες και ακυρώσιμες εντολές
- να παρουσιάζει τα αποτελέσματα μιας ενέργειας άμεσα
- να αντικαθιστά τη γραφή με το δείξιμο και την επιλογή

Το γραφικό περιβάλλον επικοινωνίας είναι βασισμένο στις αρχές του Shneiderman και πολλοί θεωρούν το mouse-based direct manipulation σαν τον ιδανικότερο τρόπο αλληλεπίδρασης ανθρώπου με υπολογιστή. Φυσικά όμως υπάρχουν και άλλες απόψεις πάνω σ’ αυτό το ζήτημα, μεταξύ αυτών, η Pattie Maes έχει επισημάνει μερικές ελλείψεις της direct manipulation αλληλεπίδρασης. Αυτή πιστεύει, λόγω της συνεχιζόμενης εξάπλωσης των υπολογιστών σε όλους τους τομείς άρα και την ολοένα και μεγαλύτερη χρήση τους από τους ανθρώπους, ότι οι υπολογιστικές μηχανές πρέπει να είναι

ενεργές και όχι παθητικά να περιμένουν από το χρήστη να θέσει εργασίες χειροκίνητα, χρησιμοποιώντας το direct manipulation. Η λύση που προτείνει η Maes [2], είναι η χρήση εφαρμογών πράκτορα (agents) στους υπολογιστές, οι οποίοι θα παρατηρούν το χρήστη και θα εκτελούν εργασίες ρουτίνας, ανάλογες με τις ανάγκες του εκάστοτε χρήστη.

Το υλικό των υπολογιστών (computer hardware) ήταν πάντα ένας πολύ σημαντικός παράγοντας που επηρέαζε την ανάπτυξη των εφαρμογών. Η επιφάνια αλληλεπίδρασης (user interface) δεν αποτελούσε εξαίρεση σ' αυτόν τον κανόνα. Το γραφικό περιβάλλον επικοινωνίας έγινε υλοποιήσιμο όταν το υλικό των υπολογιστών (computer hardware) είχε τη δυνατότητα να παράγει ακριβείς εικόνες bitmap και είχε αρκετή επεξεργαστική ισχύ ούτως ώστε διαδραστικά, να διαχειρίζεται με ακρίβεια τις αναπαραστάσεις της οθόνης. Μπορούμε να πούμε ότι ο γραφικός τρόπος επικοινωνίας του χρήστη με το σύστημα συντέλεσε μια επανάσταση στην επιστήμη των υπολογιστών.

Το επόμενο βήμα στην επικοινωνία ανθρώπου-υπολογιστή είναι το πολυμορφικό περιβάλλον επικοινωνίας (Multimodal user interface). Το πολυμορφικό περιβάλλον επικοινωνίας συνδυάζει αρκετές μορφές εισόδου και έχει τη δυνατότητα να παρουσιάσει την πληροφορία στο χρήστη συνδυάζοντας αρκετές διαφορετικές μορφές εξόδου. Οι μορφές εισόδου μπορεί να είναι απλές, όπως δύο συσκευές επιλογής ή πιο προχωρημένης τεχνολογικής αντίληψης, όπως αναγνωριστές φωνής, μηχανές όρασης, συσκευές ανίχνευσης ηλεκτρικών παλμών ή χειρονομιών. Οι απλές μορφές εισόδου σε ένα πολυμορφικό περιβάλλον δεν απαιτούν περισσότερη επεξεργαστική ισχύ από το συμβατικό γραφικό περιβάλλον επικοινωνίας, επιπρόσθετα όμως, δίνουν τη δυνατότητα στο χρήστη να έχει περισσότερους βαθμούς ελευθερίας έχοντας δύο ή περισσότερους τρόπους να τροφοδοτήσει με είσοδο το σύστημα.

2.2 Ορίζοντας το Πολυμορφικό Σύστημα

Στην επιστήμη υπολογιστών η πολυμορφική επιφάνια επικοινωνίας έχει οριστεί με πολλούς τρόπους. Ο Chatty [4], δίνει τον ορισμό σύμφωνα με τον οποίο με τον όρο πολυμορφικό σύστημα επικοινωνίας χαρακτηρίζουμε ένα σύστημα το οποίο το δέχεται διάφορες μορφές εισόδους (από τις οποίες δεν εξάγει πληροφορία) και παράγει απόκριση με διάφορες μορφές εξόδου. (Συστήματα Πολυμέσων, Multimedia Systems)

Ο Laurence Nigay και ο Joelle Coutaz [5], όρισαν την πολυμορφικότητα ως εξής:

“Πολυμορφικότητα είναι η ικανότητα του συστήματος να επικοινωνεί με το χρήστη με διαφορετικές μορφές επικοινωνίας, εξάγοντας και μεταφέροντας ερμηνεία αυτόματα.”

Και τα συστήματα πολυμέσων αλλά και τα πολυμορφικά συστήματα χρησιμοποιούν πολλαπλά κανάλια επικοινωνίας. Ο Nigay και ο Coutaz έκαναν το διαχωρισμό μεταξύ αυτών των συστημάτων υπογραμμίζοντας ότι ένα πολυμορφικό σύστημα επικοινωνίας είναι ικανό να μοντελοποιήσει το περιεχόμενο της πληροφορίας σε ένα υψηλό επίπεδο αφαιρετικότητας. Ένα πολυμορφικό σύστημα προσπαθεί να εξαγάγει νόημα. Για παράδειγμα, ένα ηλεκτρονικό σύστημα ταχυδρομείου το οποίο υποστηρίζει ήχο και βίντεο δεν είναι πολυμορφικό σύστημα εάν μόνο μεταφέρει τα δεδομένα σε ένα άλλο άτομο χωρίς να τα ερμηνεύει.

Σύμφωνα με τους Nigay και Coutaz τα δύο κύρια χαρακτηριστικά των πολυμορφικών συστημάτων είναι:

- η συνένωση διαφορετικών μορφών δεδομένων από/πρός διαφορετικές συσκευές εισόδου και εξόδου
- οι χρονικοί περιορισμοί που επιβάλλονται στην επεξεργασία της πληροφορίας από/πρός τις συσκευές εισόδου και εξόδου

Χρησιμοποίησαν αυτά τα χαρακτηριστικά για να ορίσουν έναν πίνακα αναφοράς για τα πολυμορφικά συστήματα, ο οποίος παρουσιάζεται στο Γράφημα 1.

		USE OF MODALITIES	
		Sequential	Parallel
FUSION	Combined	ALTERNATE	SYNERGISTIC
	Independent	EXCLUSIVE	CONCURRENT
		Meaning No Meaning	Meaning No Meaning
		LEVELS OF ABSTRACTION	

Γράφημα 1: Πίνακας αναφοράς για τα πολυμορφικά συστήματα.

Οι διαφορετικές διαστάσεις του πίνακα αναφοράς ορίζουν οχτώ πιθανούς τύπους συστήματος. Εξ' ορισμού, τα πολυμορφικά συστήματα χρειάζονται την τιμή "Meaning" στα "Levels of abstraction". Έτσι, υπάρχουν τέσσερις διαφορετικοί τύποι πολυμορφικών συστημάτων : τα αποκλειστικά, τα εναλλασσόμενα, τα παράλληλα και τα συνεργαζόμενα.

Η διάσταση "Use of modality" αναφέρεται στο χρόνο. Η παράλληλη λειτουργία μπορεί να επιτευχθεί σε διαφορετικά αφαιρετικά επίπεδα [5]. Το πιο σημαντικό επίπεδο είναι το επίπεδο εργασίας, το επίπεδο δηλαδή με το οποίο ο χρήστης αλληλεπιδρά. Αυτό πρέπει να φαίνεται ότι δρα παράλληλα για να κάνει το χρήστη να παρατηρεί το σύστημα σαν παράλληλο. Τα περισσότερα σύγχρονα συστήματα έχουν τη δυνατότητα να προσφέρουν μια

μίμηση της παράλληλης λειτουργίας, έστω και αν η λειτουργία στην πραγματικότητα είναι ακολουθιακή.

Η ένωση (Fusion) είναι το πιο απαιτητικό κριτήριο στον πίνακα αναφοράς. Εδώ το τμήμα “Combined” υποδηλώνει ότι διαφορετικές μορφές επικοινωνίας συνδυάζονται σε υψηλού επιπέδου συνεργαζόμενα tokens εισόδου. Ενώ το τμήμα “Independent” υποδηλώνει ότι οι μορφές επικοινωνίας είναι παράλληλες αλλά όχι ενωμένες στην επιφάνεια αλληλεπίδρασης (interface). Η συνεργαζόμενη τύπου χρήση των μορφών επικοινωνίας σημαίνει την ένωση των δεδομένων από διαφορετικές τεχνικές μοντελοποίησης. Οι Nigay και Coutaz [5], έχουν επίσης προσδιορίσει τρία επίπεδα ένωσης: το λεκτικό, το συντακτικό και το σημασιολογικό. Αυτά μπορούμε να τα αντιστοιχίσουμε στα τρία θεμελιώδη σχεδιαστικά επίπεδα που έχει ορίσει ο Foley [6].

- Η *λεκτική ένωση* αντιστοιχεί στο Binding level. Αυτό συμβαίνει όταν στοιχεία του υλικού (hardware) συνδέονται σε γεγονότα εφαρμογών. Σαν παράδειγμα λεκτικής ένωσης είναι η επιλογή πολλαπλών αντικειμένων όταν το πλήκτρο shift είναι πατημένο.
- Η *συντακτική ένωση* αντιστοιχεί στο Sequencing level. Αυτό επίπεδο περιλαμβάνει το συνδυασμό δεδομένων ώστε να αποκομίσει το σύστημα μια ολοκληρωμένη εντολή. Η αλληλουχία γεγονότων είναι σημαντική σε αυτό το επίπεδο. Ένα παράδειγμα συντακτικής ένωσης είναι ο συγχρονισμός λόγου και εισόδου μέσω συσκευής στυλό (pen input) σε μια εργασία επιλογής σε ένα χάρτη.
- Η *σημασιολογική ένωση* αντιστοιχεί στο Functional level. Αυτό καθορίζει τη λεπτομερή λειτουργικότητα της επιφάνειας αλληλεπίδρασης (interface) σε ένα σύστημα: τι πληροφορία χρειάζεται για κάθε λειτουργία, πώς να διαχειριστεί τα λάθη και πια είναι τα αποτελέσματα κάθε λειτουργίας. Η σημασιολογική ένωση ορίζει την έννοια αλλά όχι και την ακολουθία των ενεργειών ή τις συσκευές με τις οποίες οι ενέργειες εκτελούνται. Ένα παράδειγμα σημασιολογικής ένωσης είναι η επιλογή μιας πτήσης η οποία χρειάζεται τουλάχιστον δύο αεροδρόμια και η είσοδος στο σύστημα μπορεί να γίνει είτε δείχνοντας, είτε με φωνή και επιλογή ζωγραφίζοντας το δρομολόγιο στο χάρτη.

Ακόμη, αν και γνωρίζουμε ότι τα πολυμορφικά συστήματα επικοινωνίας χρησιμοποιούν πολλαπλές μορφές εισόδου και/ή εξόδου δεν έχουν καθοριστεί με σαφήνεια οι ιδιότητες τους. Υπάρχουν λοιπόν δύο κύριες κατηγορίες συστημάτων:

1. Συστήματα στα οποία οι διαφορετικές μορφές εισόδου χρησιμοποιούνται για εμπλουτισμό του direct manipulation του συστήματος. Το σύστημα είναι ένα παθητικό εργαλείο και προσπαθεί να κατανοήσει την εντολή του χρήστη μέσα από τις διαφορετικές μορφές εισόδου που το σύστημα αναγνωρίζει. Ο χρήστης είναι πάντα υπεύθυνος

για την αρχικοποίηση των λειτουργιών (initializing the operations). Έτσι, εάν ο χρήστης δεν ξέρει τι να κάνει, τίποτα δεν εκτελείται. Η λογική αυτών των συστημάτων είναι η χρήση τους σαν εργαλείο, αυτή η προσέγγιση ακολουθεί τις αρχές του direct manipulation interface, Ben Shneiderman [1].

2. Συστήματα στα οποία οι πολλαπλές μορφές επικοινωνίας χρησιμοποιούνται για να αυξηθεί ο ανθρωπομορφισμός στην επιφάνεια αλληλεπίδρασης : *το σύστημα δρα σαν ένας διαλογικός συνεργάτης*. Λόγο αυτής της προσέγγισης τα συστήματα αυτής της κατηγορίας δίνουν μεγάλη έμφαση στην πολυμορφική απόκριση του συστήματος και συνήθως χρησιμοποιούν σύνθεση και αναγνώριση φωνής, καθώς και άλλες ανθρώπινες μορφές επικοινωνίας. Αυτού του είδους τα πολυμορφικά συστήματα περιγράφονται συνήθως σαν διαλογικά συστήματα επικοινωνίας με πράκτορα (agent).

Η επιλογή μεταξύ αυτών των δύο ή και άλλων τύπων πολυμορφικών συστημάτων πρέπει να γίνεται από την αρχή, στο στάδιο προσδιορισμού των απαιτήσεων και προδιαγραφών του συστήματος που πρόκειται να υλοποιήσουμε.

2.3 Η Ιστορία των Πολυμορφικών Συστημάτων Επικοινωνίας

Αρκετοί τύποι πρωτότυπων πολυμορφικών συστημάτων έχουν υλοποιηθεί τονίζοντας την χρησιμότητα και χρηστικότητα της πολυμορφικής επικοινωνίας. Σ' αυτή τη παράγραφο θα σας παρουσιάσω μερικά από αυτά τα συστήματα.

Είδη από τα μέσα του περασμένου αιώνα συστήματα εικονικής πραγματικότητας άρχισαν να μελετούνται από τον Morton Heilig's Sensorama [7]. Αυτά τα συστήματα είναι παραπλήσια των πολυμορφικών μιας και χρησιμοποιούν αρκετές παράλληλες μορφές επικοινωνίας για είσοδο (π.χ γάντια, φωνή) αλλά και για έξοδο. Παρόλα αυτά, τα συστήματα εικονικής πραγματικότητας είναι αρκετά διαφορετικά από τα πολυμορφικά. Η πραγματική διαφορά μεταξύ αυτών των συστημάτων έχει λεπτομερώς εξηγηθεί στο [8]: "Η εικονική πραγματικότητα στοχεύει στο μιμητισμό της πραγματικότητας, εγκαθιστώντας ψευδαισθήσεις με ήχο και εικόνα, ενώ η πολυμορφικότητα προσπαθεί να εμπλουτίσει την έξοδο ενός συστήματος αλλά και το νατουραλισμό της αλληλεπίδρασης ανθρώπου-συστήματος."

Ο Bolt [9], εισήγαγε την ιδέα της πολυμορφικής επικοινωνίας 'με το σύστημά του, Put-That-There. Σε αυτό το σύστημα ο χρήστης μπορούσε να μετακινεί αντικείμενα στην οθόνη, επιλέγοντας και μιλώντας. Ο χρήστης επέλεγε ένα αντικείμενο, λέγοντας, "put that", επέλεγε τον προορισμό και έλεγε "there". Αυτό το σύστημα συνδύαζε επιλογή και ομιλία σε μια απλή

εφαρμογή. Έπειτα, πιο περίπλοκα πρωτότυπα υλοποιήθηκαν προσθέτοντας μηχανική όραση σε ένα παραπλήσιο σύστημα [10].

Ο Coehen [11], παρουσίασε δύο συστήματα, το CHORIS και το SHOPTALK τα οποία συνδύαζαν direct manipulation με φυσική γλώσσα. Στα συστήματα αυτά η χρήση της φυσικής γλώσσας γινόταν από το πληκτρολόγιο. Οι κατασκευαστές αυτών των συστημάτων παρατήρησαν ότι αυτές οι μορφές επικοινωνίας συνδυαζόμενες ξεπερνούσαν κάποιους περιορισμούς που είχε η κάθε μια μορφή δρώντας μεμονωμένα. Ο Wahlster [12], παρουσίασε ένα παρόμοιο σύστημα το XTRA.

Το CUBRION [13], είναι ένα σύστημα το οποίο χρησιμοποιεί ποντίκι μαζί με φωνή. Ο Cohen [14], υλοποίησε ένα πρωτότυπο που λεγόταν QuickSet το οποίο έκανε χρήση φωνής μαζί με εισόδους σχεδίασης γραφικών (pen device), συμβολισμών, χειρονομιών και επιλογή σημείων (pointing).

Ο Weimer και ο Ganapathy [15], περιέγραψαν ένα περιβάλλον το οποίο χρησιμοποιούσε χειρονομίες και φωνή ώστε να διευθύνουν ένα CAD σύστημα. Αυτοί έκαναν χρήση γαντιών δεδομένων για τις χειρονομίες και γυαλιών πολώσεως για να απεικονίσουν τα αντικείμενα σε τρις διαστάσεις.

Η Oniatt [16], παρουσίασε ένα πολυμορφικό σύστημα δυναμικών χαρτών που αλληλεπιδρούσαν με το χρήστη. Το σύστημα δεχόταν φωνή και σχέδια (pen device), είτε ξεχωριστά, είτε μαζί για την εκτέλεση εργασιών με χάρτες. Η Oniatt [17], παρατήρησε ότι η χρήση της φωνής αλλά και της σχεδίασης σαν μορφές επικοινωνίας στο σύστημα, παρείχαν συμπληρωματική σημασιολογική πληροφορία.

Ο Yang [18], παρουσίασε ένα σύνολο από τεχνικές οπτικής παρακολούθησης οι οποίες θα μπορούσαν να χρησιμοποιηθούν σε πολυμορφικά συστήματα. Υλοποιώντας έναν ανιχνευτή προσώπου ο οποίος μπορούσε επίσης να εστιάζει και σε συγκεκριμένα μέρη του προσώπου, όπως μάτια, χείλη και ρουθούνια. Το σύστημα ακόμη είχε την ικανότητα να εντοπίζει την διεύθυνση του βλέμματος ή την πόζα του κεφαλιού. Ο Yang χρησιμοποιώντας αυτές τις τεχνικές υλοποίησε ένα πολυμορφικό σύστημα που διάβαζε τα χείλη (lip-reading). Το σύστημα lip-reading βελτίωνε την απόδοση της αναγνώρισης φωνής χρησιμοποιώντας παράλληλα με τον αναγνωριστή φωνής και την εικόνα της κίνηση των χειλιών του χρήστη, ώστε να αποσαφηνιστούν κάποια ακουστικά στοιχεία της φωνής (audio-visual recognition).

2.4 Τα Πλεονεκτήματα του Πολυμορφικού Τρόπου Επικοινωνίας

Οι Mayburoi και Whlster [19], έδωσαν μια λίστα των πλεονεκτημάτων του πολυμορφικού τρόπου επικοινωνίας με ένα σύστημα: *αποδοτικότητα, ευκολία επικοινωνίας, φυσικότητα, ακρίβεια και συνεργασία*.

Αποδοτικότητα, καθώς ο πολυμορφικός τρόπος επικοινωνίας δίνει τη δυνατότητα να χρησιμοποιούμε κάθε φορά την κατάλληλη για κάθε εργασία, μορφή επικοινωνίας με το σύστημα.

Ευκολία επικοινωνίας, μιας και ο σκοπός της επικοινωνίας διαβιβάζεται στο σύστημα με ευκολία, λόγο των πολλών ταυτόχρονων αναφορών στο ίδιο θέμα.

Φυσικότητα, λόγω της δυνατότητας επιλογής του τρόπου συνδιαλλαγής με το σύστημα, προσεγγίζοντας έτσι την άνθρωπος-άνθρωπος επικοινωνία.

Ακρίβεια, διότι στην πολυμορφική επικοινωνία μια άλλη μορφή επικοινωνίας μπορεί να χαρακτηρίσει ένα αντικείμενο με μεγαλύτερη ακρίβεια από την κύρια μορφή.

Συνεργασία, καθώς ένα κανάλι επικοινωνίας μπορεί να βοηθήσει ένα άλλο κανάλι ώστε να βελτιωθούν οι ανακρίβειες, να τροποποιηθεί το νόημα ή να λυθούν αμφισημίες.

Κεφάλαιο 3

Αρχές Σχεδιασμού Πολυμορφικών Διαλογικών Συστημάτων

3.1 Εισαγωγή

Στο κεφάλαιο αυτό θα σας παρουσιάσω τις βασικές αρχές σχεδιασμού ενός πολυμορφικού συστήματος επικοινωνίας με έμφαση τη φωνή σαν την κύρια μορφή επικοινωνίας [20].

Για να σχεδιάσει κανείς ένα αποδοτικό και αποτελεσματικό διαλογικό σύστημα απαιτείται επιστημονική εξειδίκευση και πείρα πάνω στην επεξεργασία φωνής αλλά και φυσικού λόγου. Τα δομικά στοιχεία από τα οποία συνήθως αποτελείται ένα διαλογικό σύστημα περιλαμβάνουν:

Αναγνωριστή φωνής (automatic speech recognition-ASR)

Ο οποίος μετατρέπει τον προφορικό λόγο του χρήστη σε γραπτό χρησιμοποιώντας το φάσμα του λόγου του.

Συνθέτη φωνής (text-to-speech synthesis-TTS)

Ο οποίος μετατρέπει γραπτό λόγο σε προφορικό λόγο.

Σύστημα σημασιολογικού ελέγχου (natural language understanding-NLU)

Εδώ γίνεται σημασιολογική εξόρυξη πληροφορίας από τα δεδομένα εισόδου του χρήστη.

Διαχειριστή εφαρμογών (application manager)

Είναι υπεύθυνος για τη διαχείριση των εφαρμογών του συστήματος.

Διαχειριστή επικοινωνίας (dialog manager)

Επεξεργάζεται τα δεδομένα εισόδου του χρήστη και αποφασίζει για τις ενέργειες απόκρισης του συστήματος, συνήθως ο διαχειριστής εφαρμογών και

ο διαχειριστή επικοινωνίας συσσωματώνονται σε ένα κοινό δομικό στοιχείο ενός διαλογικού συστήματος.

Βάση δεδομένων (database)

Περιέχει πληροφορία ανάλογη με την εφαρμογή του συστήματος και την οποία προσπελάζει το σύστημα.

Διαχειριστή συγχρονισμού των δομικών στοιχείων του συστήματος (controller/event handler)

Διαχειρίζεται όλο τα δομικά μέρη του συστήματος και είναι υπεύθυνος για το συντονισμό τους.

Και επιπλέον για πολυμορφικά διαλογικά συστήματα έχουμε:

Γραφικό περιβάλλον επικοινωνίας (graphical user interface-GUI)

Είναι το μέσο επικοινωνίας μεταξύ του συστήματος και του χρήστη.

Αναγνωριστή χειρονομίας/νεύματος (gesture/sign recognition)

Υποσύστημα που δέχεται σαν είσοδο χειρονομίες/νεύματα, σαν μέσο επικοινωνίας με το σύστημα, και παράγει ανάλογη με τη φύση του συστήματος έξοδο ως προς αυτό.

Αναγνωριστή οπτικής ομιλίας (visual speech recognition)

Ο οποίος μετατρέπει τον προφορικό λόγο του χρήστη σε γραπτό χρησιμοποιώντας την εικόνα του προσώπου του χρήστη καθώς μιλάει στο σύστημα.

Η σχεδίαση ενός διαλογικού συστήματος συνήθως περιλαμβάνει τέσσερα βασικά βήματα : (i) την αρχιτεκτονική σχεδίαση του συστήματος, (ii) τη σχεδίαση των εφαρμογών του συστήματος και της συλλογή δεδομένων, (iii) τη σχεδίαση του περιβάλλοντος επικοινωνίας και (iv) τέλος την αποτίμηση του συστήματος. Αυτά τα βήματα δεν θα πρέπει να υλοποιούνται ανεξάρτητα το ένα από το άλλο, αλλά θα πρέπει να υπάρχει κάποιου είδους συντονισμός ώστε το σύστημα να εγγυάται ευστάθεια και ομαλή λειτουργία.

Η διαδικασία σχεδιασμού ενός διαλογικού συστήματος καθώς και του περιβάλλοντος αλληλεπίδρασης του με τον χρήστη είναι μια επαναλαμβανόμενη διαδικασία, όπως συμβαίνει συνήθως κατά το σχεδιασμό εφαρμογών που έχουν να κάνουν με πολλαπλές μορφές εισόδου/εξόδου. Ο κύριος λόγος αυτής της διαρκούς σχεδιαστικής επαναπροσέγγισης έγκειται στην έλλειψη συγκεκριμένων παραμέτρων οι οποίοι θα αποτελούσαν ένα αντικειμενικό μέτρο σύγκρισης για την αποτίμηση της απόδοσης του συστήματος.

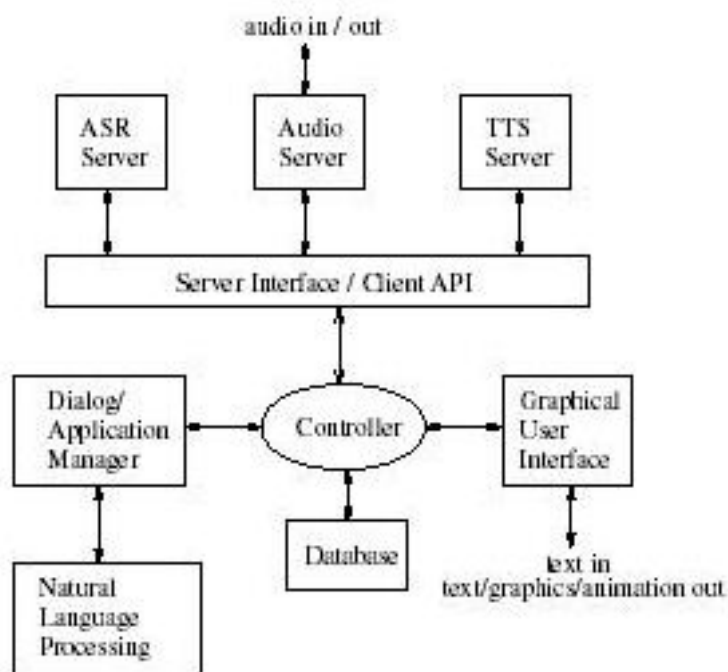
3.2 Σχεδιάζοντας Ένα Πολυμορφικό Σύστημα διαλόγου

Ένα πολύ βασικό θέμα στο σχεδιασμό ενός διαλογικού συστήματος αποτελεί η επιλογή των μορφών εισόδου και εξόδου που θα χρησιμοποιηθούν στο περιβάλλον επικοινωνίας(interface) του χρήστη με το σύστημα. Υπάρχουν αρκετοί τρόποι για να επιλέξει κανείς τον κατάλληλο συνδυασμό μορφών επικοινωνίας για ένα διαλογικό σύστημα, παρόλα αυτά είναι φανερό ότι η χρήση μόνο της φυσικής γλώσσας σαν μέσω επικοινωνίας δεν είναι πάντα και η καλύτερη επιλογή για όλες τις εφαρμογές. Σημειώστε επίσης ότι ένα διαλογικό σύστημα θα πρέπει να διέπεται από τις γενικές αρχές που διέπουν την επιστήμη των υπολογιστών, δηλαδή *γενικότητα, συμβατότητα και ενθυλάκωση*.

3.2.1 Η αρχιτεκτονική ενός πολυμορφικού συστήματος

Η αρχιτεκτονική των πολυμορφικών διαλογικών συστημάτων έχει με το πέρασμα του χρόνου εξελιχθεί από μονολιθική σε σπονδυλωτή αρχιτεκτονική με καλά ορισμένα πρωτόκολλα επικοινωνίας μεταξύ των στοιχείων της. Στο Γράφημα 1, παρουσιάζεται η γενική αρχιτεκτονική για ένα πολυμορφικό σύστημα επικοινωνίας με έμφαση στη φωνητική μορφή επικοινωνίας. Στο οποίο ο αναγνωριστής φωνής (ASR server), ο εξυπηρετητής ήχου (Audio server) και ο συνθέτης φωνής (TTS server) ελέγχονται από ένα καλά ορισμένο διαχειριστή (API application programming interface). Ο διαχειριστής είναι υπεύθυνος για τη σωστή λειτουργία του εξυπηρετητή ήχου (Audio server) και του αναγνωριστή (ASR) κατά τη διάρκεια εισόδου της φωνής στο σύστημα αλλά και του εξυπηρετητή ήχου και του συνθέτη φωνής (TTS) κατά τη διάρκεια εξόδου της φωνής από το σύστημα (απόκριση του συστήματος). Επίσης ο διαχειριστής συστήματος API μπορεί να διαχειρίζεται πολλαπλούς αναγνωριστές, συνθέτες ή εξυπηρετητές ήχου προκειμένου να έχουμε μια πολυμορφική επικοινωνία με το σύστημα. Στη συνέχεια το σύστημα αποτελείται από έναν ελεγκτή (Controller) ο οποίος «ενώνει» τα πολυμορφικά δεδομένα εισόδου (ήχο, κείμενο, χειρονομία) και συγχρονίζει τα πολυμορφικά δεδομένα απόκρισης του συστήματος (ήχο, κείμενο, εικόνα). Το γενικό διάγραμμα ροής ενός τέτοιου συστήματος έχει ως εξής: παίρνει την/τις μεταγλωττίσεις από τον/τους αναγνωριστές (ASR) και/ή γραφικό περιβάλλον επικοινωνίας (GUI), τις στέλνει στο διαχειριστή επικοινωνίας (Dialog manager)/ σύστημα σημασιολογικού ελέγχου (NLU) για επεξεργασία, παίρνει τα αποτελέσματα απ' το διαχειριστή επικοινωνίας (Dialog manager) και τα παρουσιάζει στο χρήστη. Πληροφορία απ' τη βάση δεδομένων μπορεί προαιρετικά να χρησιμοποιηθεί απ' τον ελεγκτή (Controller) μετά από απαίτηση του διαχειριστεί εφαρμογών (Application manager). Μια γλώσσα προγραμματισμού που ενδείκνυται για την κατασκευή του ελεγκτή (Controller) και του διαχειριστή επικοινωνίας (Dialog manager) είναι η PERL. Αποκλίσεις πάνω στην αρχιτεκτονική που σας παρουσίασα, αρχιτεκτονική

που στηρίζεται πάνω στο project DAPTA, υπάρχουν αρκετές στη βιβλιογραφία, π.χ. [21], όπου λειτουργικότητα απ'τον διαχειριστή εφαρμογών (Application manager) έχει μεταφερθεί στον ελεγκτή (Controller) για να μειωθεί η πολυπλοκότητα. Η βασική αρχιτεκτονική δομή παραμένει η ίδια.



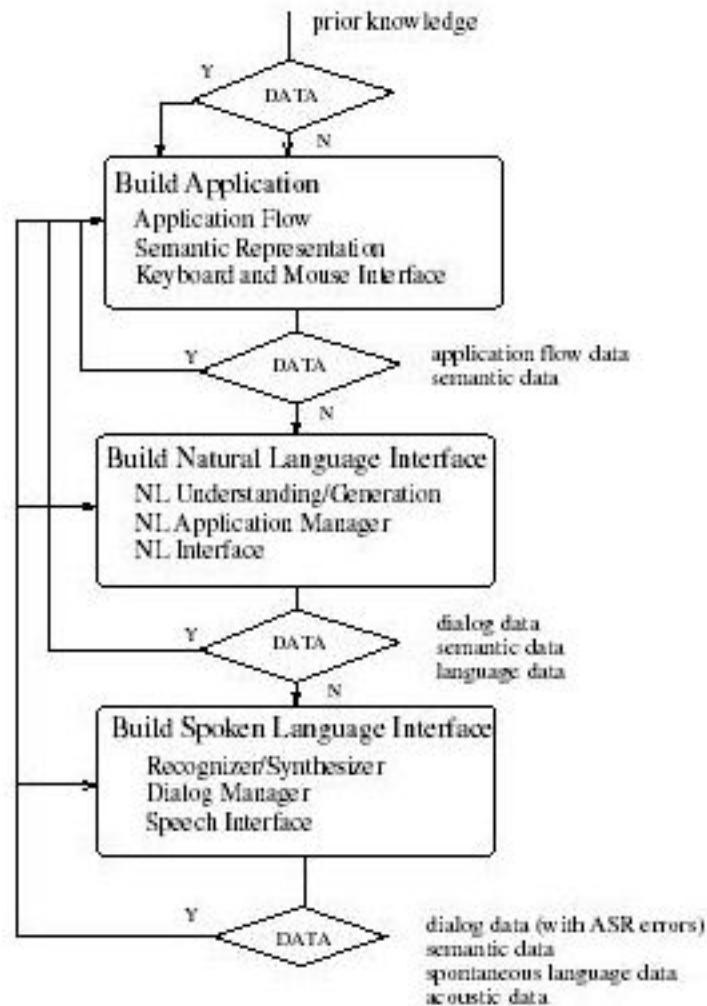
Γράφημα .1. Η Αρχιτεκτονική των Πολυμορφικών Συστημάτων

Επειδή δεν υπάρχει ομοφωνία μεταξύ των ερευνητών για το πρωτόκολλο επικοινωνίας μεταξύ των διαφόρων υποσυστημάτων του συστήματος, χρησιμοποιούνται και σύγχρονοι και ασύγχρονοι (γεγονός - ενέργεια) τρόποι επικοινωνίας μεταξύ των υποσυστημάτων. Παρόλα αυτά οι σύγχρονοι προτιμούνται όταν αυτό είναι εθικτό. Σε μερικά συστήματα η πληροφορία διαβιβάζεται μεταξύ των υποσυστημάτων με τη μορφή μηνυμάτων. Τα μηνύματα περιέχουν όλη την απαραίτητη για τον εκάστοτε εξυπηρετητή πληροφορία και έτσι οι εξυπηρετητές καθίστανται stateless. Παρόλα αυτά η μορφή και τα περιεχόμενα αυτών των μηνυμάτων διαφέρουν από σύστημα σε σύστημα. Τέλος οι διαχειριστές του συστήματος (API's) για την επικοινωνία μεταξύ του διαχειριστή επικοινωνίας (Dialog manager), του ελεγκτή (Controller), του συστήματος σημασιολογικού ελέγχου (NLU) και της βάσης δεδομένων πρέπει να οριστούν προσεκτικά.

3.2.2 Συλλογή δεδομένων και τα πειράματα WoZ

Η συλλογή δεδομένων και η ανάλυσή τους αποτελεί ένα πολύ σημαντικό μέρος στην ανάπτυξη ενός πολυμορφικού συστήματος επικοινωνίας. Το

Γράφημα 2, παρουσιάζει τα διάφορα στάδια της συλλογής δεδομένων μιας εφαρμογής, καθώς ο τρόπος επικοινωνίας του συστήματος με το χρήστη εκτυλίσσεται από το πληκτρολόγιο και το ποντίκι έως την ομιλία. Τα δεδομένα μπορούν να συλλεχθούν σε διάφορα στάδια, είτε αυτόματα, είτε με τη βοήθεια κάποιου επιβλέποντα (Wizard). Τα είδη των δεδομένων που συλλέγονται είναι ακουστικά, λεξιλογικά, σημασιολογικά και δεδομένα ροής διαλόγου όπως φαίνεται στο γράφημα. Το επίπεδο κατανόησης των λέξεων καθώς και η σημασιολογική τους ερμηνεία ανάλογα με την εφαρμογή γίνεται στα δεδομένα είτε χειροκίνητα είτε ημιαυτόματα (χρησιμοποιώντας την έξοδο των υποσυστημάτων αναγνώρισης και κατανόησης). Τα δεδομένα που συλλέγονται είναι συναφή με το σκοπό για τον οποίο υλοποιήθηκε το σύστημα και το σενάριο της εκάστοτε εφαρμογής του. Σαν αποτέλεσμα, η ανάπτυξη διαλογικών συστημάτων να είναι μια επαναληπτική διαδικασία, δηλαδή τα δεδομένα που συλλέχθηκαν χρησιμοποιούνται για να βελτιωθεί το σύστημα και επιπλέον νέα δεδομένα συλλέγονται χρησιμοποιώντας το βελτιωμένο σύστημα.



Γράφημα.2.Επαναληπτική Διαδικασία Σχεδιασμού Διαλογικών Συστημάτων, χρησιμοποιώντας Συλλογή Δεδομένων σε διάφορα Στάδια.

Το Wizard of Oz (WoZ) πείραμα είναι ένας τρόπος συλλογής δεδομένων και χρησιμοποιείται εκτενώς για το σχεδιασμό του μέσου αλληλεπίδρασης ενός συστήματος με το χρήστη. Τα WoZ πειράματα μπορούν να διαχωριστούν στα i) fully wizarded systems, ii) wizard-supervised systems. Τα fully wizarded διαλογικά συστήματα χρησιμοποιούν δύο «υπεύθυνους» (wizards) ο ένας καταγράφει την είσοδο του χρήστη και ο άλλος γράφει την έξοδο του συστήματος. Κάτι που έχει σαν αποτέλεσμα ο χρόνος απόκρισης του συστήματος να είναι μεγάλος και τα δεδομένα που λαμβάνουμε να είναι μη ρεαλιστικά λόγω αυτού του γεγονότος. Συνήθως, κατά την εκτέλεση WoZ πειραμάτων χρησιμοποιούνται εντελώς αυτόματα συστήματα για τη διεξαγωγή του πειράματος και ο υπεύθυνος (wizard) ενεργεί σαν επιβλέπων της όλης διαδικασίας, διορθώνοντας λάθη και παρεμβαίνοντας όταν τα πράγματα δεν πάνε καλά, πχ [22]. Τα WoZ πειράματα είναι ιδιαίτερος χρήσιμα για άμεσα συστήματα εισόδου-εξόδου ώστε να εξοικειωθούν με τη φωνητική μορφή επικοινωνίας.

3.2.3 Σχεδιασμός των Εφαρμογών και του Διαχειριστή Επικοινωνίας

Ο σχεδιασμός μιας καλής εφαρμογής είναι το πρώτο βήμα προκειμένου να χτίσει κανείς ένα επιτυχημένο διαλογικό σύστημα επικοινωνίας. Η σημασία που έχει η σχεδίαση μιας καλής εφαρμογής συνήθως παραμερίζεται απ' τους ενασχολούμενους με την επιστήμη της φωνής, των οποίων η έμφαση εστιάζεται στη δημιουργία ενός μέσου επικοινωνίας του συστήματος με το χρήστη με αποκλειστική χρήση της φωνής. Κάτι που έχει σαν συνέπεια, οι εφαρμογές των διαλογικών συστημάτων να εστιάζονται και να σχεδιάζονται έχοντας ως επίκεντρο το μέσο επικοινωνίας (interface) καθώς και περιορισμένο εύρος λειτουργιών και δυνατοτήτων. Στο Γράφημα 2, παρουσιάζεται ο τρόπος σχεδιασμού ενός διαλογικού συστήματος προσεγγίζοντας το με έμφαση στην εφαρμογή του συστήματος, η οποία εξελίσσεται καθώς το μέσο επικοινωνίας μεταξύ ανθρώπου και συστήματος εμπλουτίζεται με δυνατότητες χρήσης φυσικής γλώσσας αλλά και προφορικού λόγου.

Τα κύρια πλεονεκτήματα αυτής της σχεδιαστικής προσέγγισης είναι: αύξηση του καταμερισμού του συστήματος (υπάρχει δηλαδή διαχωρισμός μεταξύ μέσου επικοινωνίας του συστήματος και της εφαρμογής), δίνεται η δυνατότητα συλλογής δεδομένων σε πολλαπλά στάδια μιας και η εφαρμογή και το μέσο επικοινωνίας συνελίσσονται, τέλος το μέσο επικοινωνίας (interface) μπορεί εύκολα να προσαρμοστεί για «γενική πρόσβαση» (universal access), πχ να δημιουργεί ο κατάλληλος συνδυασμός μορφών επικοινωνίας βασιζόμενοι στον τρόπο με τον οποίο η εφαρμογή λειτουργεί (τηλεφωνικές εφαρμογές, ψηφιακός βοηθός ή PC).

Μια πολύ σημαντική εφαρμογή των διαλογικών συστημάτων είναι ο προσωπικός βοηθός/πράκτορας (personal assistant/agent), ο οποίος εμπεριέχει αρκετές υπό-εφαρμογές όπως, τηλεφωνικές εφαρμογές (ονομαστικός κατάλογος – τηλεφωνήματα), εφαρμογές μηνυμάτων (μηνύματα φωνής, e-mail, fax), εφαρμογές ανάκτησης πληροφορίας (ειδήσεις, καιρός κ.α.), οικονομικές εφαρμογές (εμπορικές συναλλαγές, τράπεζες κ.α), εφαρμογές ταξιδιωτικού πράκτορα (αεροπορικές κρατήσεις, ξενοδοχεία κ.α). Στις [23,24], μια ιεραρχική από πάνω προς τα κάτω προσέγγιση προτείνεται κατά το σχεδιασμό της εφαρμογής του συστήματος. Οι εφαρμογές τις οποίες ο πράκτορας υποστηρίζει είναι οργανωμένες σε μια δομή ιεραρχικού δέντρου (hierarchical tree-structure) στις οποίες οι χρήστες κατευθύνονται χρησιμοποιώντας τη φωνή. Μια παραπλήσια ιεραρχική δομή λειτουργίας χρησιμοποιείται και για να περιπλανηθεί κανείς στο διαδίκτυο (world wide web) μέσω μιας δικτυακής πύλης χρησιμοποιώντας το ποντίκι σαν μέσο επικοινωνίας.

Για μεγάλες διαλογικές εφαρμογές, ο χειροκίνητος τρόπος καθορισμού όλων των πιθανών καταστάσεων μετάβασης γίνεται μια πολύ χρονοβόρα διαδικασία. Σ' αυτές τις περιπτώσεις προτείνεται η μέθοδος automatic application and dialog flow generation παράδειγμα της οποίας υπάρχει στις [23,24]. Πιο συγκεκριμένα, δοθέντος μιας δομής ιεραρχικού δέντρου από διαλογικές καταστάσεις η ροή διαλόγου (dialog flow) δημιουργείται αυτόματα ως εξής, το σύστημα μπορεί να μεταβεί σε καταστάσεις οι οποίες είναι κόμβοι παιδιών, συγγενικοί κόμβοι ή πατρικός κόμβος απ' την τρέχουσα κατάσταση της εφαρμογής. Η αυτόματη δημιουργία της διαλογικής ροής μπορεί επίσης να βασιστεί σε πληροφορίες που αφορούν το προφίλ του χρήστη και σε περιορισμούς της βάσης δεδομένων. Η αυτόματη και δυναμική δημιουργία ροής διαλόγου για ένα σύστημα είναι μια νέα ενδιαφέρουσα προοπτική στη διαχείριση διαλόγου.

Ένα πολύ σημαντικό χαρακτηριστικό κάθε καλής εφαρμογής σε ένα διαλογικό σύστημα είναι η προσαρμοστικότητα. Αυτό μπορεί να επιτευχθεί χτίζοντας ένα προφίλ του χρήστη, είτε από συγκεκριμένα ερωτήματα προς το χρήστη, είτε πάλι χρησιμοποιώντας πληροφορίες που σχετίζονται με παλαιότερη αλληλεπίδραση του χρήστη με το σύστημα. Στην [25], ανάλογα με το προφίλ του χρήστη καθορίζονται και οι υπηρεσίες που θα περιέχονται σε μια συγκεκριμένη εφαρμογή, καθώς επίσης και ο τρόπος με τον οποίο το σύστημα θα παρουσιάσει τα δεδομένα στο χρήστη. Για παράδειγμα, σε μια εφαρμογή ανάκτησης πληροφορίας ο χρήστης μπορεί να καθορίσει το είδος των ειδήσεων που θέλει (πολιτικά νέα ή αθλητικά άρθρα κ.α) καθώς και τη μορφή με την οποία θέλει να τα δει (μόνο τους τίτλους των άρθρων κ.α). Η σπουδαιότητα αλλά και χρηστικότητα της προσαρμογής των εφαρμογών και της διαμόρφωσής τους στις ανάγκες του εκάστοτε χρήστη αποδεικνύεται περίτρανα σε δικτυακούς τόπους που ασχολούνται με ειδήσεις και ανάκτηση πληροφορίας.

3.2.4 Σχεδιασμός της Επιφάνειας Αλληλεπίδρασης

Ο σχεδιασμός τη επιφάνειας αλληλεπίδρασης του χρήστη με το σύστημα (Spoken and natural language interface design) έχοντας σα μέσο τη φωνή είναι το πιο ενδιαφέρον στάδιο κατά τη διάρκεια της δημιουργίας ενός διαλογικού συστήματος επικοινωνίας. Καθώς μια σειρά από αρκετά ανοιχτά ερευνητικά προβλήματα αλλά και πολλά πρακτικά θέματα θα πρέπει επιτυχημένα να συνδυαστούν προκειμένου να έχουμε ένα ρεαλιστικό αποτέλεσμα. Επιπλέον τα καλά επίπεδα αναγνώρισης και κατανόησης παίζουν πολύ σημαντικό ρόλο στη δημιουργία μιας καλής επιφάνειας αλληλεπίδρασης. Ακόμη, το διαλογικό σύστημα θα πρέπει να δίνει τη δυνατότητα περιπλάνησης στο χρήστη, παρέχοντάς του εναλλακτικούς τρόπους προκειμένου να ολοκληρώσει μια εργασία και ασφαλώς δεν θα πρέπει να αφήσει το χρήστη να χαθεί ή να παγιδευτεί στο σύστημα. Η ευστροφία, η προσαρμοστικότητα, η συνεργασία και η επίβλεψη (μόνο σε αναγκαίες περιπτώσεις) είναι μερικά από τα χαρακτηριστικά μιας καλής επιφάνειας αλληλεπίδρασης. Αυτή την περίοδο τα ανοιχτά ερευνητικά ζητήματα κατά τη διάρκεια σχεδιασμού ενός διαλογικού συστήματος, συναφή με τις επιδιώξεις που περιγράψαμε παραπάνω, έχουν να κάνουν: με την αντιμετώπιση αυθόρμητων ηχητικών φαινομένων και την αποδοτική αναγνώριση της φωνής, με τη δυναμική δημιουργία γραμματικών βασισμένων σε προγενέστερους διάλογους, με τη δυναμική δημιουργία αποκρίσεων του συστήματος στις οποίες αρέσκεται ο χρήστης βασισμένη στα προηγούμενα δεδομένα χρήσης απ' το χρήστη, με το δυναμικό έλεγχο του χρήστη απ' το σύστημα παρακινούμενο απ' το διάλογο που έχουν σαν συνάρτηση της προόδου του χρήστη. Μερικά πρακτικά θέματα εξίσου όμως σημαντικά που θα πρέπει να λάβουμε υπόψη μας προκειμένου να δημιουργήσουμε ένα ικανό και αποδοτικό πολυμορφικό σύστημα επικοινωνίας είναι η συντονισμένη απόκριση του συστήματος (δηλαδή ο συγχρονισμός ήχου, εικόνας, κειμένου), η διακοπή της ροής διαλόγου του χρήστη ή του συστήματος σε περίπτωση ανίχνευσης εισόδου, η απενεργοποίηση της εισόδου του συστήματος προκειμένου να δώσει απόκριση (system time-out) κ.α. Πρωτότυπα διαλογικών συστημάτων άρχισαν να ενσωματώνουν χαρακτηριστικά ώστε να βελτιώσουν την προσαρμοστικότητα της επιφάνειας αλληλεπίδρασης στη φωνητική είσοδο, πχ, αυτόματη απενεργοποίηση των περιορισμών για τις ερωτήσεις που δεν έχουν αποτέλεσμα, δύο επίπεδα εποπτείας του συστήματος, δυνατότητα επιβεβαίωσης της εισόδου του χρήστη, συντομότερες αποκρίσεις, επιλεκτική χρησιμοποίηση σωστών ουσιαστικών. Επιπλέον, ακόμα πιο φιλόδοξους στόχους θα παρέχει μια επιφάνεια αλληλεπίδρασης με νοημοσύνη και προσωπικότητα.

Μια προσέγγιση έχοντας ως σημείο αναφοράς αυτή τη φορά το χρήστη κατά τη σχεδίαση ενός διαλογικού συστήματος μπορεί να βοηθήσει να διευθετηθούν κάποια από τα θέματα από τα θέματα που σας παρουσιάς παραπάνω. Μια στατική ή δυναμική μοντελοποίηση του χρήστη μπορεί να

χρησιμοποιηθεί ώστε να προβλέψει το σύστημα την πρόθεση του χρήστη. Τμήματα της γραμματικής μπορούν να συσχετιστούν με κάθε μια από τις προθέσεις του χρήστη και να δημιουργηθεί αυτόματα μια εξαρτημένη γραμματική με βάση απ' το συνδυασμό αυτών των τμημάτων της γραμματικής. Επιπλέον, ένα μοντέλο των επιθυμιών των χρηστών μπορεί να χρησιμοποιηθεί για τη δημιουργία των αποκρίσεων του συστήματος: το μοντέλο των επιθυμιών του χρήστη ανανεώνεται σύμφωνα με τις εισόδους του χρήστη και τις αποκρίσεις του συστήματος. Η σχεδιαστική προσέγγιση της γραμματικής του συστήματος έχοντας ως σημείο αναφοράς το χρήστη μπορεί να βοηθήσει ώστε η διαδικασία σχεδιασμού διαλογικών συστημάτων να είναι πιο αυτοματοποιημένη και προσαρμόσιμη στις απαιτήσεις του εκάστοτε συστήματος.

Όπως τονίστηκε παραπάνω, η σχεδίαση ενός πολυμορφικού διαλογικού συστήματος επικοινωνίας εκτείνεται πέραν της υλοποίησης και μόνο των τμημάτων του συστήματος π.χ αναγνωριστή φωνής, σύστημα σημασιολογικού ελέγχου (NLU), διαχειριστή επικοινωνίας (dialog manager). Ένα πολύ σημαντικό στοιχείο στη σχεδίαση διαλογικών συστημάτων είναι η αλληλεπίδραση μεταξύ των δομικών στοιχείων του συστήματος. Για παράδειγμα, η εμπιστοσύνη στα αποτελέσματα του αναγνωριστή μπορεί να βοηθήσει ώστε να βελτιωθεί η απόδοση στο τμήμα του συστήματος που κάνει την κατανόηση, καθώς επίσης και να αποφασίσει για την κατάλληλη διαλογική στρατηγική που θα ακολουθήσει ο διαχειριστή επικοινωνίας (dialog manager). Οι προγενέστεροι διάλογοι μεταξύ του συστήματος και των χρηστών μπορούν να χρησιμοποιηθούν ώστε να βελτιωθεί η απόδοση της αναγνώρισης και κατανόησης του συστήματος (κάνοντας συνεχώς επανεκπαίδευση του συστήματος).

Τέλος, το να καταφέρουμε να σχεδιάσουμε τμήματα διαλογικών συστημάτων τα οποία δεν θα εξαρτώνται από το είδος της εφαρμογής αλλά θα είναι γενικά, σαφώς όμως χρειάζεται να υπάρχει εξάρτηση μεταξύ της εφαρμογής και της σημασιολογικής αναπαράστασης μιας και αυτό είναι απαραίτητο για το σχεδιασμό του τμήματος κατανόησης, είναι ο απώτερος σκοπός στο σχεδιασμό διαλογικών συστημάτων. Γενικοί αλγόριθμοι και εργαλεία που θα μπορούμε να τα χρησιμοποιήσουμε με ελάχιστες μετατροπές για διάφορες εφαρμογές θα επιταχύνουν την ανάπτυξη του τομέα των διαλογικών συστημάτων και θα δημιουργήσουν καινούργιες προκλήσεις.

3.2.5 Η Ανταπόκριση του Χρήστη και η Αποτίμηση του Συστήματος

Ο τρόπος που ανταποκρίνεται ο χρήστης στο σύστημα είναι μια πολύ σημαντική πηγή πληροφορίας για το σχεδιαστή διαλογικών συστημάτων. Παρά την πρόοδο που υπάρχει, ο σχεδιασμός των εφαρμογών αλλά και της επιφάνειας αλληλεπίδρασης του χρήστη με το σύστημα είναι περισσότερο μια

εμπειρική διαδικασία . Η μελέτη της ομάδας των χρηστών στους οποίους απευθύνεται το σύστημα καθώς και της εμπειρίας τους μπορεί να χρησιμοποιηθεί σαν μέτρο αποτίμησης του συστήματος, π.χ να καθορίσει εάν η εφαρμογή και η επιφάνεια αλληλεπίδρασης του συστήματος συναντά τις απαιτήσεις των χρηστών.

Όσον αφορά τώρα την αποτίμηση του συστήματος υπάρχουν κάποιοι μέθοδοι στους οποίους μπορούμε να στηριχτούμε προκειμένου να αποτιμήσουμε διαλογικά συστήματα. Αντικειμενικές μετρικές αποτίμησης υπάρχουν για ορισμένα τμήματα του συστήματος, π.χ η λεξική και σημασιολογική ακρίβεια για τον αναγνωριστή και κατανοητή αντίστοιχα του συστήματος. Αντικειμενικές μετρικές “end-to-end”, όπως ο αριθμός των ολοκληρωμένων εργασιών προς το συνολικό αριθμό των εργασιών και ο αριθμός των «κατανοητών διαλόγων» προς το σύνολο των διαλόγων που διεξήχθησαν, συχνά χρησιμοποιούνται ώστε να αποτιμηθεί η λειτουργικότητα και η αποδοτικότητα των διαλογικών συστημάτων. Σαφώς όμως ο ουσιαστικός κριτής της αποδοτικότητας ή μη ενός συστήματος είναι ο χρήστης. Υποκειμενικά κριτήρια, όπως η αποδοτικότητα του συστήματος, η ικανότητά του να αντιλαμβάνεται, η προσαρμοστικότητα και η ορθή λειτουργία του συστήματος παρέχουν πολύτιμες μετρικές για την γενική αποτίμηση και συνεχή αναβάθμιση ενός συστήματος. Στην [22], μια μίξη υποκειμενικών και αντικειμενικών κριτηρίων χρησιμοποιήθηκε για να αποτιμηθεί το σύστημα ATIS. Στην [26], υπολογίστηκε η συσχέτιση μεταξύ αντικειμενικών (ακρίβεια λέξεων, ρυθμός ολοκλήρωσης εργασιών, ρυθμός κατανοητών διαλόγων) και υποκειμενικών κριτηρίων (ικανοποίηση χρήστη). Το αποτέλεσμα αυτής της συσχέτισης είναι ότι η σχέση μεταξύ των αντικειμενικών κριτηρίων αποτίμησης και της ικανοποίησης του χρήστη είναι λίγο ή πολύ ανεξάρτητα των εφαρμογών του συστήματος.

Κεφάλαιο 4

Υλοποιώντας ένα Πολυμορφικό Σύστημα Επικοινωνίας για Παιδιά.

4.1 Εισαγωγή

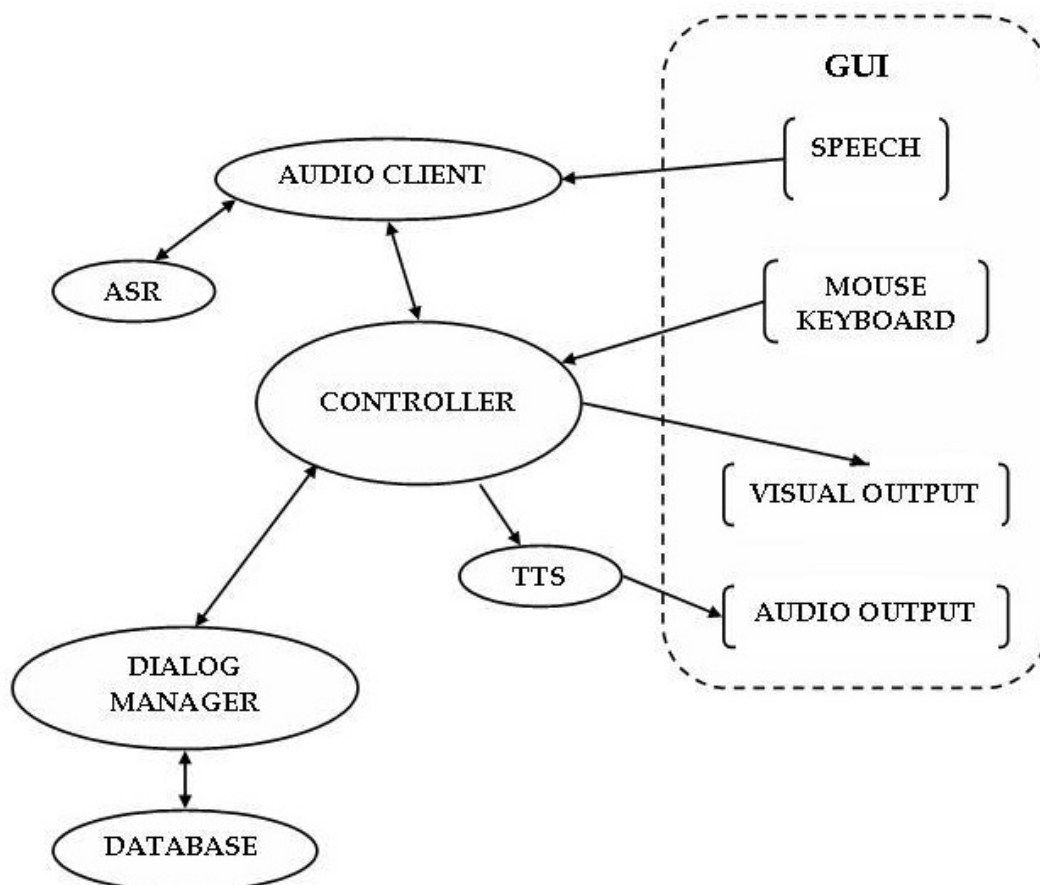
Σ' αυτό το κεφάλαιο θα σας παρουσιάσω την υλοποίηση ενός πολυμορφικού συστήματος επικοινωνίας για παιδιά το οποίο είναι μια νέα έκδοση του πρωτότυπου πολυμορφικού συστήματος επικοινωνίας για παιδιά CHIMP (σκοπός του Children's Interactive Multimedia Project ήταν να δημιουργήσει κάποιες βασικές αρχές σχεδιασμού πολυμορφικών συστημάτων επικοινωνίας για παιδιά με έμφαση στο φωνητικό περιβάλλον αλληλεπίδρασης) [27].

Από το πρωτότυπο σύστημα για το σκοπό αυτό χρησιμοποιήσαμε το γραφικό του περιβάλλον στο οποίο προσθέσαμε επιπλέον λειτουργικότητα καθώς και το σύστημα διαχείρισης (controller), το οποίο επίσης μεταποιήσαμε σε ένα βαθμό. Ενώ υλοποιήθηκαν από την αρχή το σύστημα διαλόγου (dialog manager) καθώς και η φωνητική αλληλεπίδραση με το σύστημα. Ο σχεδιασμός αλλά και η υλοποίησή των επιμέρους συστημάτων βασίστηκαν στις σχεδιαστικές αρχές που σας παρουσίασα αναλυτικά στο κεφάλαιο 3.

4.2 Η Δομή του Συστήματος

Το Γράφημα 1, παρουσιάζει τα κύρια δομικά στοιχεία του συστήματος. Όπως μπορείτε να παρατηρήσετε η καρδιά του συστήματος είναι ο controller ο οποίος διαχειρίζεται το σύστημα. Η επιφάνια επικοινωνίας του συστήματος (user interface) επιτρέπει αλληλεπίδραση με το σύστημα χρησιμοποιώντας φωνή, κείμενο, επιλογή με το ποντίκι καθώς και συνδυασμό αυτών. Η απόκριση του συστήματος στο χρήστη γίνεται παρουσιάζοντάς του ήχου, γραφικά, animation καθώς και γραπτό λόγο. Ο audio client διαχειρίζεται την ηχητική αλληλεπίδραση του συστήματος με το χρήστη. Δέχεται τη φωνή από το χρήστη την οποία στη συνέχεια μετατρέπει σε κείμενο μέσω του ASR και τη

στέλνει στον controller. Ο dialog manager επεξεργάζεται τα δεδομένα που δίνει ο χρήστης στο σύστημα και δημιουργεί την απόκριση του συστήματος. Επίσης ο dialog manager συνεργάζεται με μια πηγή πληροφορίας (database) για ανάκτηση ή καταχώρηση δεδομένων. Πιο λεπτομερής ανάπτυξη για καθένα από τα δομικά στοιχεία καθώς και του τρόπου επικοινωνίας τους, θα γίνει στην παράγραφο 4 (Αρχιτεκτονική του Συστήματος) .

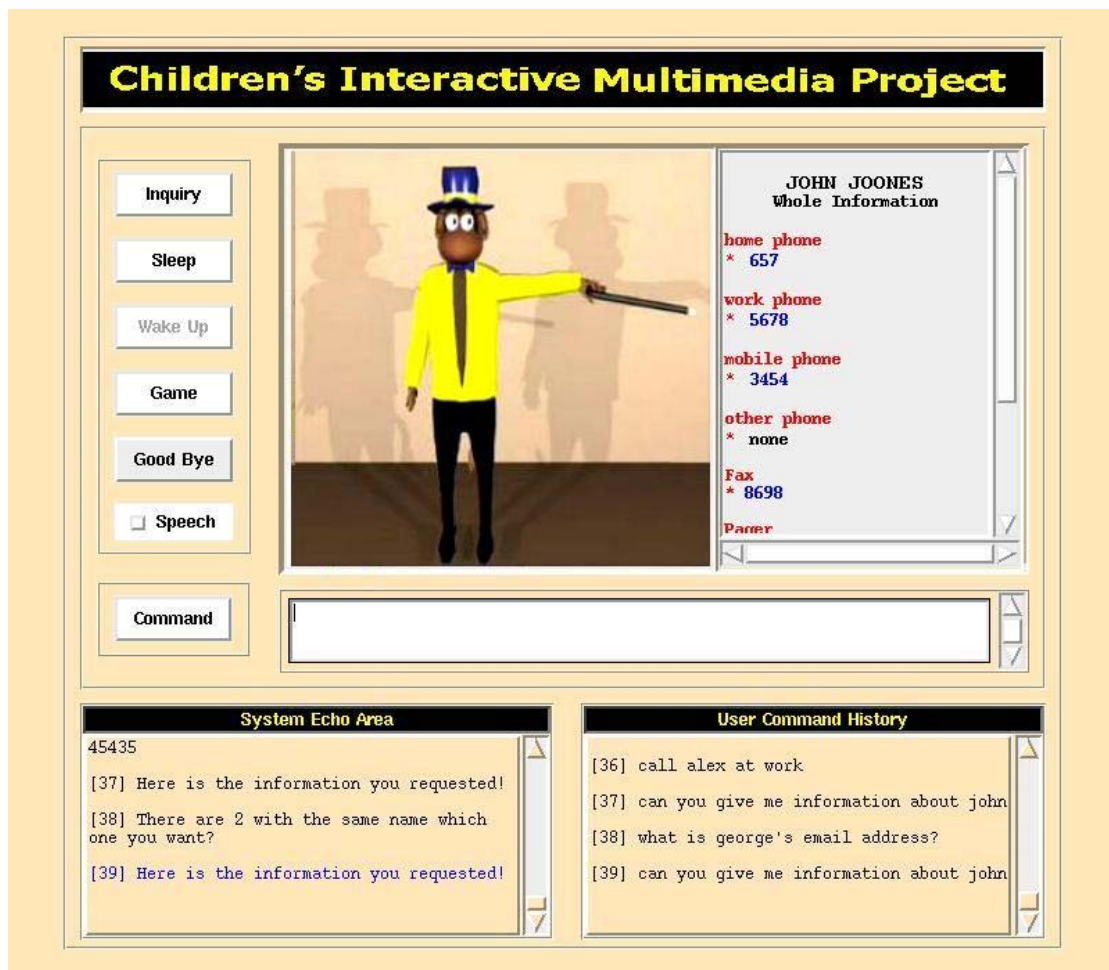


Γράφημα. 1. Το Διάγραμμα Ροής του Πολυμορφικού Συστήματος

Το σύστημα αποτελείται από τα εξής στοιχεία, όπως φαίνεται και στο Γράφημα 3: *input/output event handler(controller)*, *dialog manager*, *graphical user interface (Gui)*, *audio client*, *speech recognizer (ASR)*, *speech synthesis (TTS)*, *animator* and *database*.

Ο *input/output event handler* είναι ο controller του συστήματος, είναι αυτός που διαχειρίζεται όλα τα στοιχεία του συστήματος και συγχρονίζει την (ασύγχρονη) είσοδο από το χρήστη (φωνή, πληκτρολόγιο ή ποντίκι).

Ο *dialog manager* είναι αυτός που ορίζει τη στρατηγική και την ενέργεια στην οποία θα πρέπει να προβεί το σύστημα, επεξεργάζοντας την είσοδο του χρήστη. Στη συνέχεια αποφασίζει για την απόκριση αλλά και τον τρόπο παρουσίασής της στον χρήστη.



Γράφημα. 2. Η Όψη του Γραφικού Περιβάλλοντος Επικοινωνίας του Συστήματος (GUI).

Το *graphical user interface* (Gui) είναι το γραφικό περιβάλλον επικοινωνίας του συστήματος με το χρήστη, το οποίο αποτελείται από πέντε κύριες περιοχές, όπως βλέπεται και στο Γράφημα 2: την περιοχή γραφικών, την περιοχή του command line όπου εκεί ο χρήστης πληκτρολογεί τις εντολές του, την περιοχή κουμπιών, την περιοχή όπου παρουσιάζονται οι τελευταίες εντολές που έδωσε ο χρήστης (user command history) και την περιοχή όπου παρουσιάζονται οι αντίστοιχες, ως προς τις τελευταίες εντολές του χρήστη, αποκρίσεις του συστήματος στο χρήστη (system echo area). Ο τρόπος σχεδιασμού του Gui έγινε με τέτοιο τρόπο ούτως ώστε να υπάρχει μια συνέχεια τόσο στην όψη όσο και στην αίσθηση του Gui μεταξύ των διάφορων εφαρμογών του. Η προσωπικότητα και η εμφάνιση του κινούμενου χαρακτήρα προσέχτηκε ιδιαίτερα, μιας και απευθυνόταν σε παιδιά που είναι αρκετά απαιτητικοί χρήστες, ώστε να παρέχει στο χρήστη προσαρμοστικότητα

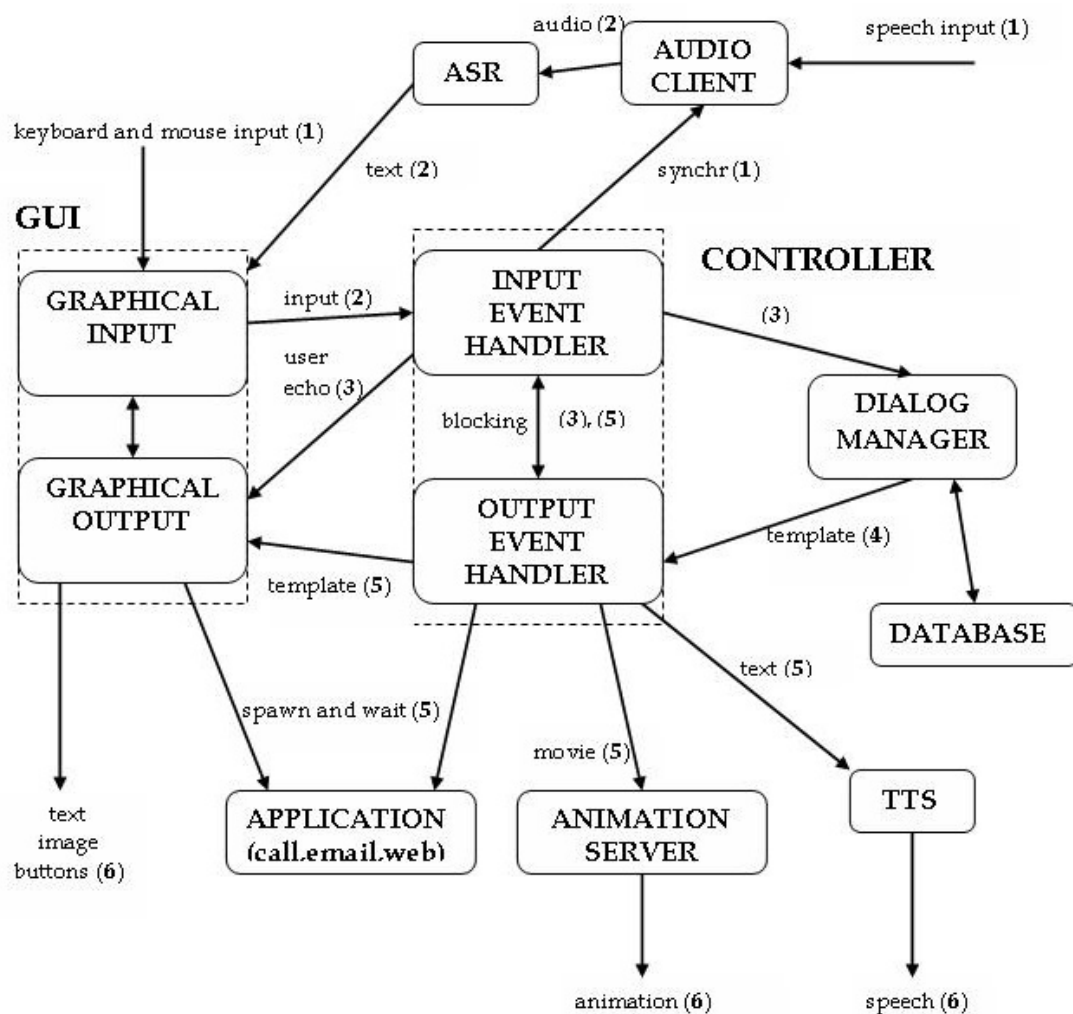
και άνεση. Τα κουμπιά που υπάρχουν στο Gui δίνουν τη δυνατότητα στο χρήστη να εκτελέσει αρκετές εντολές οι οποίες βεβαίως μπορούν να εκτελεστούν και με άλλους τρόπους, είτε πληκτρολογώντας αυτές, είτε χρησιμοποιώντας τη φωνή. Επίσης ο χρήστης έχει τη δυνατότητα επιλέγοντας με το ποντίκι στην περιοχή user command history μια από τις παλιότερες εντολές που είχε δώσει και κλικάροντάς την στη συνέχεια αυτή να επαναληφθεί, όπως επίσης κάθε φορά που πληκτρολογεί μια εντολή στο command line αυτή να δίδεται προς εκτέλεση είτε πατώντας enter στο πληκτρολόγιο, είτε πατώντας το κουμπί command στο Gui.

Ο *audio client* διαχειρίζεται τον αναγνωριστή (ASR server). Ο audio client συγχρονίζεται από τον controller, όπως βλέπεται και στο Γράφημα 3. Κάθε φορά που το σύστημα δέχεται είσοδο φωνής από τον χρήστη ο audio client την μετατρέπει σε κείμενο μέσω του ASR server.

Το *speech synthesis (TTS)* είναι υπεύθυνο για την ηχητική απόκριση του συστήματος. Διαχειρίζεται από τον output event handler, ο οποίος του στέλνει την απόκριση του συστήματος σε μορφή κειμένου και τη μετατρέπει σε φωνή.

Ο *animator* είναι υπεύθυνος για την παρουσίαση των animation στο Gui. Ο animator συγχρονίζεται από τον output event handler, ο οποίος του στέλνει το κατάλληλο κάθε φορά βίντεο που πρέπει να παρουσιάσει.

Η *database* περιέχει δεδομένα που καταχωρεί ο χρήστης του συστήματος όπως, ονόματα, τηλέφωνα, ηλεκτρονικές διευθύνσεις, διευθύνσεις ηλεκτρονικού ταχυδρομείου, fax και pager. Τη database τη διαχειρίζεται ο dialog manager και ανάλογα με τις εντολές του χρήστη κάνει ανάκτηση ή καταχωρεί σε αυτή πληροφορία.



Γράφημα. 3. Αρχιτεκτονικό Διάγραμμα Λειτουργίας του Συστήματος.

4.3 Τρόπος Λειτουργίας του Συστήματος

Στο Γράφημα 3, παρουσιάζεται μια σχηματική διάταξη λειτουργίας του συστήματος η οποία είναι χρονικά αριθμημένη σύμφωνα με τις ενέργειες που λαμβάνουν χώρα από τη στιγμή που ο χρήστης θα δώσει κάποια μορφής είσοδο, μέχρι την απόκριση του συστήματος για την ενέργεια αυτή.

Όπως παρατηρείτε, στη διάταξη υπάρχουν δύο δρόμοι εισόδου της πληροφορίας στο σύστημα. Ο ένας είναι να δώσει ο χρήστης είσοδο στο σύστημα πληκτρολογώντας κάποια εντολή στο command line ή χρησιμοποιώντας το ποντίκι, κλικάροντας κάποιο από τα κουμπιά του Gui (1) στη συνέχεια η πληροφορία μεταβιβάζεται από το Gui στον input event handler. Ο άλλος τρόπος εισόδου είναι ο χρήστης να χρησιμοποιήσει φωνή για να δώσει εντολή στο σύστημα, για να γίνει αυτό πρέπει πρώτα να πατήσει

το κουμπί speech, μιας και το σύστημά μας είναι click-to-talk μορφής, στο Gui και μετά να δώσει τη φωνητική εντολή. Μόλις λοιπόν πατήσει το κουμπί speech, ο input event handler στέλνει σήμα στον audio client (ενεργοποιεί τον audio client), ο οποίος καταγράφει τη φωνή από το χρήστη και τη στέλνει στον ASR server για αναγνώριση, όπου αφού γίνει η αναγνώριση, το κείμενο που προκύπτει τοποθετείται στο command line του Gui (2) και εν συνεχεία μόλις ο χρήστης έχει πει την εντολή, το κείμενο μεταβιβάζεται στον input event handler και ο audio client απενεργοποιείται. Από εδώ και πέρα η λειτουργία του συστήματος είναι κοινή. Μόλις λοιπόν φτάσει η είσοδος στον input event handler υπό μορφή κειμένου (2) αυτός εν συνεχεία απενεργοποιεί τον output event handler, στέλνει την είσοδο στο dialog manager για επεξεργασία καθώς και στο Gui, για να την τοποθετήσει στο system echo area (3). Έπειτα ο dialog manager παίρνει την είσοδο και την υποβάλλει σε σημασιολογικό έλεγχο, χρησιμοποιώντας συνήθως γι' αυτή την ενέργεια και δεδομένα από τη βάση. Αφού ερμηνεύσει την εντολή του χρήστη αποφασίζει για την ενέργεια που θα εκτελέσει το σύστημα καθώς και για την απόκριση που θα δώσει στο χρήστη (π.χ. τι θα απαντήσει το σύστημα, ποιο animation θα παίξει). Έχοντας αποφασίσει για όλα αυτά τα ζητήματα ο dialog manager στέλνει όλη αυτή την πληροφορία στον output event handler (4) ο οποίος έχει προηγουμένως ενεργοποιηθεί. Ο output event handler, απενεργοποιεί τον input event handler, διαβάζει την πληροφορία που του έστειλε ο dialog manager και στέλνει το κείμενο της απόκρισης στον TTS server (5), ο οποίος με τη σειρά του το μετατρέπει σε φωνή δημιουργώντας έτσι τη φωνητική απόκριση του συστήματος (6). Επίσης ο output event handler στέλνει στον animation server το βίντεο που θα παίξει (5), εκτελεί (αν έχει ανάλογη εντολή από τον dialog manager) κάποια από τις εφαρμογές (πχ calling) και στέλνει πληροφορία στο Gui (5). Το Gui μόλις λάβει την πληροφορία εκτελεί τις εντολές που αυτή του καθορίζει, παράγοντας έτσι το σύστημα απόκριση στο χρήστη (6).

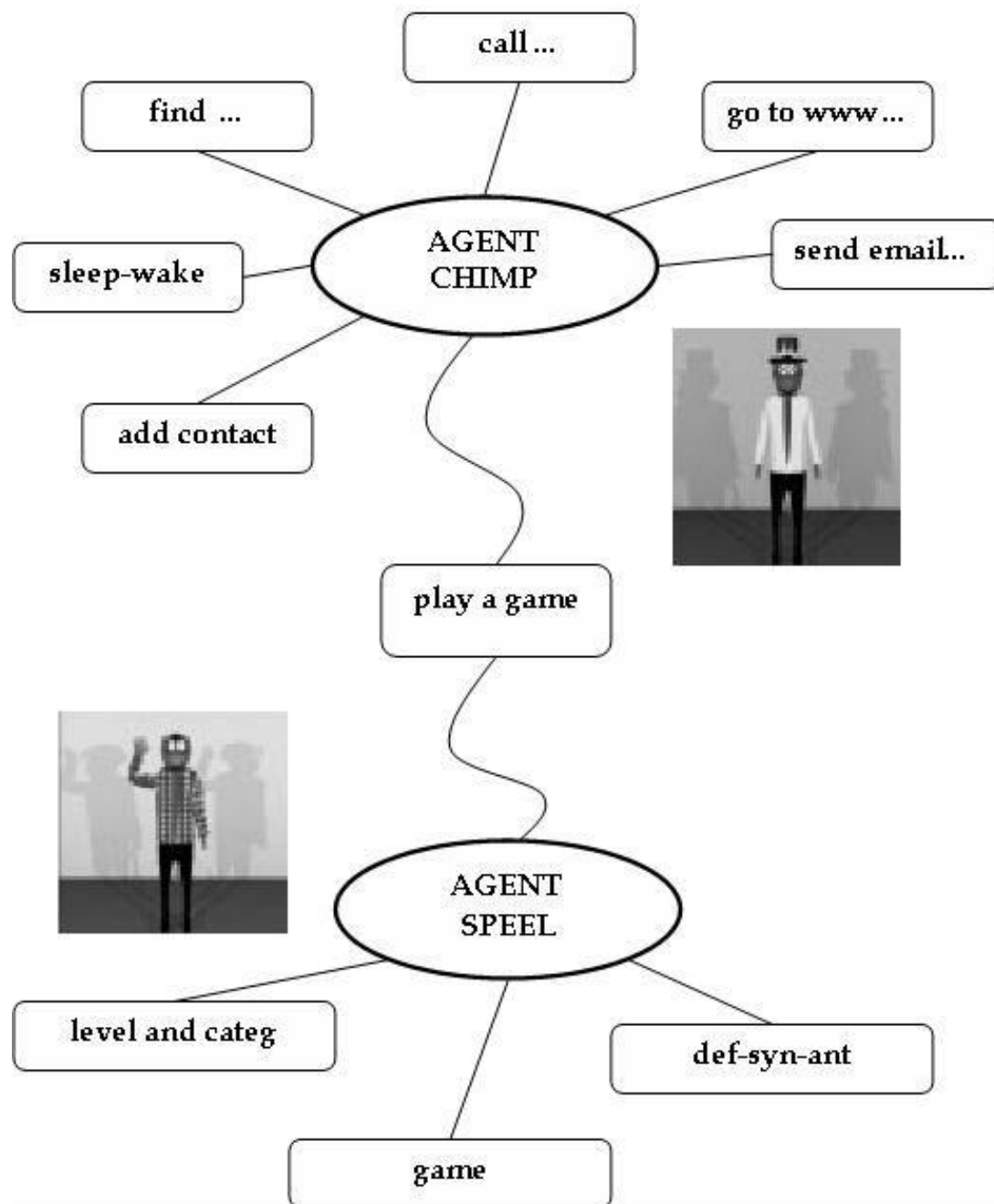
4.4 Οι Εφαρμογές του Συστήματος

Στο προηγούμενο εδάφιο σας παρουσίασα εκτενώς τον τρόπο λειτουργίας του συστήματός αναφέροντάς σας. Στο παρών εδάφιο θα σας παρουσιάσω λεπτομερώς όλο εύρος των εφαρμογών του συστήμα.

4.4.1 Η Λειτουργικότητα του Συστήματος

Όπως μπορείτε να παρατηρήσετε στο Γράφημα 4, το σύστημα αποτελείται από δύο ξεχωριστές εφαρμογές. Μια εφαρμογή communication agent (κάνει ανάκτηση πληροφορίας από database, τηλεφωνεί, στέλνει e-mails, καταχωρεί δεδομένα στη βάση και δίνει πρόσβαση στο διαδίκτυο) και ένα παιχνίδι λέξεων (spelling bee).

Οι Εφαρμογές του Συστήματος



Γράφημα. 4. Γενική Εικόνα των Εφαρμογών του Συστήματος: Ο Agent Chimp διαχειρίζεται τις προσωπικές εφαρμογές και ο Agent Spell διαχειρίζεται το παιχνίδι λέξεων.

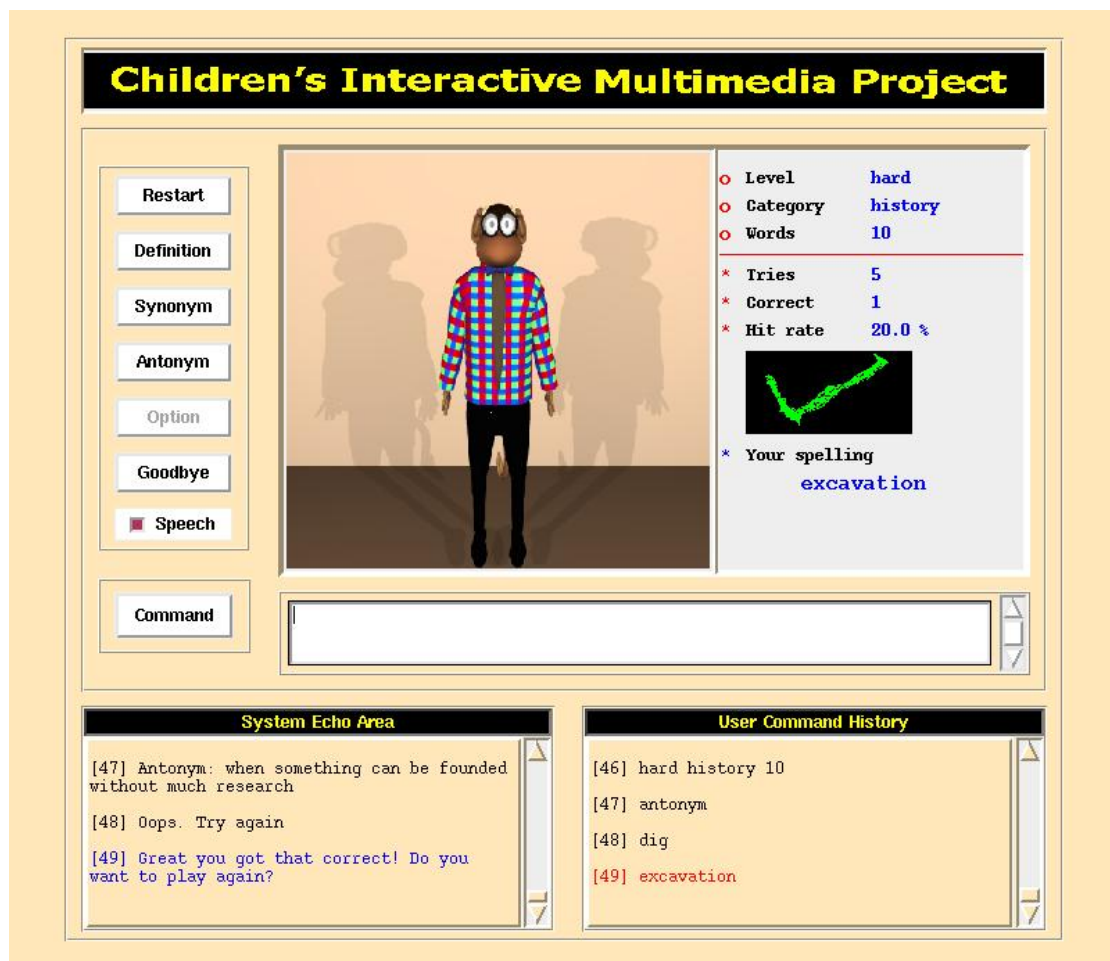
Και οι δύο εφαρμογές διαχειρίζονται από ομιλούντα κινούμενα σχέδια που αναπαριστούν δύο μαιμουδάκια, τον Agent Chimp στην εφαρμογή επικοινωνίας και τον Agent Spell στο παιχνίδι. Η προσωπικότητα και η εμφάνιση των οποίων διαφέρει μεταξύ των δύο εφαρμογών. Η προσωπικότητα, ο τρόπος ομιλίας αλλά και η εμφάνιση του Agent Chimp

σχεδιάστηκαν με τέτοιο τρόπο ώστε το αποτέλεσμα να είναι ένας ευγενικός και με σοφιστικέ ύφος personal agent, Γράφημα 2. Από την άλλη μεριά ο Agent Spell, που διαχειρίζεται το παιχνίδι λέξεων, είναι μια καρικατούρα επιμελούς προσωπικότητας, Γράφημα 5.

Ο Agent Chimp δίνει τη δυνατότητα στο χρήστη να τηλεφωνήσει σε κάποιο πρόσωπο που είναι καταχωρημένο στη βάση. Βέβαια αυτός δεν είναι ο μόνος τρόπος για να εκτελέσει κάποιος ένα τηλεφώνημα μέσω του Agent Chimp, ένας άλλος τρόπος είναι να ζητήσουμε πληροφορίες για κάποιο άτομο και αφού το βρει και μας τις εμφανίσει να κλικάρουμε με το ποντίκι πάνω στο τηλέφωνο που μας ενδιαφέρει και να πραγματοποιηθεί η κλήση. Μια άλλη λειτουργικότητα του Agent Chimp είναι η πρόσβαση στο διαδίκτυο, δίνοντας έτσι τη δυνατότητα στο χρήστη να σερφάρει. Όπως προείπαμε και παραπάνω αυτός δεν είναι ο μόνος τρόπος για να σερφάρουμε, αλλά μπορούμε και με τον ίδιο τρόπο όπως και κατά τη λειτουργία του τηλεφωνήματος να το επιτύχουμε αυτό. Επίσης μπορούμε μέσω του συστήματος να στείλουμε πάρα πολύ εύκολα emails, πληκτρολογώντας το απλά σε ένα pop up πλαίσιο και πατώντας το κουμπί Send να το στείλουμε στον προορισμό που έχουμε επιλέξει. Επίσης μπορούμε από το σύστημα να ζητήσουμε πληροφορίες. Και πιο συγκεκριμένα μπορούμε να ζητήσουμε γενικές πληροφορίες για ένα άτομο, οπότε σ' αυτή την περίπτωση το σύστημα μας βγάζει όλη την πληροφορία που έχει για το άτομο αυτό. Όπως επίσης μπορούμε να ζητήσουμε να μας πληροφορήσει το σύστημα και για συγκεκριμένα πράγματα, π.χ. πιο είναι το fax ή ο pager ή το url ή το e-mail κάποιου ατόμου και στη συνέχεια ο χρήστης έχει τη δυνατότητα αν θέλει κλικάροντας πάνω σ' αυτά, να εκτελεί ανάλογες με τα δεδομένα πληροφορίας ενέργειες. Το σύστημα έχει επίσης χαρακτηριστικά "go to sleep" και "wake up" με τη βοήθεια των οποίων η λειτουργία του personal agent είναι ελεγχόμενη από το χρήστη. Ο Agent Chimp παρέχει στο χρήστη τη δυνατότητα να καταχωρεί δεδομένα (όνομα, τηλέφωνα, fax, pager, url και e-mail) στη βάση κλικάροντας το κουμπί Inquiry και συμπληρώνοντας τα στοιχεία. Τα δεδομένα πριν καταχωρηθούν στη βάση ελέγχονταν για τη μορφή τους (πχ email χωρίς "@", url χωρίς "www.") και απαραίτητη προϋπόθεση είναι, να είναι συμπληρωμένο το πεδίο του ονόματος. Τα δεδομένα στη βάση είναι δύο μορφών, Personal friend και Business friend ανάλογα με το τι σχέση έχει ο χρήστης με τα άτομα αυτά. Όπως είπαμε το σύστημά μας περιέχει και ένα παιχνίδι λέξεων για το οποίος όμως υπεύθυνος είναι ο Agent Spell, οπότε ζητώντας του να παίξουμε το παιχνίδι μεταπηδάμε στον Agent Spell. Αναλυτική περιγραφή του τρόπου λειτουργίας του παιχνιδιού αλλά και του Agent Spell γίνετε παρακάτω.

Ο Agent Spell είναι όπως σας είπα ένα παιχνίδι λέξεων, όπου ο παίχτης προσπαθεί να μαντέψει μια κρυμμένη λέξη που έχει θέσει το σύστημα. Ο παίκτης μπορεί να ζητήσει από το σύστημα να του δώσει τον ορισμό της λέξης, ένα συνώνυμό της ή ένα αντίθετο της, ώστε να βοηθηθεί και να βρει τη λέξη. Ο κάθε παίκτης έχει τρεις ευκαιρίες για να βρει την κρυμμένη λέξη. Πριν

ξεκινήσει να παίζει ο παίκτης του ζητάτε από το σύστημα να δώσει επίπεδο (easy,medium,hard) και κατηγορία (general,science,history) λέξης με την οποία θέλει να παίζει. Αυτό μπορεί να γίνει είτε γράφοντας ο παίκτης το level και το category καθοδηγούμενος από το σύστημα, είτε επιλέγοντας το κουμπί Option και καθορίζοντας από εκεί αυτές τις παραμέτρους. Σχηματικά η λειτουργία του Agent Spell περιγράφεται με την fsm που αναπαριστά το Γράφημα 7.



Γράφημα.5. Το Γραφικού Περιβάλλοντος Επικοινωνίας του Agent Spell

Η λέξη που θέτει το σύστημα επιλέγεται τυχαία μέσα από διάφορες λέξεις που υπάρχουν σε μια βάση ώστε κάποιος που παίζει πολλές φορές να μην μαθαίνει τη σειρά των λέξεων. Επίσης ο παίκτης έχει τη δυνατότητα να παίζει όσες φορές θέλει, και για να είναι πιο ενδιαφέρον το παιχνίδι μετριοούνται οι προσπάθειές του και το σύστημα του βγάζει ποσοστό επιτυχίας επί των κρυμμένων λέξεων.

Τέλος ο χρήστης βρισκόμενος στον Agent Spell μπορεί ανά πάσα στιγμή να κάνει Restart και να ξεκινήσει καινούργιο παιχνίδι, είτε να πατήσει goodbye και να βγει από τον Agent Spell και να επιστρέψει στον Agent Chimp. Η

γραμματική που ισχύει στον Agent Chimp δεν ισχύει στον Agent Spell και αντίστροφα.

Ο σχεδιασμός του Agent Spell είναι ντετερμινιστικός και υλοποιείται όπως σας ανέφερα πρωτότερα σύμφωνα με την fsm που αναπαριστάται στο Γράφημα 7.

4.4.2 Τα Διαλογικά Χαρακτηριστικά του Συστήματος

Ο σχεδιασμός της γραμματικής του dialog manager έγινε με τέτοιο τρόπο ώστε το σύστημα να κατανοεί είσοδο από το χρήστη ο οποίος χρησιμοποιεί για το σκοπό αυτό φυσική γλώσσα.

Επίσης ο σχεδιασμός του dialog manager έγινε με τέτοιο τρόπο ώστε να υποστηρίζει πρωτοβουλίες του χρήστη, όσον αφορά τη στρατηγική αλλά και πρωτοβουλίες στρατηγικής του συστήματος. Το σύστημα έχει μεριμνήσει για προβλήματα αμφισημίας (π.χ. στην περίπτωση ύπαρξης πολλαπλών εγγραφών για μια συγκεκριμένη πληροφορία), καθώς και για λάθη στον έλεγχο (χρησιμοποιώντας σίγουρα κριτήρια επιλογής) υλοποιώντας συγκεκριμένες για το λόγο αυτό διαλογικές καταστάσεις.

Το βίντεο είναι ένα πολύ ισχυρό εργαλείο στο πολυμορφικά συστήματα, τα animation που χρησιμοποιούμε στο σύστημα παρέχουν σε αυτό μια καίρια μορφή παρουσίασης καθώς συνδυάζουν ένα ευχάριστο Interface και ένα περιβάλλον προσαρμογής ως προς την επόμενη διαλογική κατάσταση. Επίσης η μετάλλαξη της προσωπικότητας του agent κάθε φορά που ο χρήστης μεταφέρεται από τις εφαρμογές επικοινωνίας στο παιχνίδι λέξεων και αντίστροφα, βοηθάει στην προσαρμοστικότητα του χρήστη, παρέχοντάς του μια πιο ομαλή μετάβαση από το ένα επίπεδο του συστήματος στο άλλο. Κάτι που επιτυγχάνεται στο σύστημα με τη χρήση ενός animation στο οποίο ο παλιός agent χάνεται πίσω από μια κουρτίνα που πέφτει και στη συνέχεια εμφανίζεται ο καινούργιος όταν ανοίγουν οι κουρτίνες.

Τα animation που χρησιμοποιούμε στο σύστημα πέρα από μια επιπλέον μορφή εξόδου παρέχουν και χρήσιμα διαλογικά χαρακτηριστικά, γι' αυτό χρησιμοποιούνται: δεικτικές χειρονομίες για παρουσίαση πληροφορίας (κάτι που είναι ιδιαίτερος χρήσιμο στις περιπτώσεις που ο χρήστης πρέπει να επιλέξει μέσα από μια λίστα επιλογών, Γράφημα 2, σήκωμα των ώμων σε περίπτωση που δεν υπάρχει καταχώρηση ή σε περιπτώσεις λάθους και γνέψιμο του κεφαλιού δεξιά-αριστερά σε περιπτώσεις αποτυχίας είτε κουνώντας το καταφατικά σε περιπτώσεις επιτυχίας.

4.5 Η Αρχιτεκτονική του Συστήματος

Ένα αρχιτεκτονικό διάγραμμα του συστήματος είναι αυτό που φαίνεται στο Γράφημα 3, το οποίο εστιάζει στον τρόπο επικοινωνίας μεταξύ των στοιχείων του συστήματος. Η επικοινωνία των στοιχείων του συστήματος γίνεται με τη χρήση TCP/IP socket συνδέσεων. Η πληροφορία διαβιβάζεται μεταξύ των στοιχείων με τη μορφή μηνυμάτων, τα οποία περιέχουν όλη την απαραίτητη για τον εκάστοτε εξυπηρετητή πληροφορία και έτσι οι εξυπηρετητές καθίστανται stateless.

Ο controller είναι ένας ασύγχρονος διαχειριστής γεγονότων (input/output event handler) και περιλαμβάνει δύο ξεχωριστούς βρόχους διαχείρισης γεγονότων, έναν για την διαχείριση των γεγονότων εισόδου στο σύστημα και έναν για τη διαχείριση των γεγονότων εξόδου από αυτό. Η υλοποίησή του έγινε σε γλώσσα Perl και η ροή του προγράμματος μεταξύ των δύο βρόχων διαχείρισης είναι σχετικά απλή. Έτσι στο βρόχο εισόδου τα γεγονότα εισόδου υφίστανται μια προεπεξεργασία και εν συνεχεία στέλνονται στο dialog manager με τη μορφή συγκεκριμένου κειμένου (δεν υπάρχουν κεφαλαία γράμματα, σημεία στίξης και μεταξύ των λέξεων υπάρχει μόνο ένα κενό διάστημα). Ενώ στο δεύτερο βρόχο, τα templates (τα οποία είναι μια δομή δεδομένων που χρησιμοποιείται για να διαβιβαστεί πληροφορία μεταξύ των διαφόρων υποσυστημάτων) που λαμβάνονται από το dialog manager αναλύονται για να τροφοδοτηθούν τα πεδία της πολυμορφικής εξόδου. Στη συνέχεια ο output event handler, αφού έχει αναλύσει και έχει πάρει όλες τις απαραίτητες πληροφορίες από το template, τροφοδοτεί το κάθε πεδίο με τα απαραίτητα γι' αυτό δεδομένα (κείμενο για τον TTS synthesizer, βίντεο για τον animation server, template για το Gui). Αν κατά τη διάρκεια της επεξεργασία των δεδομένων από το dialog manager το σύστημα δεχθεί κάποια είσοδο, τα γεγονότα εισόδου αποθηκεύονται σε μια στοίβα από την οποία εκτελούνται στη συνέχεια. Εάν πολλαπλά γεγονότα συμβούν κατά τη διάρκεια της επεξεργασία των δεδομένων από το dialog manager μόνο το τελευταίο εκτελείται. Επίσης κατά τη διάρκεια εισόδου γεγονότων στον input event handler, γίνεται block στον output event handler και αντίστροφα. Τέλος, ο controller έχει την ικανότητα να διαχειρίζεται γεγονότα παρεμβολής από το χρήστη (δηλ. ομιλία ή πληκτρολόγηση καθώς παίζει ο TTS ή ο animator ή ο ASR server), ενημερώνοντας τον TTS synthesizer ή τον animator server ή τον ASR server να διακόψουν τη λειτουργία τους.

Το Gui (Graphical user interface) είναι ένα ασύγχρονο πρόγραμμα το οποίο διαχειρίζεται τις εισόδους και εξόδους του συστήματος από και προς το χρήστη. Έχει υλοποιηθεί σε γλώσσα Tcl/Tk και αποτελείται από δύο βρόχους. Ο πρώτος βρόχος (Graphical Input) παίρνει την είσοδο που δίνει ο χρήστης στο Gui (πατώντας κουμπιά ή πληκτρολογώντας στο command line) και τη γράφει σε μία στοίβα, από την οποία τροφοδοτείται ο event handler. Στο σημείο αυτό να σας αναφέρω ότι πατώντας ο χρήστης καθένα από τα κουμπιά του Gui, καταχωρείτε στη στοίβα η αντίστοιχη εντολή υπό μορφή κειμένου

(π.χ πατά το κουμπί: sleep, στη στοίβα καταχωρείται: go to sleep). Ο δεύτερος βρόχος (Graphical Output) είναι υπεύθυνος για την απόκριση του συστήματος, μιας και αυτός εκτελεί τις ενέργειες που έχει ορίσει ο dialog manager. Πιο συγκεκριμένα, το graphical output παίρνει την είσοδο του χρήστη από τον input event handler μέσω του user echo και την τοποθετεί στο πεδίο user command history του Gui. Στη συνέχεια παίρνει το template που του στέλνει ο output event handler από το οποίο παίρνει πληροφορία για το πια ενέργεια θα εκτελέσει, την απάντηση του συστήματος στην εντολή του χρήστη καθώς και πληροφορία απαραίτητη για την εκτέλεση των εφαρμογών (e-mail address, url). Αφού αποκωδικοποιήσει το template ξεκινά να ανανεώνει το Gui προκαλώντας έτσι την απόκριση του συστήματος, αλλάζει το γραφικό του Gui σύμφωνα με την ενέργεια που εκτελεί και προβάλλει σ' αυτό την πληροφορία που του ζήτησε ο χρήστης. Τοποθετεί την απάντηση του συστήματος στο πεδίο system echo area και εκτελεί, αν έχει τέτοια εντολή από τον dialog manager, κάποια εφαρμογή (open web browser, pop up e-mail application).

Ο TTS synthesizer που χρησιμοποιώ για παραχθεί η φωνητική απόκριση του συστήματος είναι ο Freetts-1.2.1. Είναι ένα java αρχείο το οποίο εκτελεί ο output event handler κάθε φορά που το σύστημα αποκρίνεται στο χρήστη.

Ο animator που χρησιμοποιούμε στο σύστημα προκειμένου να παράγουμε animation βίντεο στην απόκριση είναι ο mplayer, ο οποίος εκτελείται από τον output event handler.

Τέλος, λόγου του ότι το σύστημά μας είναι ασύγχρονο κατά την έναρξη της λειτουργίας του συστήματος φορτώνουμε μια εντολή και συγκεκριμένα την εντολή "Hi" στη στοίβα των εντολών.

Στην παράγραφο αυτή δεν σας παρουσίασα την αρχιτεκτονική του dialog manager καθώς και του συστήματος αναγνώρισης φωνής, κάτι που έγινε σκοπίμως, μιας και ο dialog manager καθώς και το σύστημα της φωνής αποτελούν το πιο πολύπλοκο αλλά και το ουσιαστικό κομμάτι της διπλωματικής μου εργασίας. Η αναλυτική περιγραφή τους ακολουθεί στις επόμενες παραγράφους.

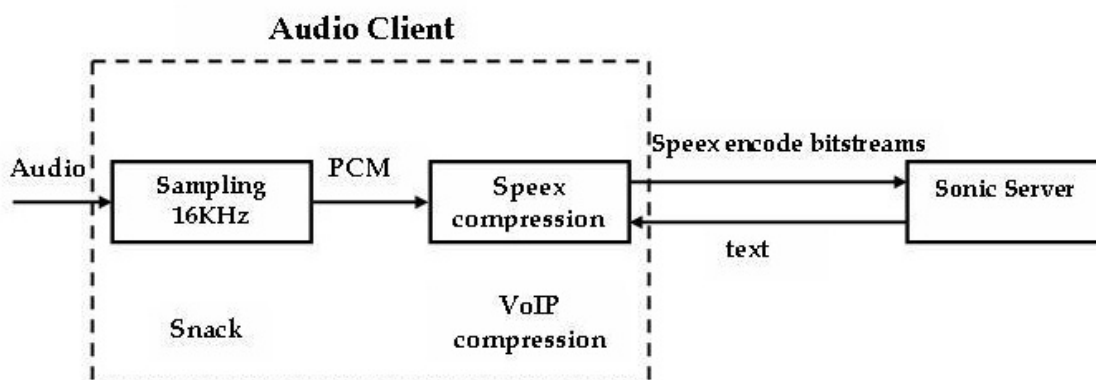
4.6 Σύστημα Αναγνώρισης Φωνής (audio client - ASR)

Στην παράγραφο αυτή θα σας παρουσιάσω αναλυτικά την αρχιτεκτονική που χρησιμοποίησα προκειμένου να υλοποιήσω το κομμάτι που κάνει την αναγνώριση της ομιλίας στο σύστημα. Όπως σας ανέφερα και στην παράγραφο 4.3 (Τρόπος Λειτουργίας του Συστήματος), ο τρόπος που χρησιμοποιούμε τον αναγνωριστή στο σύστημα είναι της μορφής click-to-talk.

Ο αναγνώριστης που χρησιμοποιήσαμε είναι ο Sonic του University of Colorado τον οποίο και έχουμε χρησιμοποιήσει σε μορφή live-mode client/server, αφού θέλαμε να δουλεύει σε μορφή click-to-talk. Δηλαδή υπάρχει ένας client ο οποίος είναι συνδεδεμένος μέσω socket TCP/IP με τον sonic-server και κάθε φορά που ο χρήστης κλικάρει το κουμπί speech στο Gui, ο client λαμβάνει την ομιλία του χρήστη και τη στέλνει στον sonic-server για αναγνώριση. Ο οποίος με τη σειρά του, αφού κάνει αναγνώριση επιστρέφει το κείμενο της αναγνώρισης στον client, σχηματικά η διαδικασία της αναγνώρισης παρουσιάζεται στο Γράφημα 6. Παρακάτω θα σας παρουσιάσω εκτενώς όλη τη διαδικασία.

Το ρόλο του client στο σύστημά μας τον παίζει ο audio client. Ο οποίος είναι ένα αρχείο γραμμένο σε Tcl/Tk και το οποίο μόλις ξεκινήσουμε το σύστημα συνδέεται μέσω ενός port (5555) με τον sonic-server. Ο audio client λοιπόν κάθε φορά που ο χρήστης κλικάρει το κουμπί speech στέλνει ένα σήμα στο server ότι θα δεχθεί δεδομένα προς αναγνώριση και ξεκινά να δειγματοληπτεί τη φωνή του χρήστη με συχνότητα δειγματοληψία 16 KHz και στη συνέχεια τη μετατρέπει σε PCM μορφή. Αυτό το κάνει με τη βοήθεια του Snack programming language, που είναι κάτι σαν βιβλιοθήκες για να διαχειρίζεται κάποιος ήχο σε Tcl/Tk και Python.

Ο Sonic στη μορφή client/server δέχεται κωδικοποιημένα τη συνεχόμενη ροή ήχου μέσα στο socket και πιο συγκεκριμένα, χρησιμοποιεί μια κωδικοποίηση που ανέπτυξε ο Jean-Mark Valin γνωστή ως Speex. Όπου για 16KHz PCM audio έχουν ρυθμό μετάδοσης δεδομένων 31.25 kB/s. Γι' αυτό το λόγο το sonic παρέχει το VoIP (Voice over IP) compression server το οποίο μετατρέπει PCM sampled audio σε Speex κωδικοποίηση.



Γράφημα .6.Το Διάγραμμα Ροής του Συστήματος Φωνής

Όπως βλέπουμε και στο Γράφημα 6, αφού μετατρέψουμε τον ήχο σε PCM μορφή, στη συνέχεια τον κωδικοποιούμε σε Speex μορφή και τον στέλνουμε στον sonic-server για αναγνώριση, ο οποίος αφού κάνει αναγνώριση στέλνει

το κείμενο που προέκυψε πίσω στον client. Ο audio client εν συνεχεία το στέλνει Gui όπου και τοποθετείται στο command line μέχρι να τελειώσει ο χρήστης την φωνητική εντολή. Μόλις ο χρήστης τελειώσει και πατήσει το speech button ξανά, για να δείξει στο σύστημα ότι έδωσε τη φωνητική εντολή τότε, ο client στέλνει σήμα στο server να σταματήσει την αναγνώριση και τοποθετεί στη στοίβα, των προς εκτέλεση εντολών, όλο το κείμενο που πείρε από τον sonic-server.

Τον sonic-server για να τον “σηκώσουμε” πρέπει να του δώσουμε δύο ορίσματα. Το ένα είναι το port στο οποίο “ακούει” τον client και το άλλο είναι ένα configuration αρχείο με το οποίο καθορίζουμε το setup με το οποίο θέλουμε να δουλεύει ο αναγνωριστής (ακουστικά μοντέλα, γλωσσικά μοντέλα, φωνήματα, λεξικό προφοράς κ.α.). Το configuration αρχείο που χρησιμοποίησα για την υλοποίηση του αναγνωριστή σε live-mode μορφή είναι:

Configuration File

```
-filler_file configuration/wsj.filler
-phone_config configuration/phoneset.cfg
-langmod_file configuration/my_model.bin
-dictionary configuration/wsj-5k.lex
-acoustic_mod tmpdir/wsj.mod
-word_end_beam 65.0
-word_entry_beam 160.0
-state_beam 160.0
-lm_scale 30.0
-rescore_lm_scale 30.0
-word_trans_penalty -12.5
-state_dur_scale 2.0
-short_word_penalty -10.0
-sample_rate 16000.0
-lm_garbage_collect 1
-max_active_states 12000
-auto_end_point 1
-end_point_padding 125
-max_word_ends 175
-filler_penalty 0.0
-live_mode 1
-vad_min_silence_frames 35
-vad_min_speech_frames 35
-push_to_talk 1
```

Τα ακουστικά μοντέλα που χρησιμοποίησα είναι αυτά του wsj (ένα παράδειγμα χρήσης του Sonic το οποίο κατασκευάστηκε από δεδομένα της Wall Street Journal). Τα γλωσσικά μοντέλα που χρησιμοποίησα τα δημιούργησα διαβάζοντας από κάτω προς τα πάνω τη γραμματική, που έχω δημιουργήσει για το Dialog Manager , δημιουργώντας έτσι όλες τις πιθανές φράσεις που δέχεται το σύστημα, που είναι άλλωστε και αυτές που με ενδιαφέρει ο αναγνωριστής να αναγνωρίζει.

Σαν λεξικό προφοράς χρησιμοποίησα αυτό του wsj, στο οποίο όμως τροποποίησα την προφορά αρκετών λέξεων ώστε να αναγνωρίζει λέξεις και με ελληνική προφορά αγγλικών, όπως επίσης πρόσθεσα και αρκετές λέξεις που σχετίζονταν με τη γραμματική που αντιλαμβάνεται το σύστημα.

4.7 Οι Διαλογικές Καταστάσεις του Συστήματος

Όπως σας ανέφερα προωτέρα (παράγραφος 4.4) το σύστημα αποτελείται από δύο ξεχωριστές εφαρμογές, η πρώτη είναι ένας communication agent που εκτελεί μια σειρά από εφαρμογές (κάνει ανάκτηση πληροφορίας από database, τηλεφωνεί, στέλνει e-mails, καταχωρεί δεδομένα στη βάση και κάνει περιπλάνηση στο διαδίκτυο) ενώ η δεύτερη είναι ένα παιχνίδι λέξεων. Η διαλογική υλοποίηση της πρώτης εφαρμογής (Agent Chimp) πραγματοποιείται με τη χρήση 19 διαφορετικών διαλογικών καταστάσεων, ενώ της δεύτερης (Agent Spell) μέσω μιας fsm. Οι διαλογικές καταστάσεις μεταξύ των οποίων “κινείται” ο Agent Chimp είναι οι εξής και χωρίζονται σε τέσσερις κύριες κατηγορίες :

Γενικές διαλογικές καταστάσεις

- Χαιρετισμός εισόδου
- Χαιρετισμός εξόδου
- Το σύστημα τίθεται σε αναμονή
- Ενεργοποίηση του συστήματος
- Κατάσταση Αμφισημίας

Διαλογικές καταστάσεις της εφαρμογής του τηλεφωνήματος

- Δεν είναι καταχωρημένο το άτομο στο οποίο θέλουμε να τηλεφωνήσει
- Τηλεφωνεί
- Το άτομο στο οποίο θέλουμε να τηλεφωνήσουμε δεν έχει καταχωρήσει τηλέφωνο
- Υπάρχουν περισσότερα του ενός άτομα με το ίδιο όνομα στο οποίο ζητάμε να τηλεφωνήσει
- Το όνομα στο οποίο ζητάμε να τηλεφωνήσει έχει περισσότερα από ένα τηλέφωνα.

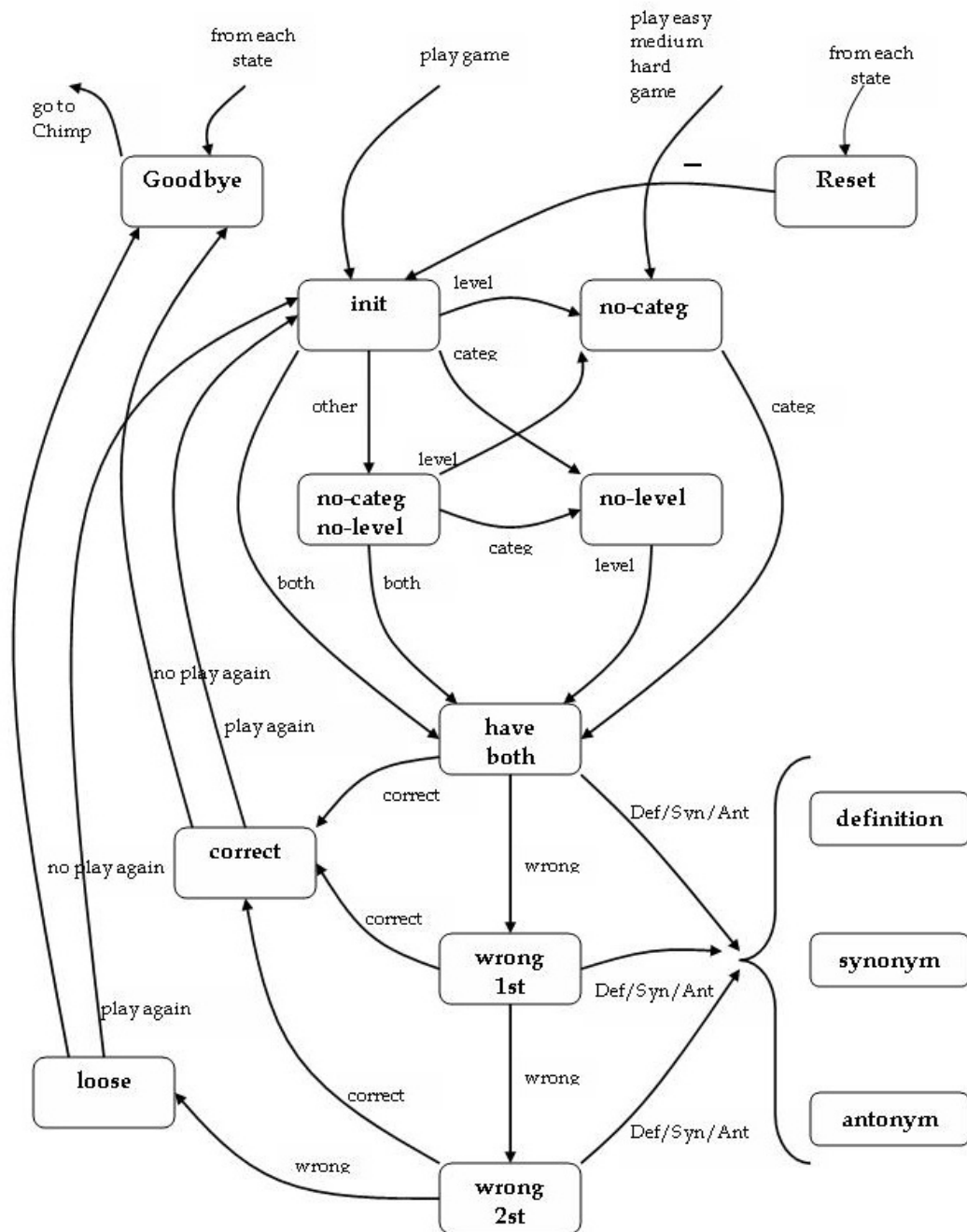
Διαλογικές καταστάσεις που σχετίζονται με ανάκτηση πληροφορίας

- Παρουσίαση συγκεκριμένης πληροφορίας για ένα άτομο.
- Παρουσίαση γενικής πληροφορίας για ένα άτομο.
- Αμφισημία μιας και υπάρχουν άτομα με το ίδιο όνομα για ανάκτηση συγκεκριμένης πληροφορίας.
- Αμφισημία μιας και υπάρχουν άτομα με το ίδιο όνομα για ανάκτηση γενικής πληροφορίας.
- Δεν είναι καταχωρημένο το άτομο για το οποίο ζητάμε πληροφορία.
- Το άτομο για το οποίο ζητάμε πληροφορία δεν έχει πληροφορία.

Διαλογικές καταστάσεις που σχετίζονται με εκτέλεση εφαρμογών

- Αποστολή e-mail
- Περιπλάνηση στο διαδίκτυο
- Παιχνίδι

Η διαλογική υλοποίηση του Agent Spell γίνεται με την fsm που ακολουθεί.

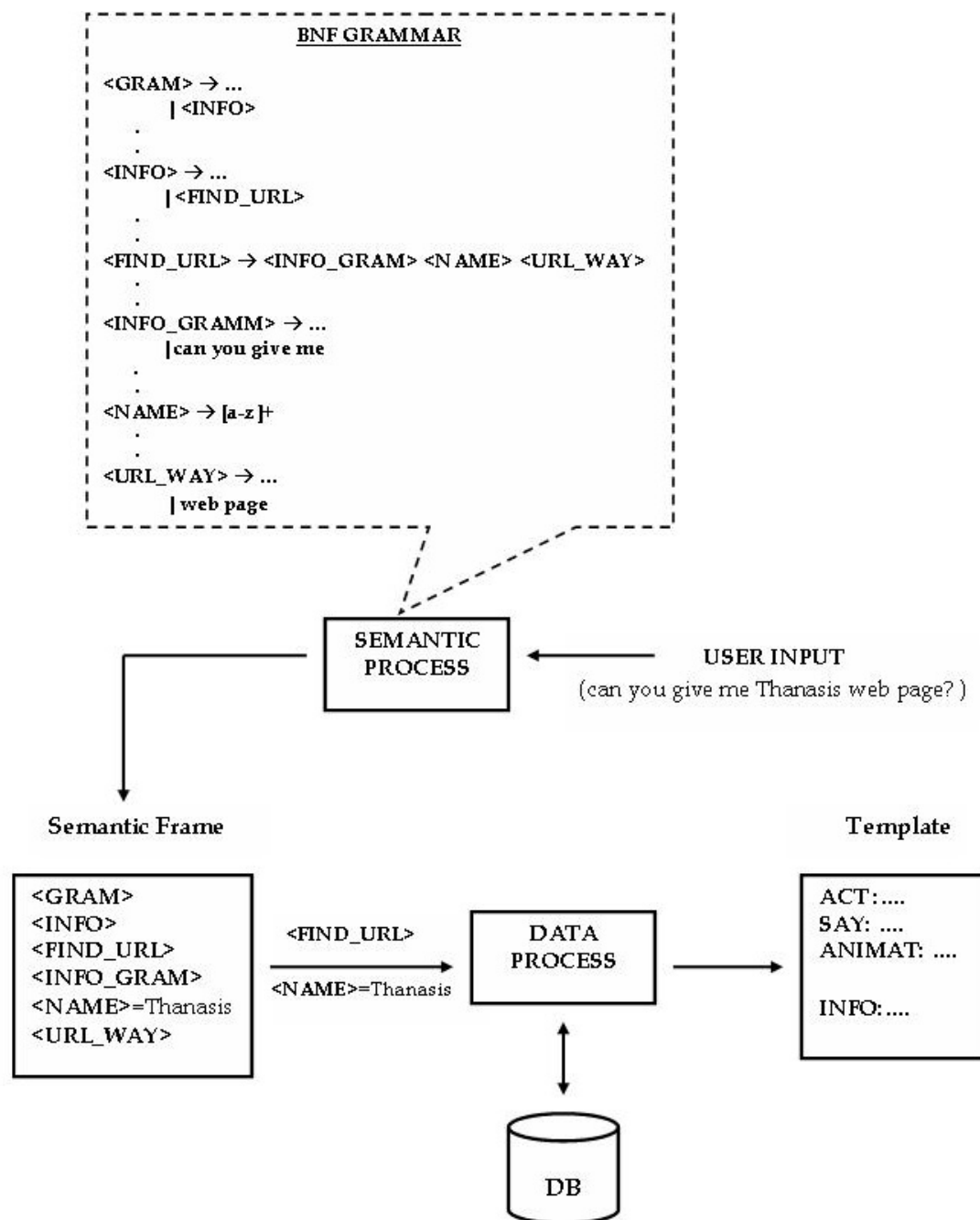


Γράφημα. 7. Η Διαλογική υλοποίηση του Agent Spell με τη χρήση μιας fsm

4.8 Dialog Manager

Αν η καρδιά του συστήματός μας είναι ο controller, τότε ο dialog manager αποτελεί το μυαλό, μιας και είναι αυτός που ορίζει τη στρατηγική και την ενέργεια στην οποία θα προβεί κάθε φορά το σύστημα.

Η δομή του dialog manager είναι αυτή που βλέπεται στο Γράφημα 8, σύμφωνα με την οποία, λαμβάνοντας το κείμενο της εισόδου του χρήστη το υποβάλλει σε σημασιολογικό έλεγχο για να κάνει εξόρυξη πληροφορίας. Στη συνέχεια ανάλογα με το είδος της πληροφορίας που εξορύσσεται (η ενέργεια που του ζητά να εκτελέσει ο χρήστης), ο dialog manager ανατρέχει στη βάση για να πάρει την ανάλογη ως προς την εντολή του χρήστη πληροφορία.



Γράφημα. 8. Το Διάγραμμα Ροής του Dialog manager

Έπειτα αφού έχει συλλέξει και έχει επεξεργαστεί όλη την απαραίτητη πληροφορία αποφασίζει σε πια διαλογική κατάσταση θα μεταβεί το σύστημα, τι θα απαντήσει στο χρήστη και πιο animation θα παίξει. Τέλος, δημιουργεί το template, το οποίο και μεταβιβάζει στον output event handler.

Ας προχωρήσουμε τώρα στην αρχιτεκτονική του dialog manager. Ο οποίος, αρχικά παίρνει την είσοδο του χρήστη και την υποβάλει σε σημασιολογική ανάλυση με τη χρήση μιας γραμματικής BNF χωρίς συμφραζόμενα, η δομή της γραμματικής BNF χωρίς συμφραζόμενα καθώς και η γραμματική που χρησιμοποιήσα υπάρχουν στο παράρτημα [1],[2]. Ο dialog manager έχοντας την είσοδο του χρήστη, κάνει Top-Down διάσχιση της γραμματικής μέχρι να κάνει match, κρατώντας σε ένα Semantic Frame όπως φαίνεται στο Γράφημα 8, όλα τα tokens** του κάθε κανόνα τον οποίο διασχίζει. Εν συνεχεία, μόλις κάνει match τη φράση του χρήστη, παίρνει από το Semantic Frame το token εκείνου του κανόνα που του φανερώνει την πρόθεση του χρήστη καθώς και πληροφορία που δίνει ο χρήστης στο σύστημα και η οποία σχετίζεται με την πρόθεση του. Πριν προχωρήσω να σας αναφέρω πως ο dialog manager χρησιμοποιεί 20 διαφορετικά σημασιολογικά tokens, από το σύνολο των tokens που περιέχει η γραμματική, για να αντιληφθεί την πρόθεση του χρήστη. Έπειτα, αφού πάρει το σημασιολογικό token και την πληροφορία του χρήστη, επεξεργάζεται τα δεδομένα ανάλογα με την πρόθεση στην οποία αναφέρεται το σημασιολογικό token (ο dialog manager έχει 20 διαφορετικές hardcoded επεξεργασίες, μία για κάθε σημασιολογικό token).

Η γενική ιδέα της επεξεργασίας είναι η εξής, ο dialog manager προσπελαύνει τη βάση σύμφωνα με τα δεδομένα που του έχει δώσει η σημασιολογική ανάλυση της εισόδου του χρήστη και ανάλογα με την πρόθεση του χρήστη ανακτά συγκεκριμένη πληροφορία. Σαν παράδειγμα παρατηρήστε το Γράφημα 8, όπου ο χρήστης ζητά την ηλεκτρονική διεύθυνση ενός ατόμου. Υποβάλλοντας σε σημασιολογική ανάλυση τη φράση αυτή συμπεραίνει ο dialog manager ότι χρήστης ζητά να του παρουσιαστεί πληροφορία για την ηλεκτρονική διεύθυνση ενός ατόμου. Στη συνέχεια ο dialog manager προσπελαύνει τη βάση για να βρει το άτομο του οποίου ο χρήστης ζήτησε πληροφορία και αφού το βρει παίρνει την πληροφορία για την ηλεκτρονική διεύθυνση. Τέλος αφού επεξεργαστεί τα δεδομένα σύμφωνα τον τρόπο που σας παρουσία προηγουμένως αποφασίζει (ανάλογα με το αποτέλεσμα της επεξεργασίας) για το ποια διαλογική κατάσταση θα εκτελέσει το σύστημα, τι θα απαντήσει στο χρήστη και πιο βίντεο θα παίξει. Δημιουργεί ένα template* στο οποίο βάζει πληροφορία, για την ενέργεια που θα εκτελέσει το σύστημα, την απάντηση, το βίντεο και την πληροφορία που ζήτησε ο χρήστης και το στέλνει στον output event handler.

*template: είναι ένα μέσο μεταφοράς πληροφορίας από ένα στοιχείο του συστήματος σε ένα άλλο, στο οποίο η πληροφορία έχει μια συγκεκριμένη δομή.

**tokens: είναι τα μη τερματικά σύμβολα

Η αρχιτεκτονική που σας παρουσίασα αφορά τον τρόπο λειτουργίας του dialog manager όταν στο σύστημα τρέχει ο communication agent, όταν στο σύστημα τρέχει η εφαρμογή του παιχνιδιού ο τρόπος λειτουργίας του dialog manager αλλάζει λίγο, αλλά η κύρια φιλοσοφία λειτουργίας παραμένει η ίδια.

Πιο συγκεκριμένα, οι δύο εφαρμογές δεν μπορεί να εκτελούνται ταυτόχρονα, αλλά είναι ξεχωριστές. Έτσι η γραμματική που χρησιμοποιείται στον communication agent για να κατανοήσει ο dialog manager την πρόθεση του χρήστη δεν χρησιμοποιείται στην εφαρμογή του παιχνιδιού. Στην οποία ο dialog manager χρησιμοποιεί για το λόγο αυτό όπως σας ανέφερα και προηγουμένως μια fsm, Γράφημα 7, καθώς και ένα Frame στο οποίο κρατάει πληροφορίες για την fsm (επίπεδο, κατηγορία, λέξη που δίνει ο χρήστης, κατάσταση, σωστές/λάθος προσπάθειες και κρυφή λέξη), Γράφημα 9.

Έτσι όταν στο σύστημα τρέχει η εφαρμογή του παιχνιδιού, ο dialog manager παίρνει την είσοδο του χρήστη και εξετάζοντας το Frame της κατάστασης στην οποία βρίσκεται, αποφασίζει σε πια κατάσταση θα μεταβεί, ανανεώνοντας παράλληλα τα δεδομένα του Frame.

FRAME

Level: easy
Category: general
User spell: laptop
Sate no: 4
Correct no: 2
Try no: 7
Origin spell: desktop

Γράφημα. 9. Το πλαίσιο στο οποίο κρατά πληροφορία το σύστημα που σχετίζεται με το παιχνίδι λέξεων

Η υλοποίηση και του dialog manager έγινε σε γλώσσα Perl, μιας και η Perl είναι αρκετά εύχρηστη στη διαχείριση κειμένων (string manipulation).

4.9 Αποτίμηση του Συστήματος

Για να αποτιμήσουμε το σύστημα καλέσαμε 15 άτομα να το αξιολογήσουν. Η αποτίμηση του συστήματος έγινε με βάση ποιοτικά χαρακτηριστικά του συστήματος. Δυστυχώς δεν συμμετείχαν παιδιά στη διαδικασία της αποτίμησης και ο λόγος είναι αφενός μεν ότι η γλώσσα που χρησιμοποιούμε στο σύστημα είναι η αγγλική, αφετέρου δεν είχαμε στη διάθεσή μας

ακουστικά μοντέλα για παιδιά για να τα χρησιμοποιήσουμε στον αναγνωριστή φωνής.

Οι χρήστες αφού πρώτα διάβασαν ένα σύντομο εγχειρίδιο χρήσης του συστήματος χρησιμοποίησαν το σύστημα καθοδηγούμενη από ένα σενάριο. Το σενάριο ήταν έτσι σχεδιασμένο ώστε οι χρήστες να δουν όλο το εύρος των εφαρμογών και δυνατοτήτων του συστήματος. Εν συνεχεία κλήθηκαν να συμπληρώσουν ένα ερωτηματολόγιο. Το εγχειρίδιο χρήσης, το σενάριο, το ερωτηματολόγιο καθώς και όλα τα στοιχεία βάση των οποίων έγινε η αποτίμηση παρατίθενται στο παράρτημα [3],[4],[5],[6].

Αξιολογώντας τα αποτελέσματα που προέκυψαν από την αποτίμηση βλέπουμε ότι το σύστημα απέσπασε γενικά θετικές κριτικές. Πιο συγκεκριμένα όπως βλέπεται και στο Γράφημα 10, οι χρήστες αξιολόγησαν κατά 80% θετικά το σύστημα ενώ το 20% το χαρακτήρισε ως μέτριο.

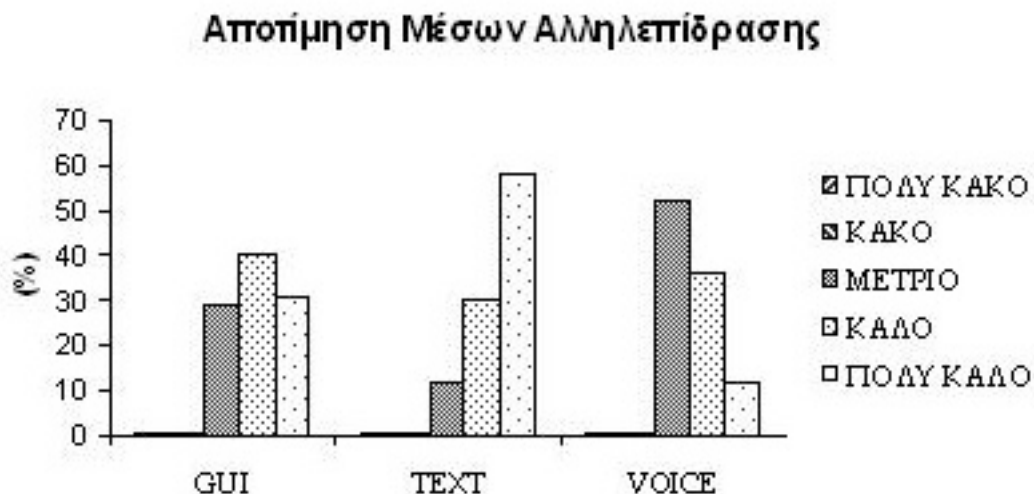


Γράφημα. 10. Τα αποτελέσματα της αποτίμησης στην ερώτηση: “Γενικά το Σύστημα είναι Καλό”

Στους χρήστες άρεσε ο τρόπος με τον οποίο τους παρουσιαζόταν η πληροφορία. Επίσης ορισμένοι εντυπωσιάστηκαν από τη φωνητική αλληλεπίδραση με το σύστημα, αν και όπως θα δούμε παρακάτω δεν ήταν η πλειονότητα. Κάτι άλλο που άρεσε στους χρήστες ήταν η δυνατότητα του συστήματος να μπορεί να εκτελεί την ίδια ενέργεια με πολλαπλούς τρόπους. Πολλά θετικά σχόλια απέσπασε το interface του συστήματος καθώς και η απλότητα επικοινωνίας με το σύστημα.

Η κριτική των χρηστών είχε κυρίως να κάνει με τα χαμηλά τις, πλείστες φορές, επίπεδα αναγνώρισης της ομιλίας κατά τη διάρκεια της φωνητικής αλληλεπίδρασής τους με το σύστημα. Ένα στοιχείο που επισημάνθηκε αρνητικά από τους χρήστες ήταν ο χρόνος απόκρισης του συστήματος και ειδικά για κάποιες εφαρμογές. Τέλος στα αρνητικά στοιχεία που επισημάναν οι χρήστες ήταν και το μικρό εύρος λειτουργιών που προσφέρει το σύστημα.

Όσο αναφορά τον τρόπο αλληλεπίδρασης των χρηστών με το σύστημα όπως βλέπουμε και στο Γράφημα 11, οι χρήστες βρήκαν πιο αποδοτικό τρόπο να επικοινωνούν με το σύστημα μέσω κειμένου και GUI παρά με τη χρήση της φωνής.



Γράφημα. 11. Τα αποτελέσματα της αποτίμησης στην ερώτηση: “Σου Άρεσε η Επικοινωνία με το Σύστημα χρησιμοποιώντας GUI/Κείμενο/Φωνή σαν Μέσο Επικοινωνίας”

Κάτι βέβαια που οφείλεται στα χαμηλά επίπεδα αναγνώρισης της ομιλίας. Σε περιπτώσεις χρηστών όμως όπου ο αναγνωριστής παρείχε καλά επίπεδα αναγνώρισης παρατηρούνταν τα αντίθετα αποτελέσματα, δηλαδή ο χρήστης να χρησιμοποιεί κατά κόρον τη φωνητική αλληλεπίδραση. Αυτή η αντίθεση πιθανός να οφείλεται στην προφορά των χρηστών, μιας και τα ακουστικά μοντέλα που χρησιμοποιήσαμε είχαν δημιουργηθεί από μη έλληνες.

Κάποια άλλα στοιχεία που προέκυψαν από την αποτίμηση είναι ότι οι χρήστες μπορούσαν εύκολα να πάρουν την πληροφορία που ήθελαν από το σύστημα. Όπως επίσης, ότι ήταν αρκετά εύκολο να κατανοήσουν το σύστημα και ήξεραν τι θα έπρεπε να πούν ή να κάνουν σε κάθε σημείο του διαλόγου μαζί του. Τέλος στην ερώτηση αν θα το ξαναχρησιμοποιούσαν το 93% απάντησε θετικά ενώ μόλις το 7% απάντησε ότι ίσως να το ξαναχρησιμοποιούσε.

Τέλος, στην ερώτηση τι επιπλέον θα θέλατε να προσθέσετε στο σύστημα η πλειονότητα θα ήθελε το σύστημα να αποκτήσει περισσότερα παιχνίδια καθώς και πιο καλή αντιληπτική ικανότητα όσον αναφορά το διάλογο με το χρήστη.

Κεφάλαιο 5

Τρόποι Βελτίωσης και Πιθανής Μελλοντικής Επέκτασης του Συστήματος

Παρακάτω θα σας παρουσιάσω συνοπτικά κάποιες πιθανές βελτιώσεις και προσθήκες που μπορούν να γίνουν στο σύστημα ώστε να το καταστήσουν πιο αποδοτικό.

Βελτίωση της Απόδοσης του Αναγνωριστή

Όπως είδατε στα αποτελέσματα της αποτίμησης του συστήματος η κύρια αιτία αρνητικών σχολίων σχετιζόταν με τα χαμηλά επίπεδα αναγνώρισης της φωνής. Κάτι που οφείλεται στα ακουστικά μοντέλα που χρησιμοποιούμε στον αναγνωριστή. Πρέπει λοιπόν να δημιουργήσουμε ακουστικά μοντέλα από έλληνες ούτως ώστε ο αναγνωριστής να μπορεί να αναγνωρίσει την αγγλική γλώσσα με την προφορά που έχει ένας έλληνας. Και πιο συγκεκριμένα να δημιουργηθούν ακουστικά μοντέλα από παιδιά μιας και το σύστημα απευθύνεται σε παιδιά.

Ένα άλλο στοιχείο για να αυξήσουμε την ικανότητα του αναγνωριστή είναι να δημιουργήσουμε ένα καλύτερο γλωσσικό μοντέλο για τον αναγνωριστή.

Μείωση του Χρόνου Απόκρισης του Συστήματος

Το δεύτερο σημείο που επισήμαναν οι χρήστες ότι δεν τους ικανοποίησε στο σύστημα ήταν ο χρόνος απόκρισής του. Πρέπει λοιπόν να βελτιώσουμε το χρόνο απόκρισης και αυτό μπορεί να επιτευχθεί με δύο τρόπους. Είτε να συγχρονίσουμε και να βελτιώσουμε τη λειτουργία και τη δομή αντίστοιχα του controller, κάτι που δεν είναι και η προτιμότερη λύση μιας και η αρχιτεκτονική του υπάρχοντος controller είναι λίγο απλοϊκή. Είτε να ξανασχεδιάσουμε τη δομή και αρχιτεκτονική του controller από την αρχή.

Προσθήκη Επιπλέον Λειτουργικότητας

Η προσθήκη επιπλέον λειτουργικότητας θα έκανε το σύστημα πιο ενδιαφέρον άρα και πιο δελεαστικό για ένα παιδί. Είναι κάτι άλλωστε που επισήμαναν και αρκετοί χρήστες που συμμετείχαν στην αποτίμηση του συστήματος. Πιο συγκεκριμένα η επιπλέον λειτουργικότητα του συστήματος έχει να κάνει με την προσθήκη παιχνιδιών και με την αναβάθμιση του ήδη υπάρχοντος (π.χ. να υπάρξει βαθμολογία ανάλογα με τη δυσκολία της λέξης με την οποία παίζει ο χρήστης και να κρατιέται κάποια λίστα με τα high score του κάθε παίκτη, στοιχεία που έλκουν τα παιδιά μιας και δημιουργεί συναγωνισμό). Όπως επίσης μπορεί να προστεθεί στο σύστημα ένας paint editor για να ζωγραφίζουν τα παιδιά μιας και όπως ξέρουμε είναι μια ασχολία που αρέσει αρκετά στα παιδιά.

Βελτίωση της Διαδραστικότητας μεταξύ Χρήστη και Συστήματος

Η βελτίωση της διαδραστικότητας θα κάνει το σύστημα πιο εύχρηστο, άρα και πιο αποδοτικό, μιας και οι χρήστες θα μπορούν εύκολα να συνδιαλέγονται μεταξύ του. Κάτι που μπορεί να επιτευχθεί αυξάνοντας την αντιληπτική ικανότητα του συστήματος.

Για να αυξηθεί η αντιληπτική ικανότητα του συστήματος, άρα το σύστημα να αντιλαμβάνεται τι του ζητά κάθε φορά ο χρήστης, πρέπει να βελτιώσουμε και να εμπλουτίσουμε τη γραμματική που χρησιμοποιεί το σύστημα διαλόγου του συστήματος (dialog manager).

Επίσης, να βελτιωθεί το σύστημα ώστε να μπορεί να παίρνει πρωτοβουλίες διαλογικής στρατηγικής ανάλογα με τη διαλογική κατάσταση στην οποία βρίσκεται και να μην περιμένει παθητικά το χρήστη να του δώσει εντολές.

Ένα άλλο σημείο που θα βελτιώνει τη διαδραστικότητα μεταξύ χρήστη και συστήματος, είναι η χρησιμοποίηση διαφορετικών animation και μηνυμάτων απόκρισης ακόμη και για ίδιες λειτουργίες μιας και η επανάληψη κουράζει το χρήστη.

Τέλος, η χρήση διαλόγων, από μέρους του συστήματος, που θα εμπεριέχουν και προσωπικά στοιχεία για τον χρήστη (π.χ. όνομα, μια φωτογραφία του χρήστη, κάτι που του αρέσει κ.τ.λ) θα έκανε το σύστημα πιο ενδιαφέρον και πιο εντυπωσιακό για το χρήστη.

Παράρτημα

1. Γραμματικές Χωρίς Συμφραζόμενα

Για τις γραμματικές χωρίς συμφραζόμενα έχουν αναπτυχθεί διάφοροι τρόποι παράστασης. Οι συμβολισμοί αυτοί περιγράφουν μόνο τους συντακτικούς κανόνες αλλά κάνουν ορισμένες παραδοχές που επιτρέπουν τη διάκριση των τερματικών από τα μη τερματικά σύμβολα. Επιπλέον, το αρχικό σύμβολο είναι πάντοτε το μη τερματικό σύμβολο του πρώτου συντακτικού κανόνα. Γενικά, οι συμβολισμοί αυτοί επιτρέπουν μια παράσταση των γραμματικών που είναι συντομότερη και πιο εύχρηστη από ότι έχουμε χρησιμοποιήσει μέχρι τώρα. Γι' αυτό χρησιμοποιούνται συχνά στα εγχειρίδια ορισμού γλωσσών προγραμματισμού.

Ένας συμβολισμός για γραμματικές που χρησιμοποιήθηκε πρώτη φορά για την περιγραφή της σύνταξης της γλώσσας προγραμματισμού ALGOL 60 και έγινε αργότερα πολύ δημοφιλής, είναι η *μορφή Backus Naur* (Backus Naur Form - BNF). Σύμφωνα με αυτό το συμβολισμό:

- Τα μη τερματικά σύμβολα περικλείονται σε γωνιακές παρενθέσεις, π.χ. $\langle \text{expr} \rangle$.
- Το σύμβολο \rightarrow στους κανόνες παραγωγής αντικαθίσταται από το σύμβολο $::=$.
- Σε έναν κανόνα παραγωγής μπορούν να συνδυαστούν τα δεξιά μέλη περισσότερων κανόνων παραγωγής για το ίδιο μη τερματικό σύμβολο, χωρισμένα με το σύμβολο $|$. Για παράδειγμα, ο κανόνας παραγωγής $\langle \text{digit} \rangle ::= 011$ είναι ισοδύναμος με τους δυο κανόνες παραγωγής $\langle \text{digit} \rangle ::= 0$ και $\langle \text{digit} \rangle ::= 1$.

Ο συμβολισμός που περιγράφηκε είναι ο καθαρός συμβολισμός BNF. Στην πράξη χρησιμοποιούνται διάφορες παραλλαγές του. Μια τέτοια παραλλαγή είναι ότι αντί του συμβόλου $::=$ χρησιμοποιείται το \rightarrow . Μια άλλη παραλλαγή είναι ότι τα τερματικά σύμβολα γράφονται με μικρά γράμματα και τα μη τερματικά σύμβολα γράφονται με κεφαλαία γράμματα.

2. Η Γραμματική του Dialog Manager σε μορφή Backus Naur

<GRAM>-> <GREETING>

- | <GOODBYE>
- | <SLEEP>
- | <WAKE>
- | <CALL>
- | <URL>
- | <MAIL>
- | <INFO>
- | <GAME>

<GREETING>-> <HI>

- | <HI> <CHIMP>

<CHIMP>-> agent chimp

- | chimp

<HI>-> hi

- | hello

<GOODBYE>-> <WORD_GOODBYE>

- | <WORD_GOODBYE> <CHIMP>

<WORD_GOODBYE>-> goodbye

- | bye bye
- | bye
- | adios
- | exit

<SLEEP>-> <SLEEP_WORD>

- | <SLEEP_WAY> <SLEEP_WORD>
- | <SLEEP_WAY> <SLEEP_WORD> <CHIMP>
- | <SLEEP_WORD> <CHIMP>

<SLEEP_WORD>-> sleep

<SLEEP_WAY>-> go to

<WAKE>-> <WAKE_WAY>

- | <WAKE_WAY> <CHIMP>

<WAKE_WAY>-> wake up

<CALL>-> <CALL_PERSON_AT_PLACE>
| <CALL_PERSON>

<CALL_PERSON_AT_PLACE>-> <WORD_CALL> <NAME> at <NAME>

<WORD_CALL>-> call
| phone
| telephone

<CALL_PERSON>-> <WORD_CALL> <NAME>

<NAME>-> (.*)

<URL>-> <PERSON_URL>
| <URL_ADDR>

<PERSON_URL>-> <URL_GRAM> <NAME> <URL_WAY>

<URL_GRAM>-> go to
| take me to
| open

<URL_WAY>-> url
| web page
| site

<URL_ADDR>-> <URL_GRAM> <WWW>

<WWW>-> www\.(.*)

<MAIL>-> <SEND_BY_MAIL>
| <SEND_BY_NAME>

<SEND_BY_MAIL>-> <MAIL_GRAM> <MAIL_ADDR>

<SEND_BY_NAME>-> <MAIL_GRAM> <NAME>

<MAIL_GRAM>-> send <MAIL_WORDS>
| <MAIL_WORDS>

<MAIL_WORDS>-> email to
| mail to
| email
| mail

<MAIL_ADDR>-> (.*)\@(.*)

<INFO>-> <FIND_URL>
| <FIND_INFO>
| <FIND_EMAIL>
| <FIND_FAX>
| <FIND_PAGER>

<FIND_URL>-> <INFO_GRAM> <NAME> <URL_WAY>

<INFO_GRAM>-> can you find me
| what is
| get me
| find me
| can you give me
| give me
| tell me

<FIND_INFO>-> <INFO_GRAM> <NAME> info
| <INFO_GRAM> information for <NAME>
| <INFO_GRAM> information about <NAME>

<FIND_EMAIL>-> <INFO_GRAM> <NAME> <EMAIL_WAY>

<EMAIL_WAY>-> email
| email address
| mail
| e-mail

<FIND_FAX>-> <INFO_GRAM> <NAME> <FAX_WAY>

<FAX_WAY>-> <FAX_WORD>
| <FAX_WORD> number

<FAX_WORD>-> fax

<FIND_PAGER>-> <INFO_GRAM> <NAME> <PAGER_WAY>

<PAGER_WAY>-> <PAGER_WORD>
| <PAGER_WORD> number

<PAGER_WORD>-> pager

<GAME>-> <PLAY_GAME>
| <PLAY_LEV_GAME>

<PLAY_GAME>-> <WANT> a new game

- | <WANT> a game
- | play a game
- | <WANT> the spelling bee
- | <WANT> the spelling game
- | play game
- | the game
- | play

<WANT>-> i want to play
| i want

<PLAY_LEV_GAME>-> <WANT> <LEVEL> game
| <WANT> a new <LEVEL> game
| <WANT> <LEVEL> spelling bee
| <WANT> <LEVEL> spelling game
| play <LEVEL> game

<LEVEL>-> <LEV_EASY>
| <LEV_MEDIUM>
| <LEV_HARD>

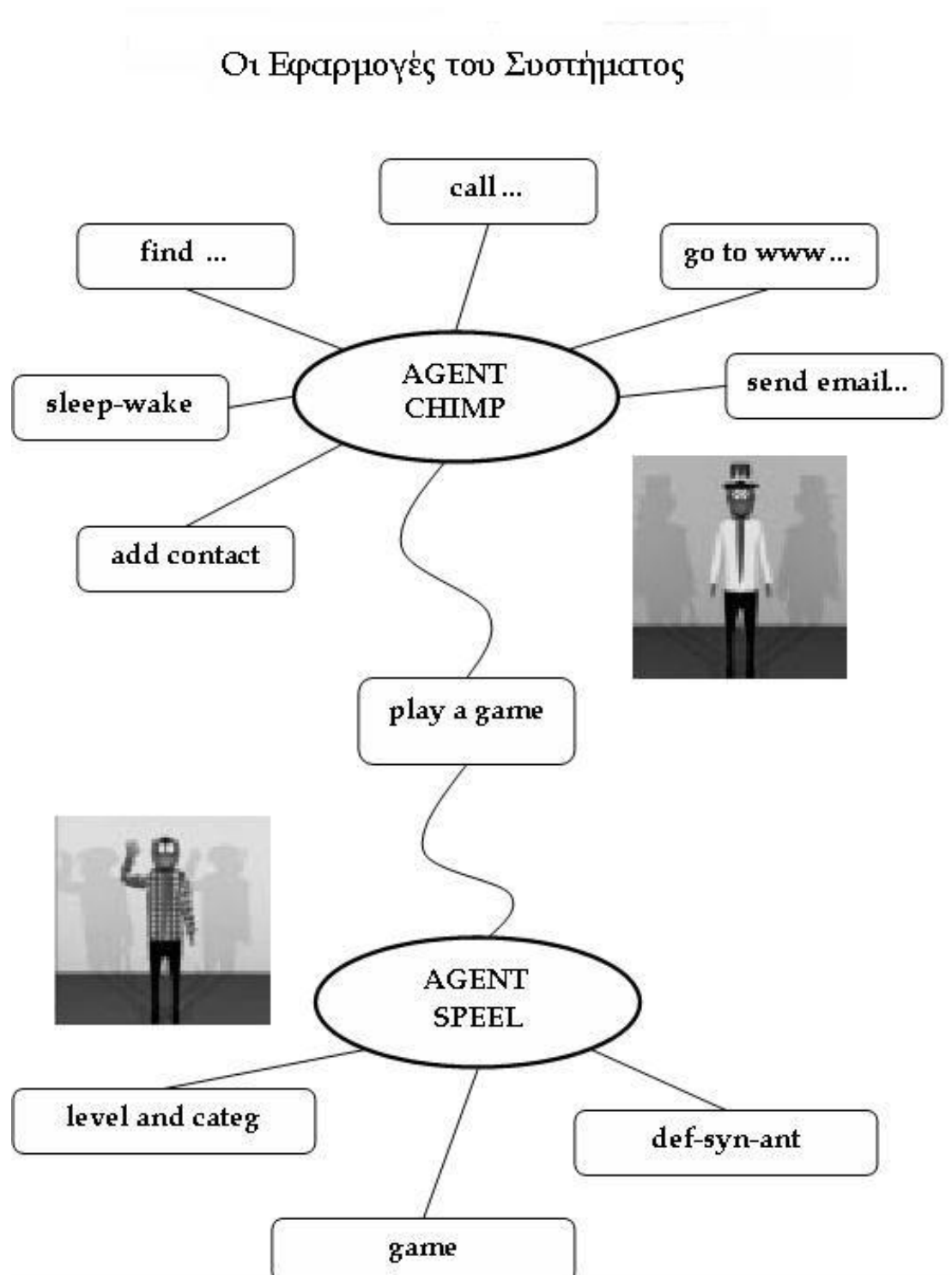
<LEV_EASY>-> easy

<LEV_MEDIUM>-> medium

<LEV_HARD>-> hard

3. Το Εγχειρίδιο Χρήσης του Συστήματος (Στα Αγγλικά)

The system that you are going to evaluate is an application for children. You can retrieve information about a person (name, telephones, url, email, fax, pager) and you can also play a computer game .



The system consisted of two distinct tasks as shown in Figure above: a communications agent application (insert/retrieve data for a person, placing phone-calls, accessing the Internet, sending email) and a computer game (find the missing word). Both applications were controlled by a conversational animated agent. The agent embodiment chosen for the system was a cartoon chimpanzee character. The personality and appearance of the agent, however, were different across the two applications.

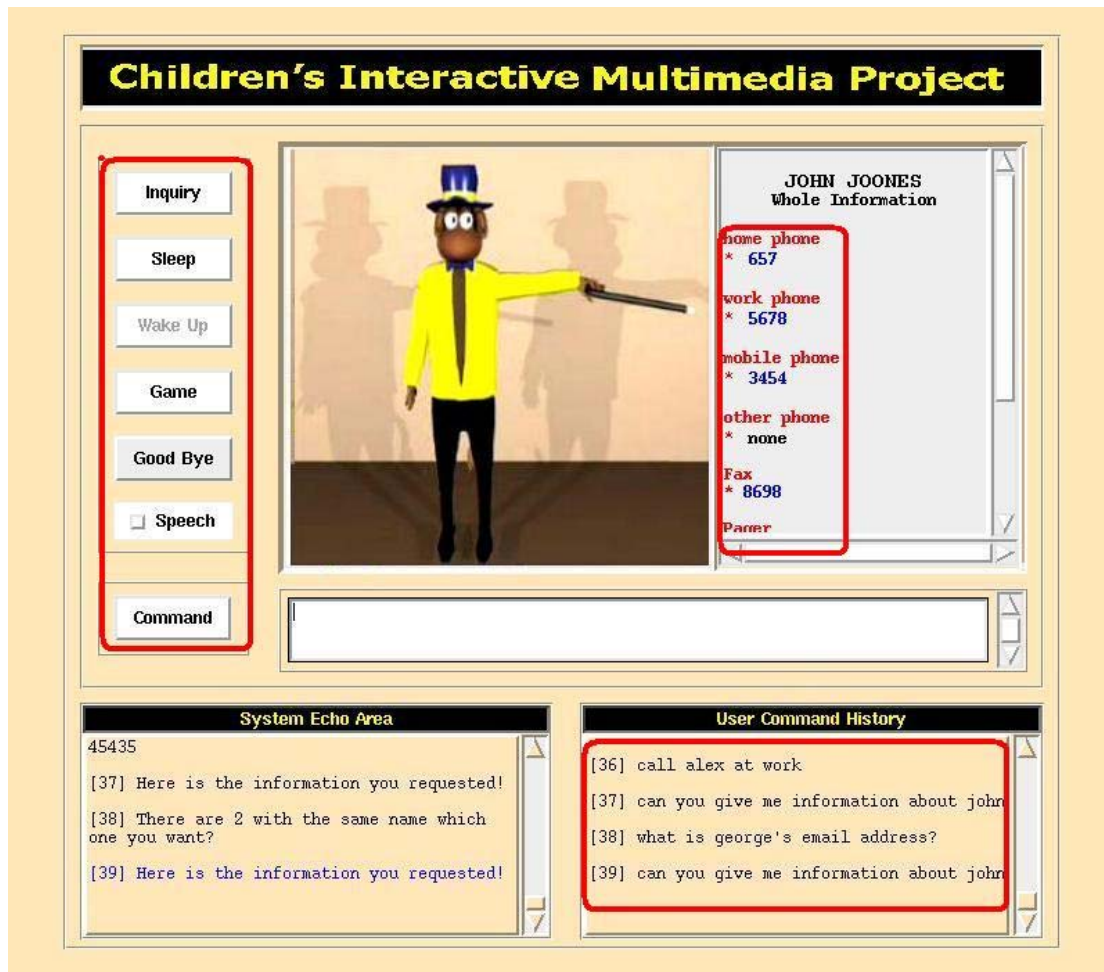
The Agent Chimp, handles the applications of system, while the Agent Spell is handles the computer game. The user could freely switch back and forth between the applications at any point during the interaction.

The user interface enables interactions using voice and typed text in natural language, mouse clicks or combinations there of. Output to the user is presented through audio, graphics, animation and textual modalities.

Using The System

You can give your commands in the system by typing them in the command line field and then press enter or press the “command” button on GUI, to execute. You can also execute a command by clicking on the appropriate buttons of GUI or click on a previous command that you have given (on the User Command History field) or on a data that the system presents you (all the data that are executable when you move the mouse on them become highlighted and are the data which contained in the fields below). For example in the figure below, if you click on a phone number the system will make a phone-call to the corresponding number.

Another way to execute a command is with voice. You press first the “speech” button and then the system recognize your voice. When you have given your command you should press the “speech” button again to execute the system your command. During the process that the system recognize your voice, on the speech button is a red light.



Below I present you the applications that each agent are responsible for:

- AGENT CHIMP

You can **insert data** for a person by clicking on the “Inquiry” button and then fill the form on the pop-up window that will appear. You fill the form by clicking each time on the appropriate button and then write the data (e.g press name button and write on the empty field the name of the person that you want to insert). After you have written the data in the empty field you should press the “add” button and the data that you have given kept on the right column. After you have given all the data you should press the “OK” button to insert the data in the database. Then the system will inform you if the insertion has completed or not (the system makes a typical check on the data if are correct).

IMPORTANT : The data should contain at least the name of the person to add a new contact in the system.

You can **make a phone-call** by typing/said to the system to call the person (Jim or Jim Green or Jim Green at home/mobile/work).

You can accessing the internet with two way by said/typing to the system to go to the persons web page or to go to the url address that you want.

You can send an e-mail with two way by said/typing to the system to send email to a person or send email to the email-address that you want.

You can find information about a person. You can ask for general information about a person (e.g give me info for a person). You can ask for specific information (phones, email, url, fax, pager) about a person (e.g can you find me person's email).

You can also greet the system, idle the system by telling it to go to sleep, reactivate the system by telling it to wake up and goodbye the system when you want to finish.

- AGENT SPELL

Agent Spell is responsible to navigates you to play a words game. To play the game you should ask it from Agent Chimp. Either telling him to play a game (you can also specify the level of the game, e.g. I want to play an easy game), or clicking on the "Game" button. Then the agent changes to Agent Spell.

The aim of game is to find the missing word that the system each time defines, for that purpose you can ask from Agent Spell to give you the definition, synonym and the antonym of the word that you should find. You can ask for the definition, synonym and antonym many times, either with GUI or with commands(typed/voice). You have three chances to find the missing word.

When you are on Agent Spell you can restart the game by clicking "Restart" button or typed/said it.

How you play the game now. You should first determine the level (easy, medium, hard) and the category (general, science, history) of the word that you want to play with. You can do that either by clicking on "Option" button or said/typed it.

During the game the system presents you some statistics and information about the game in the right column of GUI.

When you want to finish game application you click on "Goodbye" button or said/typed it.

4. Το Σενάριο της Αποτίμησης του συστήματος (Στα Αγγλικά)

Do two times the below scenario. Using only text with gui buttons the first time and voice with gui buttons the second time.

- Greet the system.
- Insert data for yourself
- Tell the system to call Green and then select from the list James Green and call him at work
- Ask information about Jack and then select from the list Jack Brown and go to his personal site
- Open the site that you have previous inserted when you inserted data in the system about yourself
- Open the site: www.tuc.gr
- Send an e-mail to your self
- Send an e-mail to Jack and then from the list select Jack Green to send him an e-mail
- Tell to Chimp to go to sleep
- Wake up him
- Execute a previous command from the User Command History field
- See information for yourself
- Play 2 games
- Exit

5. Το Ερωτηματολόγιο της Αποτίμησης του Συστήματος

Questionnaire

Questionnaire no:

Please rate with a tick the following questions

Strongly Disagree =1

Somewhat Disagree =2

Neither Agree nor Disagree =3

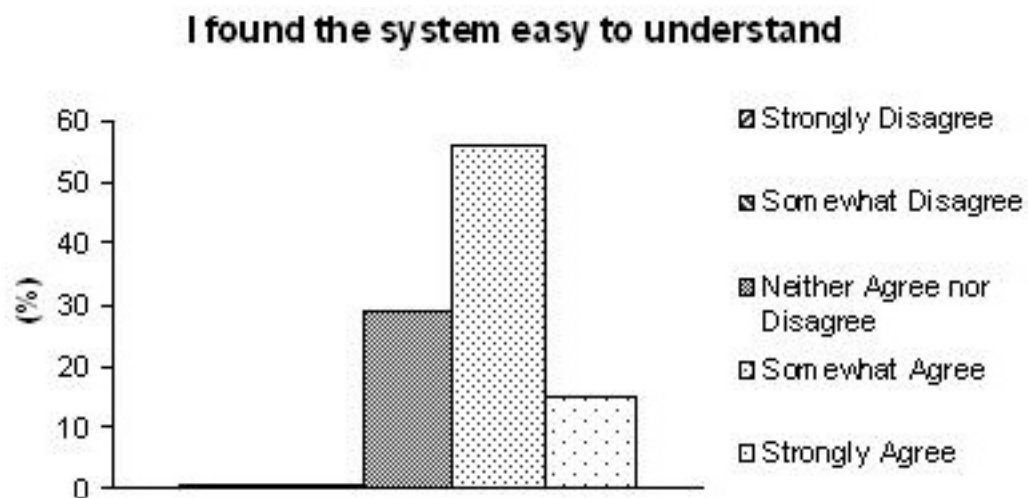
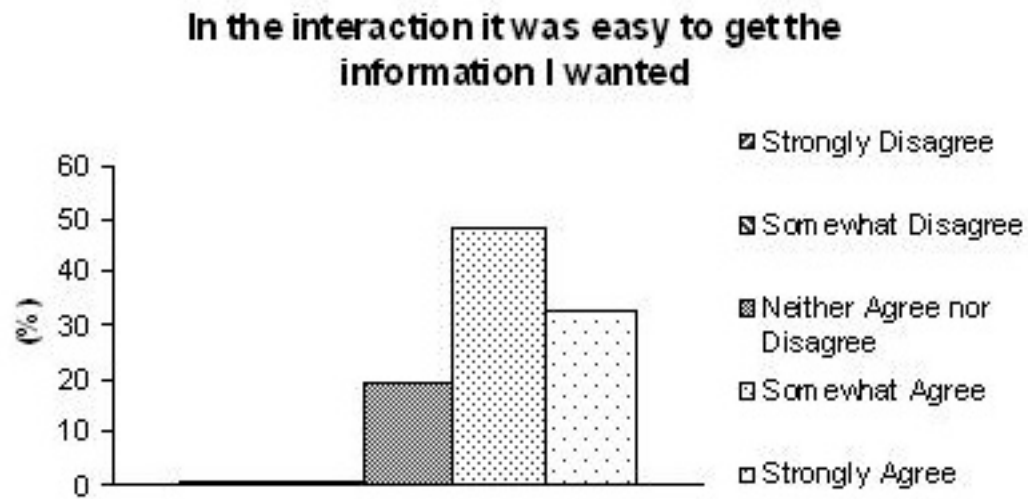
Somewhat Agree =4

Strongly Agree =5

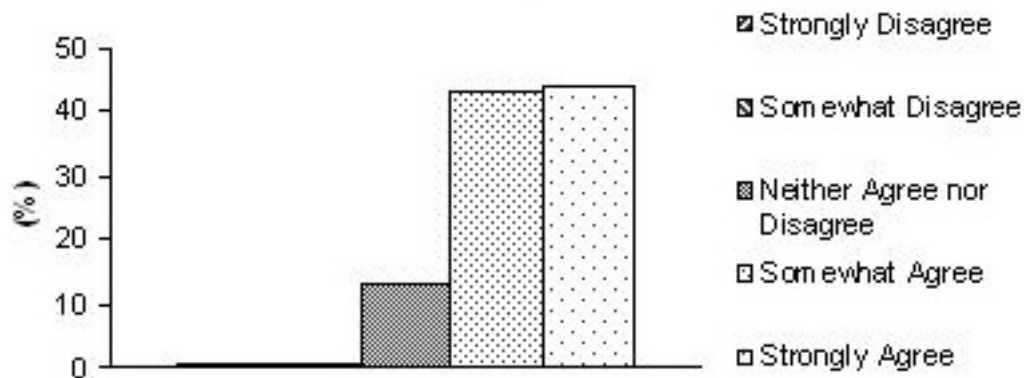
Questions	Rate				
	1	2	3	4	5
In the interaction it was easy to get the information I wanted					
I found the system easy to understand					
I knew what I could say or do at each point of the dialogue					
System response time was short					
Based on my experience in this conversation using this system I would like to use this system again					
The system worked the way I expected it to in this interaction					
Overall the system is good					
I like the Graphic User Interface (GUI) of system					
I like the Text Interface of system					
I like the Voice Interface of system					

- If you rated any of the above questions as “strongly disagree” please explain here why.
- What was that, which you like on the system?
- What was that, which you didn’t like on the system?
- What else would you like the system to contain?
- Other observations

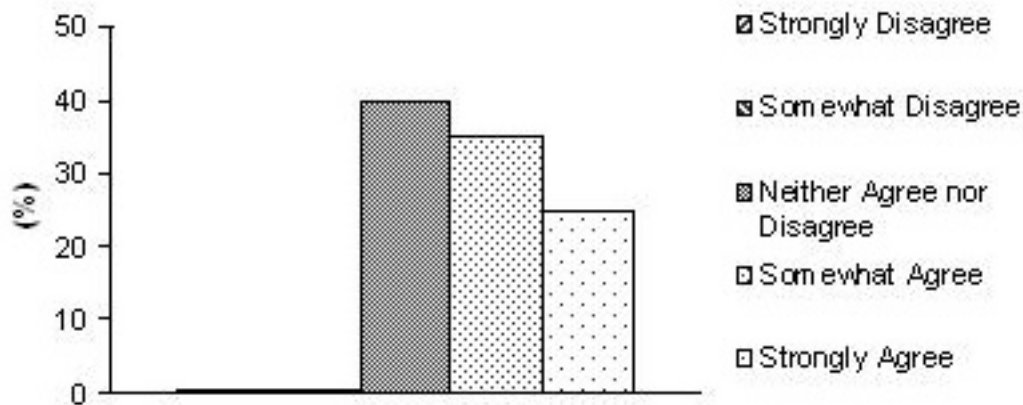
6. Τα Αποτελέσματα της Αποτίμησης του Συστήματος



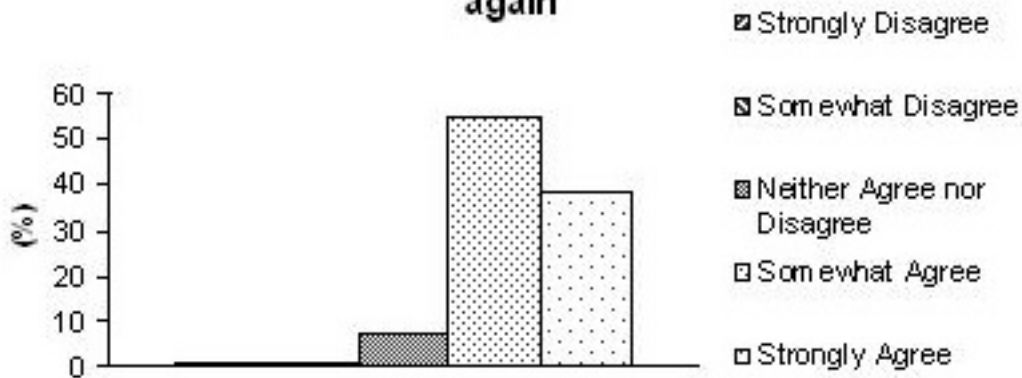
I knew what I could say or do at each point of the dialogue



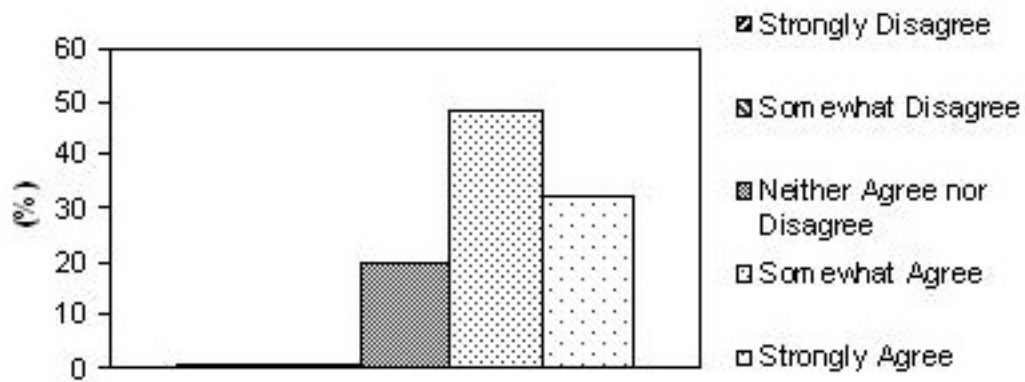
System response time was short



Based on my experience in this conversation using this system I would like to use this system again



The system worked the way I expected it to in this interaction



Βιβλιογραφία

- [1] Ben Shneiderman, The future of interactive systems and the emergence of direct manipulation. Behavior and Information Technology 1 (3), 1982, 237-256
- [2] Jim Miller, Intelligent software agents vs. user-controlled direct manipulation: a debate. Human Factors in Computing Systems. CHI '97 Extended Abstracts, ACM Press, 1997,105-106.
- [3] William Buxton and Brad A. Myers, A study in two-handed input. Human Factors in Computing Systems, CHI '86 Conference Proceedings. ACM Press, 1986,321-326.
- [4] Stephane Chatty, Extending a graphical toolkit for two-handed interaction. ACM VIST '94 Symposium on User Interface Software and Technology, ACM Press, 1994,195-204.
- [5] Joelle Coutaz, Laurence Nigay, and Daniel Salber, The MSM framework: a design space for multi-sensori-motor systems. Proceedings of EWHCI '93, Third East-West International Conference on Human-Computer Interaction, Lecture Notes in Computer Science, 753, Springer-Verlag, 1993, 231-241.
- [6] James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes, Computer Graphics: principles and practice, 2nd edition. Addison-Wesley, 1990.
- [7] Morton L. Heilig, El cine del future (The cinema of the future). Espacios, 23-24, Apartado Postal Num. 20449, Espacios SA, Mexico, 1955.
- [8] L. Schomaker, J. Nijtmans, A. Camurri, F. Lavagetto, P. Morasso, C. Benoit, T. Guiard-Marigny, B. Le Goff, J. Robert-Ribes, A. Adjoudani, I. Defee, S. Munch, K. Hartung, and J. Blauert, A Taxonomy of Multimodal Interaction in the Human Information Processing System. A Report of the Esprit Basic Research Action 8579 MIAMI. February, 1995. <http://vonkje.cogsci.kun.nl/~rnianii/reports/reports.html> (checked 15 Nov 99)

- [9] Richard A. Bolt, Put-that-there. SIGGRAPH '80 Conference Proceedings, ACM Press, 1980, 262-270
- [10] Richard A. Bolt and Edward Herranz, Two-handed gesture in multi-modal natural dialog. ACM VIST '92 Symposium on User Interface Software and Technology, ACM Press, 1992. 7-14.
- [11] Philip R. Cohen, Mary Dalrymple. Douglas B. Moran. Fernando C. N. Pereira, Joseph W. Sullivan, Robert A. Gargan Jr, Jon L. Schlossberg, and Sherman W. Tyier, Synergistic use of direct manipulation and natural language. Human Factors in Computing Systems, CHI '89 Conference roceedings, ACM Press, 1989, 227-233.
- [12] Wolfgang Wahlster, User and discourse models for multimodal communication. In intelligent User Interfaces. ACM Press, 1991, 45-67
- [13] J. G. Neal, C. Y. Thielman, Z. Dobes, S. M. Haller, and S. C. Shapiro, Natural language with integrated deictic and graphic gestures. Proceedings of the 1989 DARPA Workshop on Speech and Natural Language, Morgan Kaufmann, 1989,410-423.
- [14] Philip R. Cohen, Michael Johnston, David McGee, Sharon Oviatt, Jay Pittman, Ira Smith, Liang Chen, and Josh Clow, QuickSet: multimodal interaction for distributed applications. Proceedings of the Multimedia '97, Fifth ACM International Conference on Multimedia. ACM Press, 1997,31-40.
- [15] David Weimer and S. K. Ganapathy, A synthetic virtual environment with hand gesturing and voice input. CHI '89, Conference Proceedings, ACM Press, 1989, 235-240
- [16] Sharon Oviatt, Multimodal intefaces for dynamic interactive maps. Human Factors in Computing Systems, CM '96 Conference Proceedings, ACM Press, 1996,95-102
- [17] Sharon Oviatt, Multimodal interactive maps: designing for human performance. Human-Computer Interaction, 12,1997,93-129.
- [18] Jie Yang, Rainer Stiefelhages, Uwe Meier, and Alex Waibel, Visual tracking for multimodal human computer interaction. CHI '98, Conference Proceedings, ACM Press, 1999, 140-145

- [19] Mark T. Maybury and Wolfgang Wahlster (Eds.), Readings in Intelligent User Interfaces. Morgan Kaufmann Publishers, 1998
- [20] A. Potamianos *et al.*, Design principles and tools for multimodal dialog Systems. In Proc. ESCA Workshop on Interactive Dialogue in Multi-Modal Systems, Kloster Irsee, Germany, June 1999
- [21] S. Seneff et al, Galaxy-II: A reference architecture for conversational system development. In Proc. ICSLP, Sydney, Australia, Dec. 1998.
- [22] L. Hirschman et al, Multi-site data collection and evaluation in spoken language understanding. In Proc. of the Human Language Technology Workshop, Mar. 1993.
- [23] A. Pargellis, H.-K. Kuo, and C.-H. Lee, Automatic application generator matches user expectations to system capabilities. In Proc. ESCA Workshop Interact. Dialog. Multi-Modal Syst., Kloster Irsee, Germany, June 1999.
- [24] A. Pargellis, H.-K. Kuo, and C.-H. Lee, Automatic dialogue generator creates user defined applications. In Proc. EUROSPEECH, Budapest, Hungary, Sept. 1999.
- [25] H.-K. Kuo, A. Pargellis, and C.-H. Lee, Information and services manager customizes dialogue-based applications. In Proc. ESCA Workshop Interact. Dialog. Multi- Modal Syst., Kloster Irsee, Germany, June 1999.
- [26] M. Walker, D. Litman, C. Kamm, and A. Abella, Evaluating spoken dialogue agents with paradise: Two case studies. Computer Speech and Language, pp. 317-347, 1998.
- [27] A. Potamianos and S. Narayanan, (2002). Creating conversational interfaces for children. IEEE Transaction on Speech and Audio Processing, vol. 10, no. 2, February 2002

