University of Crete
Department of Computer Science

FO.R.T.H.
Institute of Computer Science

Development Of Interactive User Interfaces For Voice Training Aimed To Children With Hearing Loss Using Web Technologies In Real Time

(MSc. Thesis)

Myron Apostolakis

Heraklion
September 2014

Department of Computer Science
University of Crete

Development Of Interactive User Interfaces For Voice Training Aimed To Children With Hearing Loss Using Web Technologies In Real Time

Submitted to the Department of Computer Science
in partial fulfillment of the requirements for the degree of Master of Science

September 29, 2014

Author:
Myron Apostolakis
Department of Computer Science

Committee

Supervisor
Yannis Stylianou
Associate Professor, Thesis Supervisor

Member
Yannis Tzitzikas
Assistant Professor

Member
Athanasios Mouctaris
Assistant Professor

Accepted by:

Chairman of the
Graduate Studies Committee
Antonis Argiros
Professor

Heraklion, September 2014

Abstract

International research and global statistics have shown that 1.5% of children up to the age of 20 years have reduced auditory ability, 1 in 22 children of school age have impaired hearing, which means that in nowadays in Europe there are about one million hearing-impaired children, while in the U.S. 12,000 children born annually or 33 children a day with hearing loss. In Greece, statistically hard of hearing children are estimated at about 80,000. These data classify the hearing loss first among the diseases of newborns. It is often, people with hearing loss to have problems with their communication skills. Because of the lack of audio feedback the speech production system is not developed normally. Since deaf people cannot hear themselves speak, they cannot tune their voices to a more 'correct' sounding tone. More generally they cannot control their speech production system (tongue, teeth etc) properly, because they cannot realize which is the right way to do it. As a result they speak too loud for the vowels or they are misarticulating consonants. However, a person who went deaf later on in life, has a better chance of being able to speak more properly. So, everything is a matter of feedback.

The purpose of this thesis is to introduce a new approach of speech therapy multimedia tools based on the state of art web technologies and taking into account the special characteristics of hearing impaired people, in order to help them acquire better communication skills. This approach is taking advantage of special speech properties such as intensity, pitch and spectrograms using them as visual feedback, in order to teach a person with hearing loss how to improve control of his voice.

More specifically we developed a web site platform, where the user can login and practice with a collection of web-based voice games, through browser in real time. The technologies which were used for the implementation of our games is Java, Javascript, HTML5, CSS3 and frameworks like Apache Shiro and Hibernate. The database which is used is MySQL and XAMPP as web server. Voice is analyzed and converted to visual feedback. Each game could be played with a logo-therapy supervisor or even by user himself. Score of each game, is calculated and is sent to our web server for saving and statistic processing. In the end, user performance in the passage of time is displayed through graphs in real time. A logo-therapy supervisor could use these special graphs to spot possible weaknesses and propose modification of game targets as necessary. Furthermore, the evaluation of our platform is performed by specialists in speech therapy. Finally, comparison between state of art technologies (HTML5, JavaScript) and older, such as Java, in terms of flexibility and performance is taking place.

Περίληψη

Διεθνείς έρευνες και παγκόσμιες στατιστικές μετρήσεις έχουν δείξει ότι 1,5 % των παιδιών μέχρι την ηλικία των 20 ετών έχουν μειωμένη ακουστική ικανότητα ενώ 1 σε 22 παιδιά σχολικής ηλικίας έχουν προβλήματα ακοής. Το γεγονός αυτό φανερώνει ότι σήμερα στην Ευρώπη υπάρχουν περίπου ένα εκατομμύριο παιδιά με προβλήματα ακοής, ενώ στις ΗΠΑ 12.000 παιδιά γεννιούνται ετησίως με απώλεια ακοής. Στην Ελλάδα τα βαρήκοα παιδιά υπολογίζονται σε περίπου 80.000. Τα στοιχεία αυτά κατατάσσουν την απώλεια ακοής στην πρώτη θέση μεταξύ των ασθενειών των νεογνών. Είναι συχνό φαινόμενο, τα άτομα με απώλεια ακοής να έχουν προβλήματα σε επικοινωνιακό επίπεδο. Λόγω της έλλειψης της ηχητικής ανατροφοδότησης του εγκεφάλου των παιδιών, το σύστημα παραγωγής ομιλίας τους δεν αναπτύσσεται κανονικά. Δεδομένου ότι τα κωφά άτομα δεν μπορούν να ακούσουν την ομιλία τους, δεν μπορούν να συντονίσουν τις φωνές τους σε ένα πιο «σωστό» ήχο. Στην πραγματικότητα αδυνατούν να ελέγξουν τα όργανα παραγωγής λόγου (γλώσσα, δόντια κλπ.) σωστά, επειδή δεν μπορούν να συνειδητοποιήσουν ποιος είναι ο σωστός τρόπος για να το κάνουν. Ως εκ τούτου μιλούν πολύ δυνατά για τα φωνήεντα ή παράγουν λάθος τα σύμφωνα. Ωστόσο, ένα πρόσωπο που έχασε την ακοή του σε μεγαλύτερη ηλικία, έχει μεγαλύτερη πιθανότητα να μιλήσει πιο σωστά. Έτσι καταλήγουμε στο γενικότερο συμπέρασμα ότι τα πάντα είναι θέμα ανατροφοδότησης.

Ο σκοπός αυτής της διατριβής είναι να εισάγει μια νέα προσέγγιση των εργαλείων λογοθεραπείας με βάση την χρήση πολυμεσικών διαδικτυακών τεχνολογιών λαμβάνοντας υπόψη τα ιδιαίτερα χαρακτηριστικά των ατόμων με προβλήματα ακοής, ώστε να αποκτήσουν καλύτερες δεξιότητες επικοινωνίας. Η παρούσα προσέγγιση αξιοποιεί τα ακουστικά χαρακτηριστικά του λόγου, όπως την ένταση, το ύψος και τα σπεκτρογράμματα χρησιμοποιώντας τα ως οπτική ανατροφοδότηση, προκειμένου να διδάξει ένα άτομο με απώλεια ακοής πώς να βελτιώσει τον έλεγχο της φωνής του.

Πιο συγκεκριμένα έχουμε αναπτύξει κατάλληλο διαδικτυακό χώρο, όπου ο χρήστης μπορεί να συνδεθεί και να εξασκηθεί με τη συλλογή από διαδικτυακά παιχνίδια του λόγου. Οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση των παιχνιδιών είναι η Java, Javascript, HTML5, CSS3 και frameworks όπως το Apache Shiro και το Hibernate. Η βάση δεδομένων που χρησιμοποιήθηκε είναι η MySQL και ως διαδικτυακός εξυπηρετητής ο XAMPP. Τα παιχνίδια αυτά εκτελούνται μέσω του προγράμματος φυλλομετρητή και αλληλεπιδρούν με τον χρήστη αναλύοντας μια ξεχωριστή ιδιότητα της φωνής του σε πραγματικό χρόνο. Κάθε παιχνίδι θα μπορούσε να εκτελεστεί υπό την εποπτεία μιας ομάδας λογοθεραπευτών ή ακόμα και από το χρήστη τον ίδιο από οποιαδήποτε τοποθεσία. Οι βαθμολογίες κάθε παιχνιδιού, υπολογίζονται και αποστέλλονται στο διαδικτυακό εξυπηρετητή μας για να αποθηκευτούν και να επεξεργαστούν στατιστικά. Στη συνέχεια, η απόδοση των χρηστών στο πέρασμα του χρόνου εμφανίζεται σε πραγματικό χρόνο μέσω γραφημάτων. Οι επόπτες λογοθεραπευτές, θα μπορούσαν να χρησιμοποιήσουν αυτά τα ειδικά γραφήματα για να εντοπίσουν πιθανές αδυναμίες και να τροποποιήσουν τους στόχους του παιχνιδιού καταλλήλως με απώτερο στόχο την ακόμα μεγαλύτερη βελτίωση του χρήστη. Ακόμη η συλλογή των παιχνιδιών μας παρουσιάστηκε και αξιολογήθηκε από έμπειρους χρήστες αντίστοιχου λογισμικού (λογοθεραπευτές). Τέλος, πραγματοποιούμε σύγκριση των τεχνολογιών αιχμής (HTML5, JavaScript) οι οποίες χρησιμοποιήθηκαν κατά τη διάρκεια της ανάπτυξης όμοιων παιχνιδιών της παρούσας εργασίας και παλιότερων, όπως η Java, όσον αφορά την ευελιξία τους και την απόδοση τους στην παρούσα χρονική στιγμή.

Ευχαριστίες

Η διατριβή αυτή αποτελεί το τελικό στάδιο της προσπάθειας δύο περίπου ετών για την απόκτηση του μεταπτυχιακού διπλώματος ειδίκευσης στην Επιστήμη Υπολογιστών και το καταστάλαγμα της εμπειρίας που απέκτησα κατά την διάρκεια της συνεργασίας μου από την φοίτηση μου στο τμήμα Επιστήμης Υπολογιστών του πανεπιστημίου Κρήτης.

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα της εργασίας μου, καθηγητή Ιωάννη Στυλιανού, για την εμπιστοσύνη και το ενδιαφέρον που μου έδειξε. Ακόμα ευχαριστώ θερμά την διδάκτορα Κουτσογιαννάκη Μαρία, τον διδάκτορα Καφεντζή Γιώργο, την διδάκτορα ερευνήτρια Άννα Σφακιανάκη καθώς και το λογοθεραπευτή Νίκο Βενιέρη για την πολύτιμη συμβολή τους στην ολοκλήρωση της εργασίας μου. Οι προτάσεις και συμβουλές τους υπήρξαν καθοριστικές. Τέλος, θα ήθελα να ευχαριστήσω την οικογένειά μου που με στερήσεις, στηρίζει τις προσπάθειές μου κατά τη διάρκεια των σπουδών μου.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

By default the term 'deaf' person refers to someone with a hearing loss. In addition to hearing loss, people with hearing loss face a whole series of second level issues such as: language choice, communication mode, self-perception and identity.

We could say that deafness and hearing loss are associated with the volume (intensity) of sound that an individual receives, and also to the pitch (frequency) of sound. Some individuals have particular problems with hearing high or low-pitched sounds. These patients have difficulty in hearing high-pitched or low-pitched voices and have implications for teaching and learning situations. These implications can not be faced using a hearing aid. Hearing aids usage is to increase the volume of sound but cannot compensate for loss of frequency.

There are many causes of deafness. Some people are born deaf due to a hereditary condition, or had congenital problems such as those associated with rubella [1, Rubella]. Also hearing loss could happen as a result of injury, illness or exposure to excessive noise. The type of deafness or hearing loss, and the time in life that it is developed, affects person's communication ability. Most deaf and hard of hearing people use a combination of communication methods (sign language, lip reading etc).

It is often people with hearing loss to have problems with their communication skills. Because of the lack of audio feedback the speech production system it is not developed normally. Since deaf people cannot hear themselves speak, they cannot tune their voices to a more 'correct' sounding tone. More generally they cannot control their speech production system (tongue, teeth etc) properly, because they cannot realize which is the right way to do it. As a result they speak too loud for the vowels or they are misarticulating consonants. However, a person who went deaf later on in life after an accident or something, has a better chance of being able to speak more properly. So, everything is a matter of feedback.

Many speech therapy multimedia tools have been developed to help people with hearing loss to acquire better communication skills with the rest of the people. Most of them are taking advantage of special characteristics of the sound and are using multimodal information as feedback, in order to teach a person with hearing loss the proper way of pronunciation. The type of multi-modal feedback could be a combination of all senses. Some examples of feedback are audio and visual feedback, tactile feedback, synthetic face, visualized acoustic properties, automatic feedback etc [2, Klara Vicsi].

1.2 Contribution

The purpose of this thesis is to develop a web site platform, where the user can login and practice with a collection of web-based voice games, through browser in real time. To that purpose three different technologies are used. In our web-based platform voice is analyzed and converted to visual feedback. Each game could be played with a logo-therapy supervisor or even by user himself. Score of each game, is calculated and is sent to our web server for saving and statistic processing. In the end, user performance in the passage of time is displayed through graphs in real time. Logo-therapy supervisors, could use these special graphs to spot possible weaknesses and modify game targets as necessary. Furthermore, an evaluation of our game collection is presented. The evaluation includes a questionnaire filled by a specialist in logo-therapy. Finally, comparison between state of art technologies (HTML5, JavaScript) and older, such as Java, in terms of flexibility and performance is taking place.

Bibliography research (second chapter) reveals that there is no other online real time speech therapy software tool. Every other tools are standalone commercial applications. The system that we developed is powered up from web service's benefits. It is available in every time at any place. Also it has no requirements of installation. Finally, unlike existing speech therapy software tools, has no payment requirements and it is available for use by anyone who has registered in our platform free of charge.

1.3 Structure of thesis

In the first chapter, we are describing the nature of hearing loss and how hard of hearing people experience this situation. This study is essential in order to understand the importance of feedback in voice training and the type of feedback that helps people with hearing loss. In the second chapter, a bibliography research has been done in order to give more detailed description of deafness and how deaf people hear. Furthermore, basic speech therapy techniques along with speech therapy software tools which are taking advantage of several types of feedback are being presented. In the third chapter, we are presenting our implementation approach and we are giving a brief description for each cooperating part of our proposed system. In the fourth chapter, analysis implementation is being presented. More specifically we are using UML diagrams to describe our system. In the fifth chapter, evaluation of our system is being presented. We evaluate our system with special questionnaires which were filled up by experienced speech therapists. In the sixth chapter we compare our system with other commercial tools. Finally in the seventh chapter possible extensions and future work of our system are presented.

Chapter 2

Bibliography research - how deaf people hear?

2.1 Classification of hearing loss

In this point it is important to classify the types of hearing loss. Also it is important to give some definitions of basic terms that are used in this field of science. More specifically,

- Deaf/Deafness refers to an individual who has a profound hearing loss and makes use of sign language.
- Hard of hearing refers to an individual with a hearing loss who relies on residual hearing and combines speaking with lip-reading.
- Hearing impaired term describes any deviation from normal hearing, permanent or transient whose levels range from from mild hearing loss to profound deafness.
- Residual hearing refers to the percentage of hearing remaining after hearing loss.

The level of severity of hearing loss, is defined as follows [3, Rafi Shemesh] :

Range (HL: Hearing Loss)	**Categorization**
-10 to 15 dB	Normal Hearing
16-25 dB	Slight Hearing Loss
26-40 dB	Mild Hearing Loss
41-55 dB	Moderate Hearing Loss
56-70 dB	Moderate-Severe Hearing Loss
71-90 dB	Severe Hearing Loss
>90 dB	Profound Hearing Loss

Table 2.1: Levels of Hearing Loss

Furthermore we can classify people with hearing loss in the following categories:

- The age at which a person loses his hearing has a very large impact on the individual. The earlier a child is diagnosed the better off the child will be. It is desirable that the diagnosis of hearing loss in children to take place at birth. Necessary precautions can be taken earlier if they are diagnosed early enough. However, it is common hearing loss occurs up to a year before diagnosis. In "Educating the deaf: Psychology, Principles, and Practices,

Moores" [4, Moores] tells us that every day that goes by that the child is not diagnosed is a day lost. Language development cannot begin until the child has the means to communicate. If a child can not hear, and not able to obtain a hearing aid, then the child will experience serious problems in the development of communication skills.

Language development depends highly on early identification of the hearing loss and is extremely important for the development of an individual. If the child has the capability to obtain hearing aids then sooner a child is fitted for the hearing aid, the sooner that child has access to sound. It is obvious that the earlier a child is diagnosed for hearing loss, the earlier the child can begin to learn.

Children who are hard of hearing have been not identified until the school years. Additionally, children who are hard of hearing are sometimes considered to be thought of as self-opinionated or obstinate. Usually, parents say that these kids have "selective hearing" or that they don't pay too much attention. They may perform poorly in school or decide that they "don't like" school [5, Marschark]. Ranking of hearing loss can be done in several ways. **Pre-Lingual or Post-Lingual** is one of them (Table:2.2). Pre-Lingual symbolizes that hearing loss occurred before language acquisition, typically that is before the age of 2 years. Post-Lingual symbolizes that hearing loss occurred after language acquisition.

Pre-Lingual hearing loss	**Post-Lingual hearing loss**
Time of hearing loss $<$ 2 years	Time of hearing loss $>=$ 2 years

Table 2.2: Prelingual and post-lingual hearing loss

There is an enormous difference in language skills of pre-lingual and post-lingual children with hearing losses. Children who have already acquired language before the onset of hearing loss have a much easier time learning than children who have not acquired language. The implications of both types of hearing losses are many.

It is most likely for a child with pre-lingual hearing loss to have a hard time learning language. Also academic achievement may be lower and social interaction may also be difficult. On the other hand, children with post-lingual hearing loss will probably be able to preserve most of the language learned. They socialize more easily and they have higher academic achievements, especially in reading. Another term for post-lingual hearing loss is also adventitious hearing loss.

Finally, presbycusis is the loss of hearing associated with increasing age. Hearing loss is ranked as the third most prevalent chronic disorder after hypertension and arthritis. Its prevalence and severity increase with age, rising from about 30-35 percent of adults aged 65 and older to an estimated 40-50 percent of adults aged 75 and older [6, Cruickshanks KJ]. For presbycusis the hearing loss is greater for high-pitched sounds and lower for low-pitched. For example, it may be difficult for someone to hear the sound of breaking glass, and it is most difficult to understand speech in a noisy background. However, the same person may be able to clearly hear the low-pitched sound of a basso. Presbycusis usually occurs equally in both ears. Finally because of the slow rate of development of presbycusis it is common for people who suffer from it not to realize it.

- The place where the loss occurs. Hearing loss can be conductive, sensorineural, or mixed. Conductive hearing losses are more easily treated by hearing

aids. Sensorineural hearing losses cannot be helped by amplification measures. Mixed hearing losses are both conductive and sensorineural hearing losses. Usually in this case hearing aids will treat only the conductive part of the hearing loss (Table:2.3).

Name	Description
Conductive	Characterized by an obstruction in the transmission of the audio signal through the external auditory canal and/or the middle ear. All frequencies are decreased equally
Sensorineural	Characterized by the malfunction of the sensory receptors . of the inner ear. Sensorineural deafness is a lack of sound perception caused by a defect in the cochlea and/or the auditory division of the vestibulocochlear nerve.
Mixed	Mixed hearing loss consists of both conductive and sensory dysfunction

Table 2.3: Types of hearing loss

If the cause of hearing loss is hereditary, then the parents are prepared for the possibility that children can develop problems with their hearing. In this case, the parents have more time to make appropriate movements to deal with the situation more successfully. On the other hand, if the parents are unprepared for the possibility of hearing loss it is most likely to lose valuable time to take the necessary steps and make the diagnosis of hearing loss. Parents who are aware of the signs of hearing loss are more likely to examine their children for hearing loss, while the parents who do not know is likely to confuse hearing loss with other problems such as learning disabilities or behavioral disorders [4, Moores].

2.2 Learning language techniques

In this section we are going to present the basic techniques of language learning that are widely used in deaf community. Lip reading technique, sign language, speech therapy exercises, special devices such as cochlear implants and speech buddies tools are representative techniques of language learning. Especially lip reading, sign language and cochlear implants are very common to deaf people and they are used in daily base depending of the level of their hearing loss. A brief description of each technique is provided.

2.2.1 Lip reading

Lip reading (or speech reading) is a technique of interpretation of lip movements, facial expressions, tongue and residual hearing in order for a person to understand speech, when there is no normal sound available. Lip reading also is relied on information provided by the context and knowledge of the language. Although lip reading is used primarily by deaf people, sometimes is used by people with normal hearing.

In everyday life, people subconsciously use lip reading to understand better the acoustic information and some speakers are able to read speech to some extent. This is explained because each phoneme corresponds to a specific facial expression and mouth, so someone can extract what phoneme has been spoken based only visual signs, even if the sound is insufficient or distorted.

Lip reading is limited because many phonemes share the same viseme and thus is impossible to identify only from visual signs. More specifically, for sounds whose place of articulation is deep inside the mouth or throat are not detectable, such

as glottal consonants and most gestures of the tongue. Also, voiced and unvoiced pairs look identical, such as [p] and [b], [k] and [g], [t] and [d], [f] and [v], and [s] and [z] [7, Lip reading] likewise for nasalization (e.g. [m] vs. [b]). It has been estimated that only 30% to 40% of sounds in the English language are distinguishable from sight alone. Thus, for example, the phrase "where there's life, there's hope" looks identical to "where's the lavender soap" in most English dialects.

As a result, a lip reader depends on cues from the environment, from the context of the communication, and knowledge of the topic of a conversation. For example common phrases such as greetings are much easier to read. However there are difficult scenarios where speech reading is quite difficult.

These scenarios include:

- Lack of clear picture of the speaker's lips. This includes:
  - obstructions such as moustaches or hands in front of the mouth
  - the speaker's head turned aside or away
  - dark environment
  - bright back-lighting source such as a window behind the speaker, darkening the face.
- Group discussions, especially when multiple people are talking in quick succession. The challenge here is to know where to look.

2.2.2 Use of lip reading by deaf people

Lip readers who have grown up deaf may never have heard the spoken language, which makes speech reading much more difficult. Also in order to learn the individual visemes they have to receive special education where basic educational procedure is conducted by conscious training. As a result, lip reading takes a lot of effort, and can be extremely tiring. For these and other reasons, many deaf people avoid to use lip reading in order to communicate with non-signers. They prefer to use other ways, such as mime and gesture, writing, and sign language interpreters.

To quote from Dorothy Clegg's 1953 book The Listening Eye [8, Dorothy Clegg], "When you are deaf you live inside a well-corked glass bottle. You see the entrancing outside world, but it does not reach you. After learning to lip read, you are still inside the bottle, but the cork has come out and the outside world slowly but surely comes in to you." This view that lip reading, though difficult, can be successful is relatively controversial within the deaf world.

It is a common practice to combine lip reading with movements of the hands in order to represent invisible details of speech. Using cued speech has the advantage of helping speaker to develop lip-reading skills that may be useful even when there are no other cues, i.e., in communication with non-deaf, non-hard of hearing people [7, Lip reading].

2.2.3 Sign language

Sign language is a kind of language which, in order to convey communication information, instead of using the traditional sound patterns and words is using body language and gestures. This may include simultaneously formation shapes with hands, facial expressions or body orientation in order to express a speaker's thoughts. In the other hand spoken language ("oral languages") depend primarily on sound. Sign languages and spoken languages have many features in common and that is why linguists consider the two languages to be natural languages, although they have significant differences.

Development of sign language exist where there are deaf people. People who can hear but cannot speak normally also use sign language. Sign languages are governed by the rules of grammar as well as natural languages. Moreover , they exhibit linguistic idioms like spoken languages. Around the world there are hundreds of sign languages used by communities of deaf people. Some of them are officially recognized by the state and others are not. A common misunderstanding is that sign languages are the same all over the world or that sign language is international. However, although there may be common features between sign languages, each country has its own native sign language.

Australian researchers have conducted investigations who reveal that both children with hearing impairment and children with normal hearing will learn sign languages if their parents use sign language, in the same way as other children learn spoken languages.

Researchers from the United States in the 1970's began to investigate the specific characteristics of sign language in learning in order to compare learning of spoken languages and learning of sign language. For example, many signs in sign languages are iconic. Symbols of sign language look like the meaning of the symbol. For example, in the symbol HOUSE, hands forms the shape of a roof and walls. This differentiates the sign languages of the spoken sounds where usually words have no relation to their meaning. One challenge for the researchers was to find out if the use of iconic signs made learning of sign language easier for children than learning spoken languages.

"From the age of approximately six months, children learning sign language begin to "babble" on their hands, making sign-like actions in imitation of the signed language they see around them".

Research has shown that children who learn sign language experience, the same stages of language development as children who learn spoken language. Learning sign language begins at birth and continues in their childhood.

Children who learn sign language from the age of six months are starting to "babble" with their hands mimicking the signs of sign language they see in their environment. In the first year of their life, they produce the first sign just like children learning spoken languages are saying their first word. [9, Adam Schembri].

With the passage of time the children are adding more and more signs in their vocabulary. Signs such as FATHER, MOTHER, DOG, GOODBYE etc. are typical for children of this age. Also, they make the same mistakes in sign production with incorrect gestures or movements like children who are learning spoken languages and are unable at first to pronounce all the sounds properly.

Shortly before the age of two years, children are starting to combine the signs creating proposals as Milk WANT FIND THE BALL. The vocabulary of children is growing rapidly and gradually they are capable to form larger and complex sentences.At the age of 2 and 2.5 years old, they learn to form negative sentences, ask questions. At about 5 years old, they already have acquired the largest part of the grammar and syntax of their vocabulary. After that, new vocabulary acquisition, continues always throughout life.

In the case of children who can hear and come from families where one parent is deaf and another speaks, they learn spoken language and sign language together. At early ages they do not show any preference between sign language and spoken language. This shows that for young children the language is treated the same way regardless of whether it is spoken language or signed language.

2.2.4 Simple techniques - Combination of senses

In this section we refer to simple techniques which are useful for teaching a person with hearing loss and several ways to control his organs of speech when

applying speech therapy. The information which is displayed is acquired by visiting several forums and web sites [10, 11, 12] where deaf community is exchanging opinions, common problems and several issues from their lives. The speech therapy techniques mentioned on this thesis are not fully analyzed but only a first approach is presented in order to understand the psychology of an individual with hearing problems and refer to possible solutions for their problem.

In speech therapy the biggest problem of articulation is the placement of the tongue in the oral cavity. The problem becomes more intense in the case of children with special needs and children with a cleft of the lip or palate. It is extremely difficult to teach the movements of the tongue in order to produce the desired sounds. More specifically, for the parents is extremely difficult to understand the correct tongue position required to produce various sounds. The solution to this problem is the continuous practice at home, performing exercises in order to learn correct placement of tongue.

Speech is a process that takes place subconsciously without counting each step separately in order to talk. It is an automatic action who someone have performed millions of times in his life without thought. But, what happens if you are a child? Children barely understand that your tongue moves at all in order to produce sound. Additionally, an adult is trying to change placement of your tongue and complete successfully a series of difficult exercises while you are just trying to get your apple juice! Under these circumstances it is reasonable for the children to grumble during learning language process. Here are a some easy tongue placement exercises:

1. If "La" sound is the problem, look at your child's mouth. Now look at your own and try to figure out how it is produced. The sound "la" is produced when pushing the tongue out in a way that can collide with the top lip. Placing of some chocolate on the top lip could help. As the child tries to reach the chocolate, this effort would enhance the desired movement of the tongue and production of the desired sound.

2. If the "S" sound is the problem, similar actions have to take place.At first try to produce "S" sound by yourself. In order to create the "S" sound, you have to push air out past your tongue with your teeth together while you are pulling the corners of your mouth back. Try to teach the child to do the same. A nice way to teach this it is "overacting" a sound. Using funny faces during the process could convert tongue placement exercises to fun, not punishment.

3. If the "T" sound is the problem similar actions will help you. The sound "t" is produced when you trap air between the edge of your tongue and the back of your top front teeth. When the air is released quickly, then "t" sound is produced. What will your child have to do in this case? One good way is to get a child to push it's tongue up behind the teeth and hold it there is to place the straw coming out of a milk shake right behind the teeth. One sip, one practice sound. Some other sounds like "D" are produced the same way ("D" is produced from behind the front teeth, as does the "th" sound).

4. "Rrrrr" is a another common articulation problem for children. In order to create the "r" sound, your tongue is held up without touching the palate permanently. There are several ways to fix this problem. An easy one is to allow the child to "growl" and then growl into a word.

In the same philosophy with the above exercises some special speech tools are being developed called Speech Buddies. The purpose of these tools is to teach a child the right position of the tongue in it's mouth in order to provide very specific

tactile cues. One of these tools called the 'R speech Buddy' tool allows the child to feel exactly what he needs to do with his tongue in order to produce a correct /r/ sound. Children are very good tactile learners, especially in primary school. The R Speech Buddy helps to unlock a sense of feeling in order for the children to learn the correct tongue movement. The way it works is actually simple. Two simple steps are involved, placement and movement. For every difficult sound to pronounce different tools have been developed (R, S, CH, SH, and L sounds).

Furthermore, many speech therapists in order to teach a child the proper way for sound production they touch their throats while they are producing a sound and teach the child how to do the same. In this way they are feeling the vibration of the vocal chords and are learning to control more their voices. Especially for vowel phonation, other techniques related to aspiration sounds involve the placement in front of their mouth of their hands in order to feel the air that is getting in or out from it. This technique usually helps someone to pronounce consonants.

Finally the help of technology exists in this area too. Someone could try to make the ears work by using hearing aids or cochlear implants. Hearing aids make sound louder so that they fall into the sounds that the child may hear. In most cases, this is not enough to make distinctions within a spoken language, because the sound will be distorted and corrupted despite the use of very powerful hearing aids.

Cochlear implants are very advanced hearing aids that are placed into the inner ear, and replace the functionality of the ear. There are significant differences between the signal that is generated by an implant and a natural sound. Sound may be distorted and it is quite difficult to distinguish between other sounds. People who had an experience in listening (people who had hearing loss as adolescents or adults) may benefit from them and learn how to distinguish these sounds, but for children who lost their hearing in early age it is very difficult to learn and can take years of intensive training.

As a general outcome of the above analysis every sense is used to achieve a better sound production. All senses vision, audio, tactile, taste, olfaction are combined together to provide the patient with multimodal information. This kind of information is capable to teach the right way for placing and moving the tongue inside the child's mouth. Also, it could teach the right level of intensity of several vowels in order to avoid speaking too loud or too quiet. The same guidelines of speech therapy could be used for developing speech therapy software tools. Audio-visual feedback could be more easy managed, processed, and finally presented in a computer's screen. In the next sections we present a list of software tools that are developed to serve as speech therapy tools.

2.3 Related Work & Examples of Speech Therapy Software Multimedia Tools

In this section we are reviewing research [13, Maxine Eskenazi] in many areas of spoken language technology for education and especially for language learning for people with hearing loss. The main population target is consisting of children with post-lingual hearing loss.

The field is highly multidisciplinary. Computer science, statistics, signal processing, second language acquisition, cognitive science and linguistics are combined together for better results. Several names have been used for this field, such as Computer-Assisted Language Learning (CALL) and Computer-Assisted Language Technologies (CALT)(for the purpose of this work we will use the term that has been employed to describe work in Spoken Language Technology for Education, SLATE). We will review results by researchers using spoken language technology

for education. More specifically, researchers develop education applications using ASP (automatic speech processing), sometimes using natural language processing and/or spoken dialogue processing where the processing techniques are created or modified for this application. As previous bibliography research refers, "many of the techniques used in non-native pronunciation detection could be used for handicapped speech as well" [13, Maxine Eskenazi]. As a result of it we also are including in our report, multimedia tools which are used for second language learning tool. A brief description of each tool's functionality is provided. Before this description we focus to the types of feedback which are used in the majority of the tools.

2.3.1 Visual auditory feedback based on acoustic properties of speech

The goal of the computer assisted speech training systems is to provide sufficient auditory and visual feedback to the user in order to indicate corrective directions to pronunciation. Several training methods exists, which differ from each other mainly in the type of feedback [2, Klara Vicsi]. In the following (Table: 2.4) we can see speech properties which are used in many software applications as metrics in comparative process [14, Overview of SpeechViewer III]. Also a further explanation is provided.

Speech properties
Pitch or fundamental frequency
Speech waveforms
Prosody
Speech rate
Spectrogramms
Phoneme pronunciation
Articulation and coarticulation

Table 2.4: Frequently used speech properties

2.3.1.1 Pitch or fundamental frequency

Speech signal is characterized by voiced, unvoiced and silence regions [15, Sakshat Virtual Labs]. Voiced speech is produced because of the near periodic vibration of vocal folds. On the other hand, the random like vibration produces unvoiced speech. For silence region there is no vibration. In English and Greek language the biggest part of speech signals are voiced and include vowels, semivowels and other voiced components. Voiced regions of speech signals are similar to near periodic signal in the time domain representation. For the voiced speech segments we could assume to be periodic for speech processing purposes. This periodicity of voiced regions defines "pitch period T0" in the time domain and "Pitch frequency" or Fundamental Frequency "F0" in the frequency domain. Pitch is an important property of voiced speech. It contains personalized information depending on speaker. It is also essential for speech coding.

2.3.1.2 Speech waveforms

Waveforms are often used for speech visualization as in Figure 2.1. Speech waveforms are not very useful as they are difficult to be understood by students, however Bernstein and Christian [16, Bernstein J and Christian B] wrote in their paper that experiments have shown in such cases a visual display of the talker not only improves the word identification accuracy, but also the speech rhythm

and timing [17, Markham D and Nagano Madesen Y]. Today many commercial pronunciation tools offer this type of visual feedback.

A waveform is a two dimensional representation of a sound. The two dimensions in a waveform display are time and intensity. Vertical dimension is intensity and the horizontal dimension is time. Waveforms are also known as time domain representations of sound because they represent changes in intensity over time. Actually the intensity dimension is a display of sound pressure. Sound pressure is a calculation of small variations in air pressure which are perceivable as sound. People will hear louder sound with greater variations in sound pressure.[18, Waveform definition].

There are two types of speech sound source:

1. periodic vibration of the vocal folds resulting in voiced speech
2. aperiodic sound produced by turbulence at some constriction in the vocal tract resulting in voiceless speech.

The first type is being displayed in a waveform like a near periodic signal for voiced parts of speech signal, while second type is being displayed like noise.

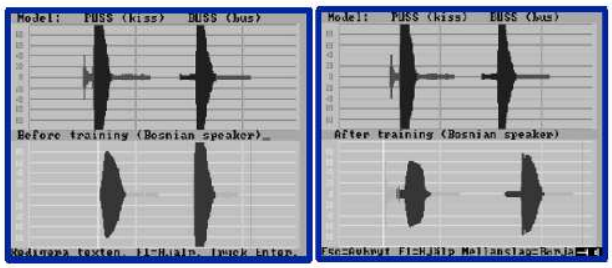


Figure 2.1: Wave form display in the IBM Speech Viewer

2.3.1.3 Prosody

Prosody in etymology, is the rhythm, stress, and intonation of speech. Prosody may reflect different characteristics of the speaker or the expression: the emotions of the speaker; the type of the utterance (explanation, question, or order); or different components of dialect that may not be encoded by punctuation or choice of vocabulary such as irony or sarcasm.

Regarding acoustics, the prosody includes variety in syllable length, loudness and pitch. In sign language communications, prosody includes the rhythm, length, and pressure of signals, alongside mouthing and facial expressions. Prosody is ordinarily non attendant in writing, which can sporadically lead reader to misunderstanding. Orthographic techniques to check or substitute for prosody incorporate accentuation (commas, exclamation marks, question marks, scare quotes, and ellipses), and typographic styling for emphasis (italic, strong, and underlined content). Children with hearing loss face prosody issues, because it is difficult for them to learn how to use speech rate properly or to ask a question (increase of pitch at the end of a sentence) [19, Prosody].

2.3.1.4 Speech rate

Speech rate is characterized as the rate at which a speaker executes the articulatory movements needed for speech. Researchers and clinicians have recommended that it is an important variable to measure during a diagnosis and to change when treating people who stammer. It has likewise been depicted as a component that

may help the onset, improvement, and support of stammering for some kids[20, Mark W Pellowski].

2.3.1.5 Spectrograms

A spectrogram, or sonogram, is a visual representation of the range of frequencies in a sound. Spectrograms also are called spectral waterfalls, voiceprints, or voicegrams. Spectrogram plots selected input signal's amplitude as a function of frequency and time in excellent shade. Spectrograms could be utilized to recognize spoken words phonetically as each phonem has a specific spectrogram print. They are utilized broadly in the research field of music, sonar, radar, speech processing, seismology etc. Figures 2.2, 2.3 underneath demonstrate spectrogram, where frequencies are on the vertical axis and time on the horizontal axis [21, Spectrogram].

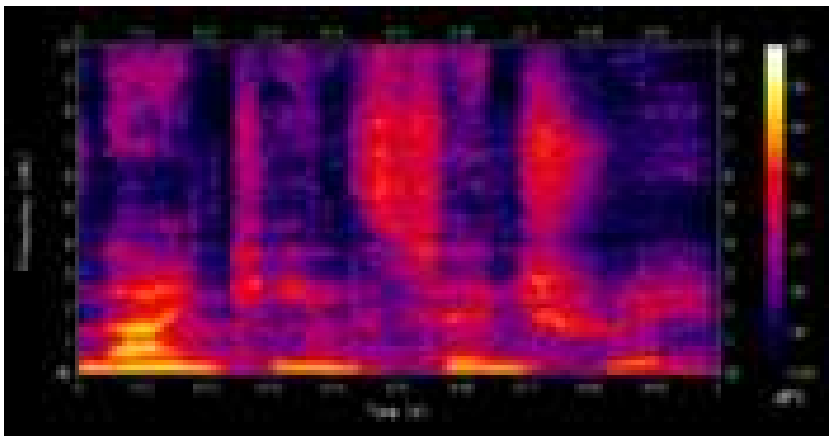


Figure 2.2: Typical spectrogram of the spoken words "nineteenth century".

In Figure 2.2 the lower frequencies are more dense because it is a male voice.You can see that the color intensity increases with the density.

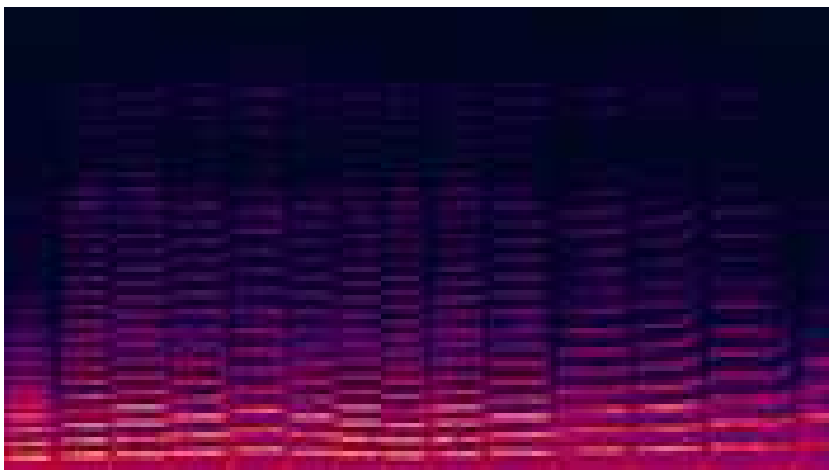


Figure 2.3: Spectrogram of the actual recording violin playing.

In Figure 2.3 you can note the harmonics occurring at integer multiples of the fundamental frequency.

2.3.1.6 Phoneme pronunciation

The term linguistics is the set of spoken sounds in any given language that serve to recognize a single word from an alternate. A phoneme may comprise of a

few phonetically different articulations, which are viewed as identical by listeners/speakers, since one articulation may be substituted for an alternate without any change of importance. Accordingly /p/ and/b/ are discrete phonemes in English because they differentiate such words as "pet" and "bet", while the light and dark /l/ sounds in "little" are not separate phonemes since they may be transposed without changing meaning. [22, David J Ertmer].

2.3.1.7 Articulation and co-articulation

By definition, articulation is the demonstration of vocal articulation. In simple words how we pronounce a speech sound. Despite the fact that articulation may appear easy and is not something that we do on purpose, in reality it is a complex procedure where we utilize the structures and muscles within our mouths to make specific movements that create particular sounds or a combo of sounds. The structures that we use to articulate, are called articulators and include: lips, teeth, tongue, top of the mouth, jaw, and lungs.

Co-articulation exists when a conceptually isolated speech sound is affected by a preceding or a following speech sound. There are two kinds of co-articulation: anticipatory co-articulation, when a characteristic of a speech sound is expected due to the creation of a preceding speech sound; and preservative co-articulation, when the impacts of a sound are seen due to the sound that follows.

Co-articulation in phonetics refers to two different phenomena. Firstly, stands for the assimilation of the place of articulation of one speech sound to that of an adjacent speech sound. For example, while the sound /n/ of English normally has an alveolar place of articulation, in the word {tenth} it is pronounced with a dental place of articulation because the following sound, /θ/, is {dental}. Secondly co-articulation refers to, the production of a co-articulated consonant, that is, a consonant with two simultaneous places of articulation. An example of such a sound is the voiceless labial-velar plosive /$\widehat{kp}$/ found in many West African languages. The term co-articulation may also refer to the transition from one articulatory gesture to another.

In next Figure 2.4 we demonstrate how energy of each formant is changing over time though spectrograms of the words *bed*, *dead*, and the nonword [geg] spoken by an American English speaker. White lines display second and third formant. As we can notice energy is influenced because of the presence of consonants in each word. At the beginning of the word *bed*, the second and third formants have a lower frequency than they do at the beginning of the word *dead*.The second formant is noticeably rising for the initial [b] from a comparatively low locus. In the word *dead*, the second formant is fairly steady at the beginning and the third formant drops a little. In [geg], the second and third formants come close to each other at the margins of the vowel, where the [g] consonants have the most influence over the formant frequencies [23, A Course in Phonetics].

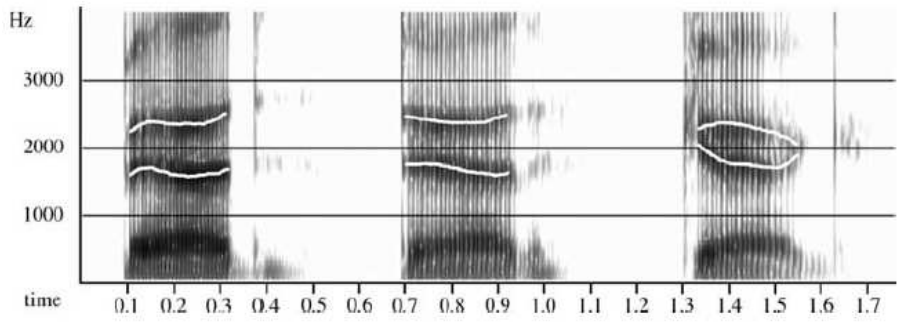


Figure 2.4: Spectrograms of the words *bed*, *dead*, and the nonword [geg].

2.3.1.8 Data visualization

Visualized data might be graphs of the above parameters. The efficacy of a system relies on the acoustic processing methods. The acoustical parameters used, and on the algorithm of the visualization. The visualized sound parameters - the sound pictures - must be fascinating and phonetically correct, giving feedback on whether the real articulation is correct or not and why.

Experiences on the depictions with spectral data propose their potential use as pronunciation feedback. It is critical to underline that the results depend first on the understanding of the parameters, secondly on the technique for visual presentation and thirdly on the directions on the most proficient method to translate the depictions. For instance, the spectrum interpretation by the IBM "Discourse Viewer" of the /u/ sound in Figure 2.5 is dry and hardly understandable for young children, but the other type of its visualization, presented in Figure 2.7, is clear and more suitable for small ages: an apple falling off a tree, when the pronunciation is correct[2, Klara Vicsi].

Figure 2.5: Spectrum interpretation U sound

Figure 2.6: Incorrectly pronounced U sound

Others have tried different things with utilizing a real-time spectrogram depiction of speech to give articulation feedback [22, David J Ertmer]. Generally they use comparative algorithms, but these pictures are too complicated for 5-year-old children.

2.3.2 Types of feedback

2.3.2.1 Audio and visual feedback

The scientists in KTH (Royal Institute of Technology, in Stockholm) created a speech intelligibility test to look at the part of visual data in speech intelligibility

Figure 2.7: Correctly pronounced U sound

- specifically, body gestures and lip reading. Noisy synthetic and natural speech sound was supplemented by an visible face and the intelligibility of the speech was tested.

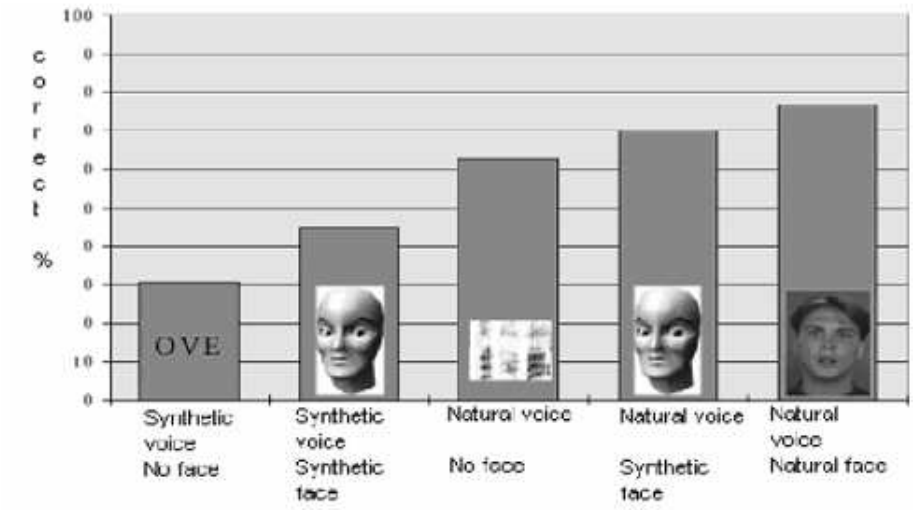


Figure 2.8: Combining speech reading, body gesture and synthesized face

The results obtained are displayed in Figure 2.8. It is obvious that the results show the improvement of intelligibility, when visual information is also present to the subjects [2, Klara Vicsi].

2.3.2.2 Synthetic Face

A visual representation of the trainees' articulator is an immediate and helpful technique. These are the process-oriented systems [2, Klara Vicsi]. The animated artificial agents, for example, model visual gestures in speech, utilizing a parametrically controlled visual speech synthesis based on a 3D polygonal model of a face.

In IDIAP (Dalle Molle Institute for Perceptual Artificial Intelligence), a speech reading system spots and tracks the lips of a speaker over a picture sequence to

concentrate visual speech data. The extracted characteristics portray the state of the lips and the intensity of the mouth area as suggested in Figure 2.9. The principle modes of intensity variety principally represent illumination and speaker differences instead speech data. Smaller modes of intensity variety represent speech data and portray the visibility of teeth and tongue. IDIAP simulates these features using Gaussian distribution and temporal dependencies using Hidden Markov Models.

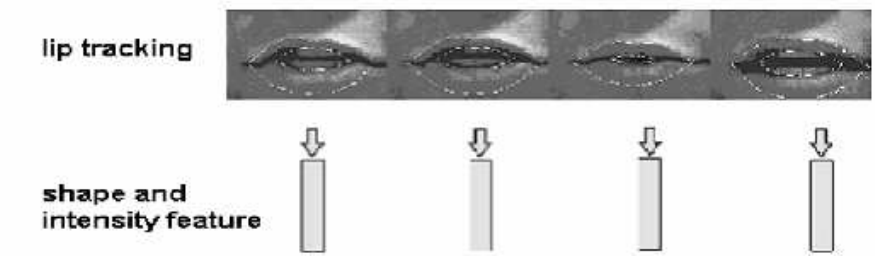


Figure 2.9: Extraction of Visual Speech Features

The animated agents can enhance learning and language education. Human faces advance interpersonal communication since they are informative, emotional and personalized. In different communication situations when data are vague and fuzzy we join together numerous sources of data audio and visual. At the time speech is produced, faces are useful linguistically and the auditory and visual features of speech are often complementary. Indeed, animated faces, for example, "BALDI" [24, Massaro Dominic W Light Joanna] can give feedback that people can't by turning semi-transparent to demonstrate the movements of the tongue inside the mouth from several aspects, or by displaying visual patterns that denote acoustic phonetic features of sounds.

2.3.2.3 Visualized Speech Properties

An alternate approach to help students learn speech is to visualize the acoustic properties of speech signal which are mentioned in the previous sections. These systems get speech signals and perform well if the measured acoustic - phonetic properties relate satisfactorily to the articulation movement. Speech properties might be displayed as sound pictures. Subsequently if the visualization methodology is right, then there is a correspondence between the articulation and the sound pictures Figure 2.10.

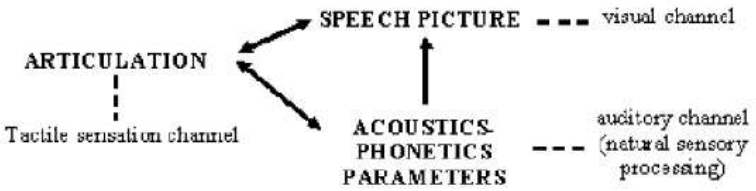


Figure 2.10: Correspondence between the articulation and the sound pictures

2.3.2.4 Automatic Feedback

In the speech learning process, the right sound or visual feedback, is extremely essential and helps the speech improvement of the trainees. In addition, numerous speech-training software tools have a sort of automatic feedback too, relying on in view of the acoustic similarities between the trainee's speech and a template. In the SPECO [25, Vicsi K Roach P Oster A Kacic Z Barczikay P Tantos A

Catari F Bakcsi Zs and Sfakianaki A] and in the ISTRA (Indiana Speech Training Aid, [26, Communication Disorders Technology Inc]) systems for children with hearing loss, the similarity between a metric of each new utterance and a stored template estimates the desirable acoustic similarity metric which is used to measure performance of the trainee.

In the second group of speech training systems phoneme-based Hidden Markov Models from automatic speech recognition technology (ASR) are used in order to evaluate pronunciation. However evaluation of ASR systems from educators gives ambiguous results. Sometimes automatic feedback does not work correct on the basis of the automatic speech recognition technology, misleading the trainees to get deteriorate results than those without utilizing any automatic feedback. From speech technology aspect, the challenge is whether today's ASR algorithms can be used to detect good and poor pronunciations of a known word spoken by a child.

By differentiation, the typical objective of the ASR is to order all utterances accurately, regardless of the possibility that they are not pronounced correctly. ASR systems can utilize either one kind of feedback or all; it relies on upon the actual purpose of the tool.

2.3.3 Speech therapy software tools

In this section we provide a short description of Computer-Aided Speech and Language Therapy (CASLT) that are being developed in scientific community. Furthermore, an extended description and results are displayed from two important speech therapy tools (SPECO, BALDI) in order to highlight the benefits for their users. The list which is presented here is not intended to be an exhaustive one but only indicative and informative. The main purpose of presenting these speech therapy tool is to give the reader the opportunity to understand the process of language learning for people with hearing impairments and the main features that are helping the user in this direction. The underlying speech technologies are not analyzed but only mentioned. The interactive tools are intended to encourage the acquisition of language skills in the areas of basic phonatory skills, phonetic articulation and language comprehension primarily for children.[27, Oscar Saz Shou Chun Yin Eduardo Lleida Richard Rose Carlos Vaquero William R Rodriguez].

2.3.3.1 Comunica project

"Comunica project" was developed by scientists of the Aragon Institute for Engineering Research (I3a) with the supervision of the CPEE "Alborada".

Three tools are part of the "Comunica" project [27, Oscar Saz Shou Chun Yin Eduardo Lleida Richard Rose Carlos Vaquero William R Rodriguez]:

1. "PreLingua" teaches basic phonation skills to children with neuromuscular issues.

2. "Vocaliza" aims to train mainly proper articulations of language.

3. "Cuentame" introduces language comprehension to impaired children.

2.3.3.2 PreLingua

PreLingua accumulates a set of game-like applications that use speech processing to exercise children with speech developmental delays, aiming to help speech therapy procedure. A feature extraction diagram is used for the training of five speech properties in the games (voice activity, intensity, breathing, tone and vocalization).

Voice activity games are developed for children with a developmental disability that delays their speech, compared to infants who still do not associate their production of sounds to changes in their environment. The output of the system is a binary voice activity signal focused around a variable threshold over the framewise energy of the input signal. When input signal is present, a reaction in the screen of the computer in the form of animated shapes and colors is produced. Extremely straightforward feedback is given in these games, as they are oriented to small children with severe disabilities. This kind of games have also been recommended by specialist and instructors as helpful for the early excitation of infants with severe disorders.

Intensity games permit a patient who has quite recently taken in the capacity to recognize speech production to learn to figure out how to control the volume of that production. Speech intensity is calculated as the framewise energy of the input signal and is also used for the Voice Activity Detection (VAD). In intensity games, an animated character passes screen from left to right (i.e. maze) and its position in the vertical axis is corresponding to the intensity of the speech production. With this technique, the user has to modulate the intensity to avoid obstacles or interact with secondary characters on screen by raising or lowering the volume of speech.

Breathing games utilize the assessed sonority value and applies a limit over it to discover low sonority frames associated to unvoiced areas. The detection of these unvoiced speech areas creates a movement in the screen (a character blows windmills or a ball climbs up a blowpipe) resembling traditional techniques in speech therapy to train this property.

Tone games follow the same approach as intensity games however they require the user to control the fundamental frequency or pitch instead of intensity, which is also needed for a correct speech production. The fundamental frequency, is used where the main character (butterfly) moves up and down as the user rises or lowers the fundamental tone to make it interact with other characters, while the pitch curve is shown on the upper right corner to help the therapist. Vocalization games goal is to transmit to the child the proper articulation of the vowels. In order to fulfill it's purpose vocalization games, plot the formant map with the correct standard distribution of the vowels. Because vowel map depends from language, vocalization games were are initially developed to the five Spanish vowels:/a/, /e/,/i/, /o/ and/u/. In the games, extraction of formants is made using LPC analysis and the result is depicted in the screen in the formant map, where the user can compare that vowel to the standard values. In improved versions of the game vocal tract normalization would be further needed to adapt the standard values of formants to every user.

All the games within the "PreLingua" framework do not require any previous configuration apart from the use of a microphone and their educative value, relies on the robustness of the speech processing and in the use of simple interfaces to provide of reinforcement and stimulation to the users (very young children with severe disabilities).

2.3.3.3 Vocaliza

"Vocaliza's" main purpose is to train articulation of the user in isolated words and short phrases. While the basic task of "Vocaliza" is to focus on the articulatory aspect of the language, it also introduces the user to the semantics and syntax levels of language with several activities. "Vocaliza"'s configuration interface is the way in which the therapist creates the profiles for the different users of the application. These profiles contain all the data related each patient practice's with "Vocaliza" (words to practice, acoustic data and interface necessities of each kid). When a user profile is made, the core of the application is consisted of four activities which

Figure 2.11: Tone game in PreLingua

are created for speech and language training. Speech technologies are used in order to supply user correct feedback. Below this structure, the user interface takes as input patient's speech; only the output of the system (text, audio and images) will be displayed in automatic way with the completeness of activities by the patient, not requiring any supervision by the therapist. Activities for speech and language training, the use of speech techniques and the user interface in "Vocaliza" are described in the following sections.

2.3.3.3.1 Activities for language training

To make speech and language therapy fascinating for kids, "Vocaliza" pratices three levels of the language (phonological, semantic and syntactic) presenting several activities. The phonological level of practicing is encouraging the user to pronounce a set of words which are preselected by a speech therapist during the configuration procedure to focus on the special needs of every user. The application uses ASR decoding on the pronunciation to accept, reject and evaluate the accepted utterances via a word-level pronunciation verification (PV) calculation and displaying a score as the final outcome of the game.

The semantic level is practiced presenting a riddle game which are preselected by a speech therapist. The application is making a question to the user providing three possible answers. The user must pronounce correct answer and ASR system must accept it, in order to continue with the next riddle. The application will display again score relying upon the capability of the user to solve the riddle.

The user is practicing with the syntactic level uttering a set of phrases, which are preselected by a speech therapist. Once again, the application is using ASR in order to decode and accept the input pronunciation. If input pronunciation is accepted, evaluation is taking place and score is displayed to the user.

2.3.3.3.2 Speech technologies for speech and language therapy

Speech technologies which are used by "Vocaliza" are ASR, speech synthesis, acoustic user adaptation and PV (pronunciation verification). ASR is the main technology of the application. Speech therapy activities needs ASR to decode user pronunciation, and to decide which word sequence had correct pronunciation. In next step application informs user that the game has been completed successfully. Therefore, high performance of the ASR system embedded in the application is strongly needed. Evaluation is done over a corpus with several impaired young children.

Speech synthesis gives an approach to display the user correct pronunciation of a word or sentence, pointing out the correct pronunciation in the speech therapy

activities. Every word, phrase and riddle is synthesized to be displayed to the end user of the application during the games.

Speaker adaptation enables the application to calculate speaker-dependent acoustic models adapted to each user. Speaker adaptation is strongly needed for obtaining high performance, since impaired speech can have negative affect in performance of ASR, so that users who suffer from severe speech issues would not be able take advantage of the application.

PV is the route in which the application provides an evaluation in the improvement of user communication skills. "Vocaliza" uses a word-level Likelihood Ratio (LR)-based Utterance Verification (UV)-procedure to assign a metric of confidence to each hypothesized word in an utterance. This technique calculates the distance (as a ratio) between the likelihood of the input pronunciation to two models (one generated from non-impaired speech and one adapted to impaired speech).

2.3.3.4 Cuentame

"Cuentame" ("Tell me" in Spanish) is developed for children with delays in oral language learning and aims to improve their communicative skills. It shares same philosophy with "Vocaliza". "Cuentame" allows children to interact with the application without supervise after necessary configuration of the application by the speech therapist.

2.3.3.4.1 Activities for language training

Three activities are developed into the application. All of them consists in scenarios of increasing levels of difficulty. Each scenario has to be solved by the user via speech. User is prompted to pronounce fully structured phrases in all the activities via several audio-visual rewards. In question-answering activities system asks user an open-ended question. In next step, user has to provide an answer that matches the set of possible correct answers of that the program has generated. In figure 2.12 is depicted how application chooses all the possible answers. Then the speech therapist selects the question that will be displayed to the patient and a one-word answer to it (because therapist has to type only one word, configuring all the activities is simplified). A certain number of correct sentences over the data provided are generated by syntax and semantic analysis. When the user answers the question, an ASR system looks for the keywords generated in the configuration step.

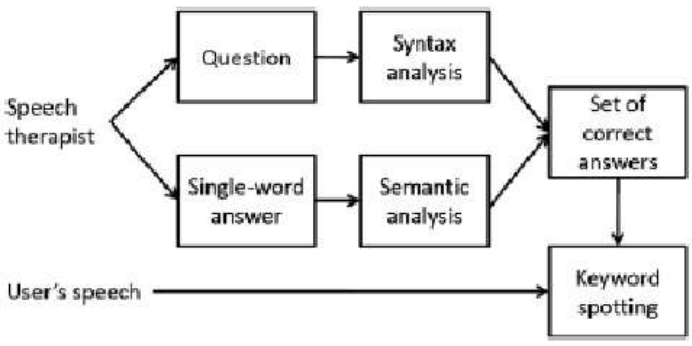


Figure 2.12: Generation of possible answers in "Cuentame"

The descriptive activities goal is the description of an object by the user due to a given group of attributes (shape, color, etc.); the user has to describe the object until filling up all the attributes. Once again, the user has to use natural language

and a set of possible correct phrases in order to give description of objects. Each attribute is generated as in previous figure.

The dialog activities are developed to take after an oral command control interface in which a certain environment is displaying to the user (house, school, shop). The user can interact with the environment with several actions (open, take, push, etc) and can use several objects (door, chair, TV, etc) and is asked to pronounce pairs of them (action-object) following a scenario of actions that lead to the desired target achievement proposed by the application and the therapist (for example, turn on the TV).

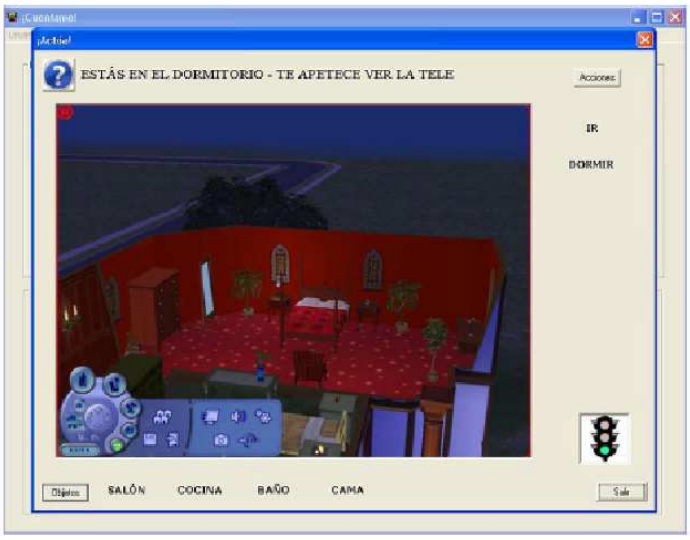


Figure 2.13: "Cuentame" interface

2.3.3.5 SPECO

The SPECO Project was founded by the EU through the INCO-COPERNICUS program (Contract no. 977126) in 1999 [28, K Vicsi and A Vary]. In SPECO project an audio-visual pronunciation teaching and training tool has been developed for use by 5-10 years old children. Correction of disordered speech progresses by real time visual display of speech properties, in a way that is easy to understand and fascinating for young children. The development of the speech by this method is taking place basically on visual feedback using the intact vision channel of the hearing impaired child. However, during practice limited auditory channel is being used too, by giving auditory information synchronised with the vision. This multimodal training and teaching system have been developed for four languages English, Swedish, Slovenian and Hungarian.

SPECO system consists of two sections: the first section is a language-independent frame program, named as Measuring System and Editor while the second is a Language Dependent Reference Database file. Their combination is the Teaching and Training Support, which is the application for users. Generally, SPECO project has the ability to adapt teaching and training support of any language using a well-defined database of the language. It calculates the different acoustic-phonetic properties of the speech signal, supports user in selection of reference speech examples and in placement of the symbolic pictures and background pictures into their correct places. It is possible to create a vocabulary with a special structure, according to the language.

The SPECO system has great flexibility. As it is used in many cases of speech disorders, allowing the speech therapists to use it depending on the speech

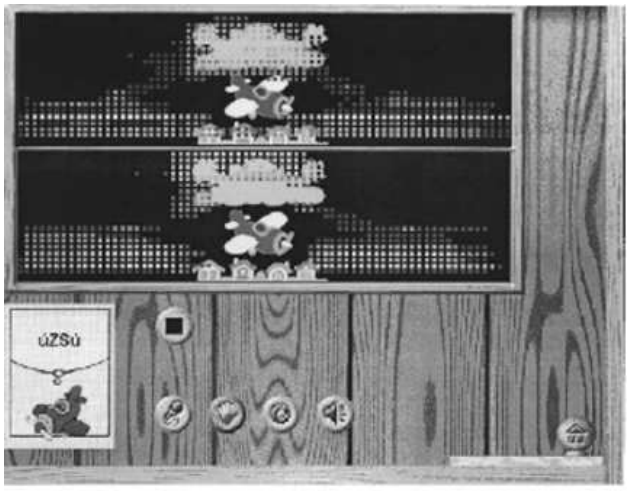


Figure 2.14: Comparing spectrograms of "uZu" (below) and reference (top)

defect. These are for example, the different speech disorders with normal hearing, with hearing impairment, etc. and in the special therapy in the case of cochlear implants.

2.3.3.6 Baldi

BALDI a 3-D computer-animated talking head [29, Dominic W. Massaro],[30, Baldi Youtube video] was developed relying on the value of visible speech in face-to-face communication. The quality and intelligibility of visible speech is simulated to regularly talking people. BALDI's visible speech can be used with either synthesized or natural auditory speech. BALDI simulates the inside of the mouth having teeth, tongue, and palate and his internal articulatory movements have been trained with electropalatography and ultrasound data from natural speech. Principles from linguistics, psychology and pedagogy where combined in order to help users with language delays and issues. BALDI can be used by individuals who are learning a new language.

It is possible using computer-based instruction to include embodied conversational agents rather than simply text or disembodied voices in lessons. Several reasons why the use of audiovisual data from a talking head is so successful exist. These include:

(a) the information in visible speech,

(b) the robustness of visual speech,

(c) the complementarity of auditory and visual speech, and

(d) the optimal integration of these two sources of information.

Figure 2.15: BALDI, a computer-animated talking head

Chapter 3

Background & Requirements

An extensive bibliography research has been done on speech therapy tools for children about 5-12 years old with several levels of hearing loss. The majority of speech therapy tools can be divided in two big categories. In the first category, the assisting tools are consisting of a set of simple game-alike speech exercises where a child has to interact with a computer in order to achieve certain goals. The interaction is achieved through audio and visual feedback where certain speech properties are viewed (pitch, voice intensity, rhythm, fricative/affricative pronunciation etc). Also guidelines in the placement of speech organs (tongue, teeth, palate etc) are provided through pictures. Some examples of the first category are SPECO and Communica Project [2, Klara Vicsi], [25, Vicsi K Roach P Oster A Kacic Z Barczikay P Tantos A Catari F Bakcsi Zs and Sfakianaki A].

In the second category, tools are consisting of a set of simple speech exercises where a child is guided to complete through a virtual talking head. In these exercises a child is trained in order to develop skills about certain speech properties (pitch, voice intensity, rhythm, fricative/affricative pronunciation etc). Additionally, this approach is taking advantage of the facial expressions which are created in the process of communication. Furthermore, a child can learn how to use speech organs (tongue, teeth, palate etc) easily because of the ability of the tool to view the placement of the internal organs of speech for every speech syllable / target (transparent skin, several views of mouth). Facial expressions with the combination of audio feedback are crucial for the understanding of meaning. BALDI and Vivian [29, Dominic W. Massaro],[31, Sascha Fagel & Katja Madany],[30, Baldi youtube video] are the most representative examples on this category.

.

3.1 Our approach

The main disadvantage of the existing tools is that they are developed for commercial use. Therefore, the cost to obtain a speech therapy tool is quite high especially if it is oriented for public use (e.g in public schools for educational purposes). Moreover, these tools are not easily adaptive and flexible. As they are oriented for standalone commercial use, the update process lasts in time and costs money as most of the times to get an updated version requires to pay for the whole program again. Furthermore, none of the tools is developed for use by Greek children.

These disadvantages motivated us to propose a flexible, free distributed design approach. Our proposed tool is developed in Greek for use via Web. Therefore, an online speech therapy tool is suggested which will be available 24 hours a day for everyone. This speech therapy tool will be aimed for use by Greek children 5-12 years old, with several levels of hearing loss and will be free of charge. As it will

be available through WEB, it has no update and distribution limitations.

More specifically, speech therapy tools are in the form of browser game collection. Input is received through microphone, and users of the tool receive feedback through screen (visual) and speakers (audio). In each browser game a speech property is being tested. The user tries to achieve certain goals for this speech property. Speech properties which are tested are pitch detection, voice intensity and phoneme pronunciation through spectrogram recognization. However, more speech properties can be added in future. After all in web-based applications this is quite easy.

Additionally, statistical analysis is provided in order to follow children's performance on each task. Users of browser games will be called to login to the system in order to keep their statistics. Special information graphs are generated demonstrating children's performance through time for difference tasks. Moreover, through the "performance statistics" feedback can also be provided to the supervisors of the tool. Games not so assisting on children can be replaced by others. Some indicative scenario examples of the user interaction with the proposed browser games are described below:

1. Pitch detection. The user is talking to the microphone. In the screen appears a spaceship which is travelling in space and an asteroid. User has to try to land starship on asteroid only by changing the pitch of his voice. Starship is looping over the space until starship lands on asteroid.

2. Voice intensity. The user is talking to the microphone. In the screen appears a spaceship which is travelling in space and several asteroids which they form several patterns. The user is trying to manage voice intensity in order to reach every asteroid. Starship is travelling until end of screen is reached.

3. Phoneme pronunciation - Spectrogram recognition. The user is talking to the microphone. Spectrograms for each phoneme is produced. User tries to match his spectrogram production with reference spectrograms that are provided through our web page.

3.2 Implementation

In order to implement our design approach we took advantage of the abstraction and scalability of one of the mainstream frameworks such as Apache Tomcat and Apache Shiro [32, Apache Tomcat], [33, Apache Shiro]. This approach provides us with the necessary technologies in order to achieve content and appearance separation, database abstraction according to the MVC (Model View Content) model along with advanced user management and platform agnostic data source technologies such as REST. One indicative solution could use Apache Tomcat along with Apache Shiro, Hibernate, RestEasy and MySQL [34, RestEasy], [35, MySQL], [36, Hibernate]. In a possible scenario of the interaction of the user with the system, the user will be presented with a login screen, type his credentials, be authenticated and redirected to a web site with the available browser games. In order to accomplish these tasks a coordination of several steps will be required. The Apache Shiro, that is a Java security framework, will retrieve the available credentials from the database through the Hibernate ORM and compare it with those provided by the user. If these credentials match to each other, then user is redirected to the home page of our web site and the available browser games will be presented to the user. After the successful login and the completion of one of the available browser games, the game application will connect with Apache Tomcat in order to save the scores achieved by the user and retrieve statistical information about previous

games. This is achieved through the combined use of the Apache Shiro, the Hibernate ORM and RestEasy modules. The architecture of the described procedure is displayed in the following diagram figure 3.1.

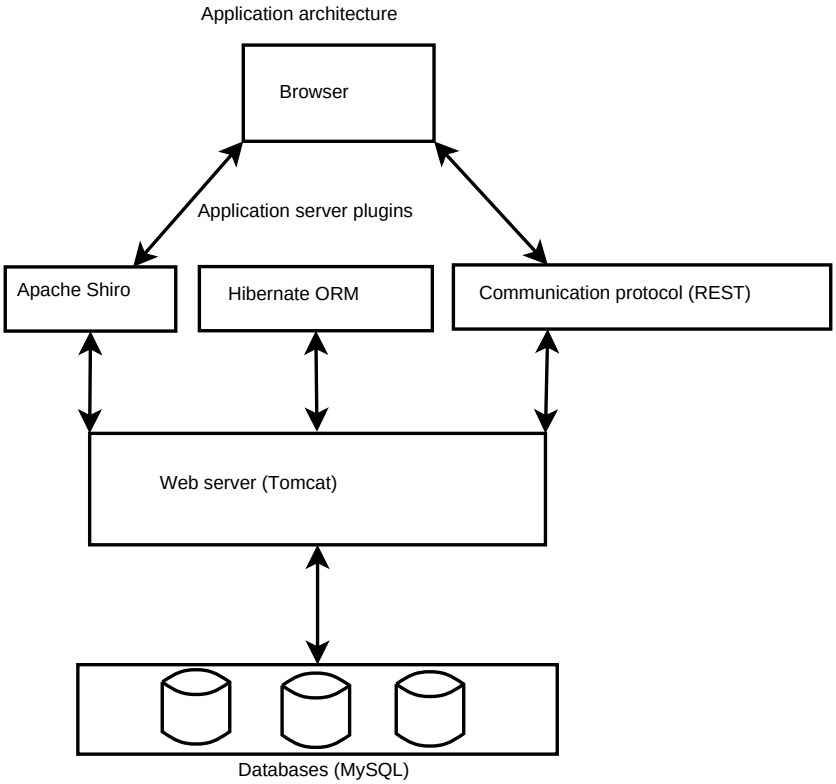


Figure 3.1: Architecture of our approach

The browser games are written either as Java applications (applets) or Javascript, that are receiving input from the sound devices of the running system. A comparison of both technologies is taking place. User interaction will be achieved through speech parameters and image variations that will follow speech parameters(visual feedback). JAVA applets and Javascript applications are implemented in Eclipse and tested through all known browsers for their functionality (IE, Mozilla Firefox, Chrome, Safari, Opera).

3.3 Brief description of each cooperating part of our system

As it is mentioned in previous chapters, the field is highly multidisciplinary. It benefits from knowledge in computer science, statistics and signal processing. Also a designer of game interfaces for children has to take into consideration the childhood nature in order to earn their interest. This could be achieved through attractive colors and interesting scenarios or missions of each game.

In our basic scene for our games a spaceship travels through space and has to land for supplies to several asteroids. Each asteroid symbolize a space station. Further more the height of each asteroid stands for one speech property, in our case pitch and intensity (sound pressure level). The system draws a spaceship in new height position according to the estimated pitch and SPL. Both calculations and drawings are taking place in real time. The result of this effort is the desired voice training for each level.

Audio channel is not used because we focus only in visual feedback. Also we consider our user profile to be consisted from children 5-12 years old with

post lingual severe hearing loss. Therefore, it would be much less important to provide feedback from audio channel too. Despite this fact, audio feedback could be implemented in future versions. Spectrograms of phoneme pronunciation is also provided in real time for comparison with reference spectrograms. In this section we will present in simple words a brief description of each involving part and how everything is cooperating with each other in order to achieve desired user interaction experience.

3.3.1 Speech processing

3.3.1.1 Pitch estimation

Main purpose of a pitch detection algorithm (PDA) is to calculate the pitch of a quasiperiodic or virtually periodic signal. Some typical examples of periodic signal could be a digital recording of speech or a musical note or tone. Pitch detection algorithms could be calculated either in time domain or in the frequency domain or in both domains. PDAs are used in various areas (e.g. phonetics, speech coding etc) and so different demands are placed upon the algorithm. Nowadays there is no single ideal PDA, so several algorithms exist, most of them are classified in the categories below [37, Pitch detection algorithm].

3.3.1.1.1 Time-domain approaches

In the time domain, a PDA calculates the period of a quasiperiodic signal, then inverts that value in order to estimate frequency. One basic methodology would be to measure the distance between zero crossing points of the signal (i.e. the Zero-crossing rate). However, this may not work equally well with complex waveforms because they are made out of multiple sine waves with differing periods. Despite that fact, zero-crossing can be a useful measure sometimes, e.g. in some speech applications where there is only one single source. Because of the algorithm's simplicity it is "cheap" to implement.

More clever methodologies compare segments of the signal with other segments moved by trial period to find a match. This is basic algorithm functionality of autocorrelation algorithms like AMDF (average magnitude difference function) or ASMDF (Average Squared Mean Difference Function). These algorithms can produce excellent results for highly periodic signals but when they are used on noisy signals they have false detection problems (often "octave errors") and - in their basic implementations - do not deal well with polyphonic sounds (which involve multiple musical notes of different pitches).

Basic core of current time-domain pitch detector algorithms is created with additional improvements to bring the performance more in line with a human evaluation of pitch. For instance, YIN algorithm is based upon autocorrelation [37, Pitch detection algorithm].

3.3.1.1.2 Frequency-domain approaches

In the frequency domain, calculation of polyphonic signal is possible usually using the periodogram to convert the signal to frequency spectrum. Processing power grows up as the desired accuracy increases, despite the well-known efficiency of the FFT which is a part of estimating periodogram algorithm, makes it suitably efficient for many purposes.

Steps of popular frequency domain algorithms include: the harmonic product spectrum; cepstral analysis and maximum likelihood which attempts to match the frequency domain characteristics to pre-defined frequency maps (useful for

detecting pitch of fixed tuning instruments); and the detection of peaks due to harmonic series.[37, Pitch detection algorithm].

3.3.1.1.3 Spectral/temporal approaches

Spectral and/or temporal pitch detection algorithms, for example the YAAPT pitch tracking, in order to detect pitch they combine time domain processing utilizing an autocorrelation function such as normalized cross correlation, and frequency domain processing using spectral information. Next step is to find final pitch track among the candidates estimated from the two domains, utilizing dynamic programming. Benefits of these approaches is that the tracking error in one domain can be reduced by the process in the other domain [37, Pitch detection algorithm].

3.3.1.1.4 Fundamental frequency of speech

The fundamental frequency of speech ranges from 40 Hz (for example low-pitched male voices) to 600 Hz (for example children or high-pitched female voices). In order to detect pitch, autocorrelation methods need at least two pitch periods. For instance if someone wants to detect a fundamental frequency of 40 Hz then at least 50 milliseconds (ms) of the speech signal are required for processing. However, during 50 ms the fundamental frequency is not necessarily constant in the entire length of the window[37, Pitch detection algorithm].

3.3.1.1.5 YIN algorithm - The method

For the purposes of our work, we selected to implement YIN algorithm in order to detect pitch. It is based on the well-known autocorrelation method with a number of modifications that combine to prevent errors. The algorithm has several desirable features. There is no upper limit on the frequency search range, so the algorithm is suited for high-pitched voices and music. The algorithm is relatively simple and may be implemented efficiently and with low latency, and it involves few parameters that must be tuned. It is based on a signal model (periodic signal) that may be extended in several ways to handle various forms of aperiodicity that occur in particular applications. [38, YIN a fundamental frequency estimator for speech and music]. YIN algorithm includes 6 steps for pitch estimation. These are:

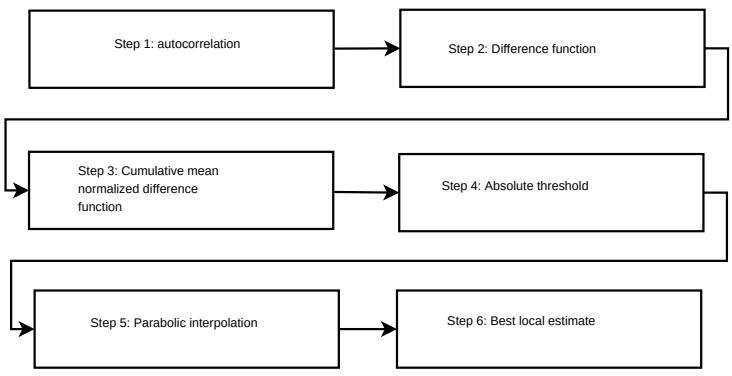


Figure 3.2: Basic flowchart for YIN algorithm

A more detailed description of each step is given below:

Step 1:The autocorrelation method

The autocorrelation function (ACF) of a discrete signal x_t may be defined as

$$r_t(\tau) = \sum_{j=\tau+1}^{\tau+W} x_j x_(j+\tau) \tag{3.1}$$

where $r_t(\tau)$ is the autocorrelation function of lag τ, calculated at time index t and W is the integration window size.

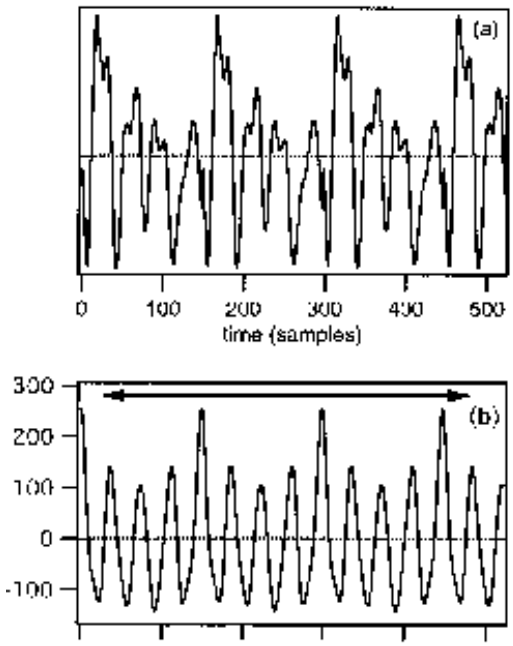


Figure 3.3: (a):Example of a speech waveform. (b):Autocorrelation function

Figure 3.3(b) show the autocorrelation function of the signal of Figure 3.3(a) in same figure. The ACF shows peaks at multiples of the period. The "autocorrelation method" chooses the highest non-zero-lag peak by exhaustive search within a range of lags horizontal arrows in Figure 3.3(b). The "autocorrelation method" chooses the highest non-zero-lag peak by exhaustive search within a range of lags (horizontal arrows in Figure 3.3). Obviously if the lower limit is too close to zero, the algorithm may erroneously choose the zero-lag peak. Conversely, if the higher limit is large enough, it may erroneously choose a higher-order peak.

The autocorrelation method compares the signal to its shifted self. In that sense it is related to the Average Magnitude Difference Function (AMDF) method that performs its comparison using differences rather than products, and more generally to time-domain methods that measure intervals between events in time. The ACF is the Fourier transform of the power spectrum, and can be seen as measuring the regular spacing of harmonics within that spectrum. The cepstrum method replaces the power spectrum by the log magnitude spectrum and thus puts less weight on high - amplitude parts of the spectrum (particularly near the first formant that often dominates the ACF).

Similar "spectral whitening" effects can be obtained by linear predictive inverse filtering or center-clipping, or by splitting the signal over a bank of filters, calculating ACFs within each channel, and adding the results after amplitude normalization. Auditory models based on autocorrelation are currently one of the more popular ways to explain pitch perception. Despite its appeal and many efforts to improve its performance, the autocorrelation method makes too many errors for

many applications. The following steps are designed to reduce error rates.

Step 2:Difference function

We start by modeling the signal x_t as a periodic function with period T, by definition invariant for a time shift of T:

$$x_t - x_{t+T} = 0, \forall t \tag{3.2}$$

The same is true after taking the square and averaging over a window:

$$\sum_{j=\tau+1}^{\tau+W} (x_j - x_{j+\tau})^2 = 0 \tag{3.3}$$

Conversely, an unknown period may be found by forming the difference function:

$$d_t(\tau) = \sum_{j=1}^{W} (x_j - x_{j+\tau})^2 \tag{3.4}$$

and searching for the values of τ for which the function is zero. There is an infinite set of such values, all multiples of the period. The difference function calculated from the signal in Figure 3.3(a) is illustrated in Figure 3.4.

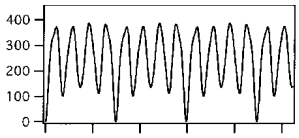


Figure 3.4: Difference function calculated for the speech signal of Figure 3.3 (a)

The squared sum may be expanded and the function expressed in terms of the ACF:

$$d_t(\tau) = r_t(0) + r_{t+T}(0) - 2r_t(\tau) \tag{3.5}$$

The first two terms are energy terms. Were they constant, the difference function $d_t(\tau)$ would vary as the opposite of $r_t(\tau)$, and searching for a minimum of one or the maximum of the other would give the same result. However, the second energy term also varies with τ, implying that maxima of $r_t(\tau)$ and minima of $d_t(\tau)$ may sometimes not coincide.

Step 3: Cumulative mean normalized difference

The difference function of Figure 3.4 is zero at zero lag and often non-zero at the period because of imperfect periodicity. Unless a lower limit is set on the search range, the algorithm must choose the zero-lag dip instead of the period dip and the method must fail. Even if a limit is set, a strong resonance at the first formant (F1) might produce a series of secondary dips, one of which might be deeper than the period dip. A lower limit on the search range is not a satisfactory way of avoiding this problem because the ranges of F1 and F0 are known to overlap. The solution that is proposed is to replace the difference function by the *"cumulative*

mean normalized difference function":

$$d'_t(\tau) = \begin{cases} 1, & \text{if } \tau{=}0 \\ \frac{d_t(\tau)}{(1/\tau)\sum_{j=1}^{\tau}(d_t(j))}, & \text{otherwise} \end{cases} \tag{3.6}$$

This new function is obtained by dividing each value of the old by its average over shorter-lag values. It differs from d(τ) in that it starts at 1 rather than 0, tends to remain large at low lags, and drops below 1 only where d(τ) falls below average Figure 3.5. Replacing d by d' reduces "too high" errors, as reflected by an error rate of 1.69% (instead of 1.95%). A second benefit is to do away with the upper frequency limit of the search range, no longer needed to avoid the zero-lag dip. A third benefit is to normalize the function for the next error-reduction step.

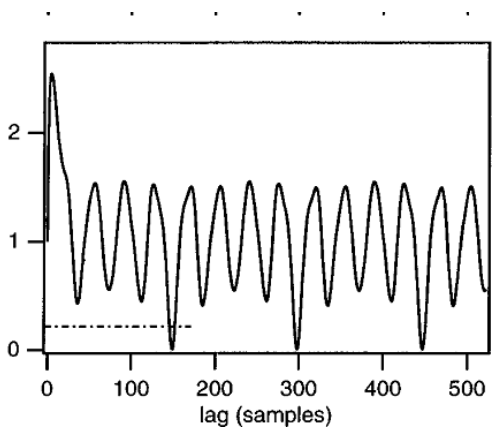


Figure 3.5: Cumulative mean normalized difference function of Figure 3.4 (a)

Step 4:Absolute threshold

It easily happens that one of the higher-order dips of the difference function in Figure 3.4 is deeper than the period dip. If it falls within the search range, the result is a subharmonic error, sometimes called "octave error" (improperly because not necessarily in a power of 2 ratio with the correct value). The autocorrelation method is likewise prone to choosing a high-order peak.

The solution we propose is to set an absolute threshold and choose the smallest value of τ, that gives a minimum of d' deeper than that threshold. If none is found, the global minimum is chosen instead. With a threshold of 0.1, the error rate drops to 0.78 % (from 1.69%) as a consequence of a reduction of "too low" errors accompanied by a very slight increase of "too high" errors. This step implements the word "smallest" in the phrase "the period is the smallest positive member of a set" (the previous step implemented the word "positive"). The threshold determines the list of candidates admitted to the set, and can be interpreted as the proportion of aperiodic power tolerated within a "periodic" signal. To see this, consider the identity:

$$2(x_t^2 + x_{t+T}^2) = (x_t + x_{t+T})^2 + (x_t - x_{t+T})^2 \tag{3.7}$$

Taking the average over a window and dividing by 4,

$$\frac{1}{2W}\sum_{j=t+1}^{t+W}(x_j^2 + x_{j+\tau}^2) = \frac{1}{4W}\sum_{j=t+1}^{t+W}(x_j^2 + x_{j+\tau})^2 + \frac{1}{4W}X\sum_{j=t+1}^{t+W}(x_j^2 - x_{j+\tau})^2 \tag{3.8}$$

The left-hand side approximates the power of the signal. The two terms on the right-hand side, both positive, constitute a partition of this power. The second is zero if the signal is periodic with period T, and is unaffected by adding or subtracting periodic components at that period. It can be interpreted as the "aperiodic power" component of the signal power. With t=T the numerator of Equation 3.6 is proportional to periodic power whereas its denominator, average of $d(\tau)$ for τ, between 0 and T, is approximately twice the signal power. Thus, d'(T) is proportional to the aperiodic/total power ratio. A candidate T is accepted in the set if this ratio is below threshold. We'll see later on that the exact value of this threshold does not critically affect error rates.

Step 5:Parabolic interpolation

The previous steps work as advertised if the period is a multiple of the sampling period. If not, the estimate may be incorrect by up to half the sampling period. Worse, the larger value of $d'(\tau)$ sampled away from the dip may interfere with the process that chooses among dips, thus causing a gross error. A solution to this problem is parabolic interpolation. Each local minimum of $d'(\tau)$ and its immediate neighbors is fit by a parabola, and the ordinate of the interpolated minimum is used in the dip-selection process. The abscissa of the selected minimum then serves as a period estimate. Actually, one finds that the estimate obtained in this way is slightly biased. To avoid this bias, the abscissa of the corresponding minimum of the raw difference function $d(\tau)$ is used instead.

Interpolation of $d'(\tau)$ or $d(\tau)$ is computationally cheaper than upsampling the signal, and accurate to the extent that $d(\tau)$ can be modeled as a quadratic function near the dip. Simple reasoning argues that this should be the case if the signal is band-limited. First, recall that the ACF is the Fourier transform of the power spectrum: if the signal x_t is bandlimited, so is its ACF. Second, the ACF is a sum of cosines, which can be approximated near zero by a Taylor series with even powers. Terms of degree 4 or more come mainly from the highest frequency components, and if these are absent or weak the function is accurately represented by lower order terms (quadratic and constant). Finally, note that the period peak has the same shape as the zero-lag peak, and the same shape (modulo a change in sign) as the period dip of $d(\tau)$, which in turn is similar to that of $d'(\tau)$. Thus, parabolic interpolation of a dip is accurate unless the signal contains strong high-frequency components (in practice,above about one-quarter of the sampling rate).

Step 6:Best local estimate

The role of integration in Eqs. 3.3 and 3.4 is to ensure that estimates are stable and do not fluctuate on the time scale of the fundamental period. Conversely, any such fluctuation, if observed, should not be considered genuine. It is sometimes found, for nonstationary speech intervals, that the estimate fails at a certain phase of the period that usually coincides with a relatively high value of $d'(T_t)$, where T_t is the period estimate at time t. At another phase (time t') the estimate may be correct and the value of $d'(T_{t'})$ smaller. Step 6 takes advantage of this fact, by "shopping" around the vicinity of each analysis point for a better estimate.

The algorithm is the following. For each time index t, search for a minimum of $d'_\vartheta(T_\vartheta)$ for ϑ within a small interval $[t\text{-}T_{max}/2,\ t\text{+}T_{max}/2]$, where T_ϑ is the estimate at time ϑ and T_{max} is the largest expected period. Based on this initial estimate, the estimation algorithm is applied again with a restricted search range to obtain the final estimate. Using T_{max}=25 ms and a final search range of $\pm$20% of the initial estimate, step 6 reduced the error rate to 0.5% (from 0.77%). Step 6 is reminiscent of median smoothing or dynamic programming techniques, but

differs in that it takes into account a relatively short interval and bases its choice on quality rather than mere continuity. The combination of steps 1-6 constitutes a new method (YIN). It is worth noting how the steps build upon one another. Replacing the ACF (step 1) by the difference function (step 2) paves the way for the cumulative mean normalization operation (step 3), upon which are based the threshold scheme (step 4) and the measure d'(T) that selects the best local estimate (step 6). Parabolic interpolation (step 5) is independent from other steps, although it relies on the spectral properties of the ACF (step 1).

3.3.1.2 SPL estimation

Sound pressure or acoustic pressure is the local pressure deviation from the atmospheric pressure, caused by a sound wave. We can calculate sound pressure in air using a microphone, and in water with a hydrophone. The SI unit for sound pressure p is the pascal (symbol: Pa). Sound pressure level (SPL) is a logarithmic metric of the effective sound pressure of a sound relative to a reference value. It is measured in decibels (dB) above a standard reference level. The standard reference sound pressure in air or other gases is 20 μPa, which is usually considered the threshold of human hearing (at 1 kHz) [39, SPL].

$$L_p = 10 \log_{10} \left(\frac{{p_{\text{rms}}}^2}{{p_{\text{ref}}}^2} \right) = 20 \log_{10} \left(\frac{p_{\text{rms}}}{p_{\text{ref}}} \right) \text{dB} \tag{3.9}$$

where p_{ref} is the reference sound pressure and p_{rms} is the rms sound pressure being measured.

Sometimes variants are used such as dB (SPL), dBSPL, or dBSPL. The commonly used reference sound pressure in air is p_{ref} = 20 μPa (rms) or 0.0002 dynes/cm2, which is usually considered the threshold of human hearing (roughly the sound of a mosquito flying 3 m away). Most sound level measurements will be made relative to this level, meaning 1 pascal will equal an SPL of 94 dB. In other media, such as underwater, a reference level of 1 μPa is used. These references are defined in ANSI S1.1-1994.

The lower limit of audibility is defined as SPL of 0 dB, but the upper limit is not as clearly defined. While 1 atm (194 dB Peak or 191 dB SPL) is the largest pressure variation an undistorted sound wave can have in Earth's atmosphere, larger sound waves can be present in other atmospheres or other media such as under water, or through the Earth.

Ears detect changes in sound pressure. Human hearing does not have a flat spectral sensitivity (frequency response) relative to frequency versus amplitude. Humans do not perceive low- and high-frequency sounds as well as they perceive sounds near 2,000 Hz, as shown in the equal-loudness contour in Figure 3.6. Because the frequency response of human hearing changes with amplitude, three weightings have been established for measuring sound pressure: A, B and C. A-weighting applies to sound pressures levels up to 55 dB, B-weighting applies to sound pressures levels between 55 and 85 dB, and C-weighting is for measuring sound pressure levels above 85dB.

In order to distinguish the different sound measures a suffix is used: A-weighted sound pressure level is written either as dBA or LA. B-weighted sound pressure level is written either as dBB or LB, and C-weighted sound pressure level is written either as dBC or LC. Unweighted sound pressure level is called "linear sound pressure level" and is often written as dBL or just L. Some sound measuring instruments use the letter "Z" as an indication of linear SPL.

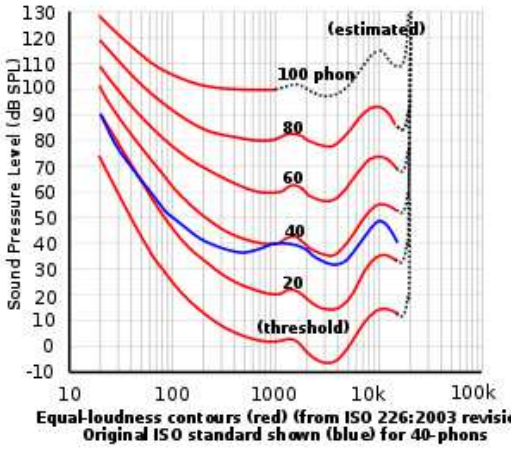


Figure 3.6: Equal-loudness contour

3.3.2 Apache Shiro

Apache Shiro is a compelling and adaptable open-source security framework that cleanly handles authentication, authorization, enterprise session management and cryptography.

Apache Shiro's main goal is to be easy to use and comprehend. Security can be exceptionally complex sometimes, even painful, but it doesn't have to be. A framework should solve complexities as soon as possible and supply user a easy and practical API that help developer's to develop secure application(s) [33, Apache Shiro].

Here are some things that Apache Shiro supports:

- Authenticate a user to verify their identity
- Perform access control for a user, such as:
  - Determine if a user is assigned a certain security role or not
  - Determine if a user is permitted to do something or not
- Use a Session API in any environment, even without web or EJB containers.
- React to events during authentication, access control, or during a session's lifetime.
- Aggregate one or more data sources of user security data and present this all as a single composite user 'view'.
- Enable Single Sign On (SSO) functionality
- Enable 'Remember Me' services for user association without login ... and much more - all integrated into a cohesive easy-to-use API.

Shiro attempts to achieve these objectives for all possible application environments - from the simplest command line application to the largest enterprise applications, without constraining conditions on other 3rd party frameworks, containers, or application servers. Obviously the project intends to integrate into these environments wherever possible, but it could be used out-of-the-case in any environment.

3.3.2.1 Apache Shiro Features

Apache Shiro is an understandable application security framework with many capabilities. The following diagram displays where Shiro focuses its development so far

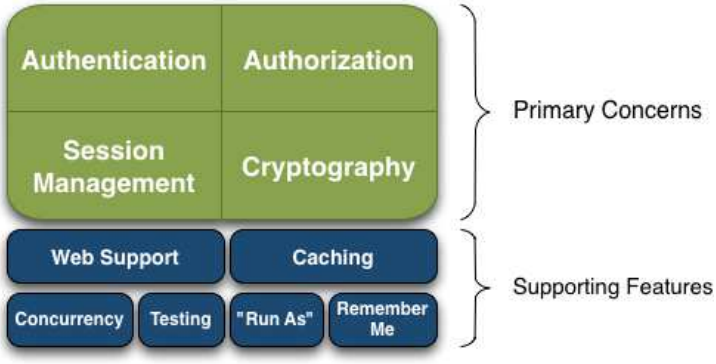


Figure 3.7: Shiro features

Shiro focused what the Shiro development team calls "the four cornerstones of application security" - Authentication, Authorization, Session Management, and Cryptography:

- Authentication: Sometimes referred to as 'login', this is the act of proving a user is who they say they are.
- Authorization: The process of access control, i.e. determining 'who' has access to 'what'.
- Session Management: Managing user-specific sessions, even in non-web or EJB applications.
- Cryptography: Keeping data secure using cryptographic algorithms while still being easy to use.

There are also additional features to support and reinforce these concerns in different application environments, especially:

- Web Support: Shiro's web support APIs help easily secure web applications.
- Caching: Caching is a first-tier citizen in Apache Shiro's API to ensure that security operations remain fast and efficient.
- Concurrency: Apache Shiro supports multi-threaded applications with its concurrency features.
- Testing: Test support exists to help you write unit and integration tests and ensure your code will be secured as expected.
- "Run As": A feature that allows users to assume the identity of another user (if they are allowed), sometimes useful in administrative scenarios.
- "Remember Me": Remember users' identities across sessions so they only need to log in when mandatory.

3.3.3 Hibernate ORM

Hibernate ORM (Hibernate in short) is an object-relational mapping library for the Java language, providing a framework for mapping an object-oriented domain model to a traditional relational database. Hibernate solves object-relational impedance mismatch problems by replacing direct persistence-related database accesses with high-level object handling functions. Hibernate is a free software that is distributed under the GNU Lesser General Public License. Hibernate's core feature is mapping from Java classes to database tables (and from Java data types to SQL data types). Hibernate also provides data query and retrieval features. It creates SQL calls and helps the developer to avoid manual result set handling and object conversion. Applications who use Hibernate can be transferred to supported SQL databases with little performance overhead [36, Hibernate].

3.3.3.1 Mapping

Mapping Java classes to database tables is accomplished through the configuration of an XML file or by using Java Annotations. When using an XML file, Hibernate can generate skeleton source code for the persistence classes. This is unnecessary when annotations are used. Hibernate can use the XML file or the annotations to maintain the database schema. Facilities to arrange one-to-many and many-to-many relationships between classes are provided. In addition to managing associations between objects, Hibernate can also manage reflexive associations where an object has a one-to-many relationship with other instances of its own type.

Hibernate supports the mapping of custom value types. This makes the following scenarios possible:

- Overriding the default SQL type that Hibernate chooses when mapping a column to a property.
- Mapping Java Enum to columns as if they were regular properties.
- Mapping a single property to multiple columns.

Definition: Objects in a front-end application follow OOP principles, while objects in the back-end follow database normalization principles, resulting in different representation requirements. This problem is called "object-relational impedance mismatch". Mapping is a way of resolving the impedance mismatch problem. Mapping tells the ORM tool which java class object an application is needed to be store in which table of database.

3.3.3.2 HQL

Hibernate provides an SQL inspired language called Hibernate Query Language (HQL) which allows SQL-like queries to be written against Hibernate's data objects. Criteria Queries are provided as an object-oriented alternative to HQL. Criteria Query is used to modify the objects and provide the restriction for the objects.

3.3.3.3 Persistence

Hibernate provides transparent persistence for Plain Old Java Objects (POJOs). The only strict requirement for a persistent class is a no-argument constructor, not necessarily public. Proper behavior in some applications also requires special attention to the equals() and hashCode() methods.

Collections of data objects are typically stored in Java collection objects such as Set and List. Java generics, introduced in Java 5, are supported. Hibernate can be configured to lazy load associated collections. Lazy loading is the default as of Hibernate 3. Related objects can be configured to cascade operations from one to the other. For example, a parent Album object can be configured to cascade its save and/or delete operation to its child Track objects. This can reduce development time and ensure referential integrity. A dirty checking feature avoids unnecessary database write actions by performing SQL updates only on the modified fields of persistent objects.

3.3.3.4 Integration

Hibernate can be used both in standalone Java applications and in Java EE applications using servlets, EJB session beans, and JBI service components. It can also be included as a feature in other programming languages. For example, Adobe integrated Hibernate into version 9 of ColdFusion (which runs on J2EE app servers) with an abstraction layer of new functions and syntax added into CFML.

3.3.3.5 Entities and components

In Hibernate jargon, an entity is a stand-alone object in Hibernate's persistent mechanism which can be manipulated independently of other objects. In contrast, a component is subordinate to an entity and can be manipulated only with respect to that entity. For example, an Album object may represent an entity but the Tracks object associated with the Album objects would represent a component of the Album entity if it is assumed that Tracks can only be saved or retrieved from the database through the Album object. Unlike J2EE, it can switch databases.

3.3.4 MySQL

MySQL is (since March 2014) ranked as the world's second most popular open-source relational database management system (RDBMS). My of MySQL was inspired by co-founder Michael Widenius's daughter, My. The SQL is an acronym for Structured Query Language. The MySQL project source code is distributed under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. [35, MySQL description].

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP and XAMPP open source web application software (and other 'AMP' software). Commercial editions are available too with extra features. Some representative applications which use MySQL include: TYPO3, MODx, Joomla, WordPress and others. Also several popular websites, such as Wikipedia, Google, Facebook have used MySQL.

3.3.5 XAMPP

XAMPP's name is an acronym for [40, XAMPP description]:

- X (to be read as "cross", meaning cross-platform)
- Apache HTTP Server
- MySQL
- PHP
- Perl

In order to use XAMPP a single zip, tar, 7z, or exe file to be downloaded and executed is reuired. Additionally no configuration of the various components that consist the web server is required. XAMPP periodically downloads latest updates in order to take advantage from latest releases of Apache, MySQL, PHP and Perl. It also provides extra features such as OpenSSL and phpMyAdmin. Further more self-contained, and multiple instances of XAMPP can exist on a single computer. Finally a given instance can be transferred from one computer system to another.

As developers of XAMPP declare XAMPP is intended to be used only as a development tool, in order to allow website designers and programmers to check their work on their own computers with no access to the Internet. To make this effort painless as possible, several important security features are disabled by default. Nevertheless, XAMPP can be used to actually serve web pages on the World Wide Web. A special tool is provided for password in order to secure the most important parts of the package.

XAMPP also stands for creating and managing several databases such as MySQL and SQLite. At the moment installation of XAMPP is ready, it is possible to treat a localhost just like a remote host by connecting using an FTP client. In the case of installing a content management system (CMS) like Joomla or WordPress utilizing a program like FileZilla has many advantages. Onother option is to connect to localhost via FTP with an HTML editor. The default FTP user is "newuser", the default FTP password is "wampp". The default MySQL user is "root" while there is no default MySQL password.

- XAMPP 1.8.3-4 for Windows, including:
  - Apache 2.4.9
  - MySQL 5.6.16
  - PHP 5.5.11
  - phpMyAdmin 4.1.12
  - FileZilla FTP Server 0.9.41
  - Tomcat 7.0.42
  - Strawberry Perl 5.16.3.1 Portable
  - XAMPP Control Panel 3.2.1
- XAMPP 1.8.3-4 for Linux, including:
  - Apache 2.4.9
  - MySQL 5.6.16
  - PHP 5.5.11
  - phpMyAdmin 4.1.12
  - OpenSSL 1.0.1

3.3.6 Representational state transfer (REST)

3.3.6.1 What is REST?

REST is named by Roy Fielding in his Ph.D. dissertation to describe an architecture style of networked systems. REST is an acronym standing for Representational State Transfer [41, Rest description].

3.3.6.2 Why is it called Representational State Transfer?

The Web is comprised of resources. A resource is any item of interest. For example, the Boeing Aircraft Corp may define a 747 resource. Clients may access that resource with this URL: http://www.boeing.com/aircraft/747

A representation of the resource is returned (e.g., Boeing747.html). The representation places the client application in a state. The result of the client traversing a hyperlink in Boeing747.html is another resource is accessed. The new representation places the client application into yet another state. Thus, the client application changes (transfers) state with each resource representation –> Representational State Transfer! Here is Roy Fielding's [42, Architectural Styles and the Design of Network-based Software Architectures] explanation of the meaning of Representational State Transfer:

"Representational State Transfer is intended to evoke an image of how a well-designed Web application behaves: a network of web pages (a virtual state-machine), where the user progresses through an application by selecting links (state transitions), resulting in the next page (representing the next state of the application) being transferred to the user and rendered for their use."

3.3.6.3 Motivation for REST

The motivation for REST was to conceive the features of the Web which made the Web successful. Subsequently these features are being used to guide the progress of the Web.

3.3.6.4 REST - An Architectural Style, Not a Standard

REST is not a standard neither a specification to be found in W3C. IBM or Microsoft can not sell a REST developer's toolkit. That is because REST is just an architectural style. You can't package up that style. You can only comprehend it, and use it in order to develop your Web services following in that style similar to the client-server architectural style. There is no client-server standard. Although REST is not a standard, it use standards:

- HTTP
- URL
- XML/HTML/GIF/JPEG/etc (Resource Representations)
- text/xml, text/html, image/gif, image/jpeg, etc (MIME Types)

3.3.6.5 The Classic REST System

The Web is a REST system by itself! Representative popular Web services are book-ordering services, search services, online dictionary services and others. So it's possible that you have been using REST, building REST services and you didn't even know it. REST is interested in the "big picture" of the World Wide Web and does not deal with implementation details (for example using Java servlets or CGI to implement a Web service). Here is an example of creating a Web service from the REST "big picture" aspect.

3.3.6.6 Parts Depot Web Services

Parts Depot, Inc (fictitious company) has deployed some web services to enable its customers to:

- get a list of parts

- get detailed information about a particular part
- submit a Purchase Order (PO)

Let's consider how each of these services are implemented in a RESTful fashion.

3.3.6.7 Get Parts List

The web service makes available a URL to a parts list resource. For example, a client would use this URL to get the parts list: http://www.parts-depot.com/parts

Note that "how" the web service generates the parts list is completely transparent to the client. All the client knows is that if he/she submits the above URL then a document containing the list of parts is returned. Since the implementation is transparent to clients, Parts Depot is free to modify the underlying implementation of this resource without impacting clients. This is loose coupling.

Here's the document that the client receives:

```
<?xml version="1.0"?>
<p:Parts xmlns:p="http://www.parts-depot.com"
  xmlns:xlink="http://www.w3.org/1999/xlink">
    <Part id="00345" xlink:href="http://www.parts-depot.com/parts/00345"/>
    <Part id="00346" xlink:href="http://www.parts-depot.com/parts/00346"/>
    <Part id="00347" xlink:href="http://www.parts-depot.com/parts/00347"/>
    <Part id="00348" xlink:href="http://www.parts-depot.com/parts/00348"/>
</p:Parts>
```

[Assume that through content negotiation the service determined that the client wants the representation as XML (for machine-to-machine processing)] Note that the parts list has links to get detailed info about each part. This is a key feature of REST. The client transfers from one state to the next by examining and choosing from among the alternative URLs in the response document.

3.3.6.8 Get Detailed Part Data

The web service makes available a URL to each part resource. Example, here's how a client requests part 00345: http://www.parts-depot.com/parts/00345

Here's the document that the client receives:

```
<?xml version="1.0"?>
 <p:Part xmlns:p="http://www.parts-depot.com"
  xmlns:xlink="http://www.w3.org/1999/xlink">
  <Part-ID>00345</Part-ID>
  <Name>Widget-A</Name>
  <Description>This part is used within the frap assembly</Description>
  <Specification xlink:href="http://www.parts-depot.com/parts/00345/specification"/>
  <UnitCost currency="USD">0.10</UnitCost>
  <Quantity>10</Quantity>
 </p:Part>
```

Again observe how this data is linked to still more data - the specification for this part may be found by traversing the hyperlink. Each response document allows the client to drill down to get more detailed information.

3.3.6.9 Submit PO

The web service makes available a URL to submit a PO. The client creates a PO instance document which conforms to the PO schema that Parts Depot has designed (and publicized in a WSDL document). The client submits PO.xml as the payload of an HTTP POST.

The PO service responds to the HTTP POST with a URL to the submitted PO. Thus, the client can retrieve the PO any time thereafter (to update/edit it). The PO has become a piece of information which is shared between the client and the server. The shared information (PO) is given an address (URL) by the server and is exposed as a Web service.

3.3.6.10 Logical URLs versus Physical URLs

A resource is a conceptual entity. A representation is a concrete manifestation of the resource. This URL: http://www.parts-depot.com/parts/00345

is a logical URL, not a physical URL. Thus, there doesn't need to be, for example, a static HTML page for each part. In fact, if there were a million parts then a million static HTML pages would not be a very attractive design.

[Implementation detail: Parts Depot could implement the service that gets detailed data about a particular part by employing a Java Servlet which parses the string after the host name, uses the part number to query the parts database, formulate the query results as XML, and then return the XML as the payload of the HTTP response.]

As a matter of style URLs should not reveal the implementation technique used. You need to be free to change your implementation without impacting clients or having misleading URLs.

3.3.6.11 REST Web Services Characteristics

Here are the characteristics of REST:

- Client-Server: a pull-based interaction style: consuming components pull representations.
- Stateless: each request from client to server must contain all the information necessary to understand the request, and cannot take advantage of any stored context on the server.
- Cache: to improve network efficiency responses must be capable of being labeled as cacheable or non-cacheable.
- Uniform interface: all resources are accessed with a generic interface (e.g., HTTP GET, POST, PUT, DELETE).
- Named resources - the system is comprised of resources which are named using a URL.
- Interconnected resource representations - the representations of the resources are interconnected using URLs, thereby enabling a client to progress from one state to another.
- Layered components - intermediaries, such as proxy servers, cache servers, gateways, etc, can be inserted between clients and resources to support performance, security, etc.

3.3.6.12 Principles of REST Web Service Design

1. The key to creating Web Services in a REST network (i.e., the Web) is to identify all of the conceptual entities that you wish to expose as services. Above we saw some examples of resources: parts list, detailed part data, purchase order.

2. Create a URL to each resource. The resources should be nouns, not verbs. For example, do not use this: http://www.parts-depot.com/parts/getPart?id=00345

   Note the verb, getPart. Instead, use a noun:

   http://www.parts-depot.com/parts/00345

3. Categorize your resources according to whether clients can just receive a representation of the resource, or whether clients can modify (add to) the resource. For the former, make those resources accessible using an HTTP GET. For the later, make those resources accessible using HTTP POST, PUT, and/or DELETE.

4. All resources accessible via HTTP GET should be side-effect free. That is, the resource should just return a representation of the resource. Invoking the resource should not result in modifying the resource.

5. No man/woman is an island. Likewise, no representation should be an island. In other words, put hyperlinks within resource representations to enable clients to drill down for more information, and/or to obtain related information.

6. Design to reveal data gradually. Don't reveal everything in a single response document. Provide hyperlinks to obtain more details.

7. Specify the format of response data using a schema (DTD, W3C Schema, RelaxNG, or Schematron). For those services that require a POST or PUT to it, also provide a schema to specify the format of the response.

8. Describe how your services are to be invoked using either a WSDL document, or simply an HTML document.

3.3.6.13 RestEasy

RESTEasy is a JBoss project that provides various frameworks to help you build RESTful Web Services and RESTful Java applications. It is a fully certified and portable implementation of the JAX-RS specification. JAX-RS is a new JCP specification that provides a Java API for RESTful Web Services over the HTTP protocol. [34, RestEasy]

RESTEasy can run in any Servlet container, but tighter integration with the JBoss Application Server is also available to make the user experience nicer in that environment.

3.3.6.13.1 RestEasy Features

Here are the features of RestEasy:

- Fully certified JAX-RS implementation
- Portable to any app-server/Tomcat that runs on JDK 6 or higher
- Embeddedable server implementation for junit testing

- Client framework that leverages JAX-RS annotations so that you can write HTTP clients easily (JAX-RS only defines server bindings)
- Client "Browser" cache. Supports HTTP 1.1 caching semantics including cache revalidation
- Server in-memory cache. Local response cache. Automatically handles ETag generation and cache revalidation
- Rich set of providers for: XML, JSON, YAML, Fastinfoset, Multipart, XOP, Atom, etc.
- JAXB marshalling into XML, JSON, Jackson, Fastinfoset, and Atom as well as wrappers for maps, arrays, lists, and sets of JAXB Objects.
- GZIP content-encoding. Automatic GZIP compression/decompression support in client and server frameworks
- Asynchronous HTTP (Comet) abstractions for JBoss Web, Tomcat 6, and Servlet 3.0
- Asynchronous Job Service.
- Rich interceptor model.
- OAuth2 and Distributed SSO with JBoss AS7
- Digital Signature and encryption support with S/MIME and DOSETA
- EJB, Seam, Guice, Spring, and Spring MVC integration

3.3.7 Java

Java is a concurrent, class-based, object-oriented computer programming language with minimum implementation dependencies as possible. Java aims to let application developers to write portable and platform independent code. Java applications are compiled to bytecode (class file) that can execute on any Java Virtual Machine (JVM) independent of computer architecture. Since 2014 Java is, one of the most popular programming languages, especially for client-server web applications. Java was originally designe by James Gosling at Sun Microsystems (merged into Oracle Corporation) and it's first release was in 1995 as a core component of Sun Microsystems' Java platform. Java is related to C and C++ regarding it's syntax, but it has fewer low-level facilities than either of them [43, Java]. Since May 2007, Sun relicensed Java under the GNU General Public License. Others have also developed alternative implementations of Sun technologies, like GNU Compiler for Java (bytecode compiler), GNU Classpath (standard libraries), and IcedTea - Web (browser plugin for applets).

3.3.8 JavaScript

JavaScript (JS) is a dynamic computer programming language. Common use of Javascript is to build client-side scripts to enhance user interaction, browser controlling, asynchronously communication, and modify the document content that is displayed. Javascript could be used in server-side network programming (with Node.js), game development and development of desktop and mobile applications [44, JavaScript].

Although JavaScript adopts many naming conventions from Java, the two languages have minimum relationship. Object-oriented, imperative, and functional

programming styles are some basic features of JavaScript. Additionally JavaScript can be used outside of web pages - for example, in PDF documents, site - specific browsers, and desktop widgets which is equally inportant. Nowadays JavaScript VMs and platforms can be used to build server-side web applications something that increased popularity JavaScript even more.

3.3.8.1 Web Audio API

Audio on the web has been in early stages so far and until very recently plugins such as Flash and QuickTime were required in order to be delivered. The introduction of the audio element in HTML5 was a significant step for basic streaming audio playback. But, what if you want to develop more complex audio applications. For advanced web-based games or interactive applications, another approach is needed. Web Audio Api, is a Javascript API specification which aims to include the capabilities found in modern game audio engines as well as some of the mixing, processing, and filtering tasks that are found in modern desktop audio production applications [45, Web Audio API].

3.3.9 HTML5, JSP, XML

3.3.9.1 HTML5

HTML5 is a markup language which is utilized for organizing and presenting content for the internet. It is the fifth review of the HTML standard and since December 2012, a candidate suggestion of the World Wide Web Consortium (W3C). Its core proposes to improve the language with support for the latest multimedia while ensures readability by humans and comprehensibility by computers and devices (web browsers, parsers, etc)[46, HTML5].

HTML5 attempts to define a single markup language that can be written both in HTML or XHTML format, including detailed processing models to enhance interoperability. Also HTML5 extends, improves and rationalises the markup available for documents, and introduces markup and application programming interfaces (APIs) for complex web applications. Further more, HTML5 is a candidate technology for building cross-platform mobile applications. Many characteristics of HTML5 have been created to be able to execute on low-powered devices (for example smartphones and tablets).

More specifically, HTML5 introduces many new syntactic features. Some of them include the new <video>, <audio> and <canvas> elements, as well as the integration of scalable vector graphics (SVG) content (replacing generic <object> tags), and MathML for mathematical formulas. These features are helping develpers to include and handle multimedia and graphical content on the web pages without having to use non-free tools and software. Also new elements, such as <section>, <article>, <header> and <nav>, are designed to enhance semantic content of documents. Further more, some of the old elements of previous versions of HTML have been removed, deprecated or redefined such as <a>, <cite> and <menu>. The APIs and Document Object Model (DOM) have been placed in HTML5 specification. Finally HTML5 also takes care that syntax errors will be treated uniformly by all conforming browsers and other user agents, defining in some detail the required processing for invalid documents.

3.3.9.2 JSP

JavaServer Pages (JSP) is a technology which aims to help developers to build dynamically web pages based on HTML, XML, or others. At first it was released in 1999 by Sun Microsystems. JSP shares common features with PHP, but it uses

the Java programming language instead. In order to utilize JSP, a compatible web server, is required (for example as Apache Tomcat or Jetty)[47, JSP].

3.3.9.3 XML

Extensible Markup Language (XML) is defined as a markup language that consists a set of rules for formatting documents in order to be human-readable and machine-readable. The XML specifications are maintained by the World Wide Web Consortium (W3C). XML aims to maintain generality, simplicity, and usability in the Internet. Practically XML is a textual data format with strong support via Unicode for different human languages which is widely used for the representation of spontaneous data structures (a typical example is messages of web services).[48, XML].

3.3.10 CSS3

Cascading Style Sheets (CSS) is a style sheet language intending to describe the appearance and formatting of a document written in a markup language. Besides of styling of style web pages and user interfaces (usually written in HTML and XHTML), CSS3 can be used to any XML document, including plain XML, SVG and XUL. CSS is a state-of-the-art specification of the web and almost all web pages use CSS style sheets to portray their presentation [49, CSS3].

CSS is basically aims to separate document content from document presentation, including elements such as the layout, colors, and fonts. This separation leads to content accessibility improvement, more flexibility and control in presentation characteristics, enable several pages to share same format, is reducing complexity and enables repetition in the structural content.

CSS can also permit the same markup page to be presented in various styles for various rendering methods (for example on-screen, in print, by voice, or Braille-based tactile devices). Further more it can be utilized to display differently a web page depending on the screen size or device on which it is being viewed. Despite the fact that usually the developer of a document links that document to a CSS file, readers can utilize different style sheet, perhaps depending on their own computer, to override the one the author has specified. On the other hand, if the author or the reader did not link the document to a specific style sheet the default style of the browser then will be used. Also when more than one rule matches against a particular element, CSS specifies a priority scheme to determine which style rules will be applied. In this technique, priorities or weights are estimated and assigned to rules, so that the results are predictable. The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998), and they also operate a free CSS validation service.

Chapter 4

Analysis of implementation - Methodology

4.1 Client - Server model

The client - server model of computing is a distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients. Often clients and servers communicate over a computer network on separate hardware, but both client and server may reside in the same system. A server host runs one or more server programs which share their resources with clients. A client does not share any of its resources, but requests a server's content or service function. Clients therefore initiate communication sessions with servers which await incoming requests [50, Client - Server model]. Examples of computer applications that use the client - server model are Email, network printing, and the World Wide Web.

4.1.1 Server analysis

Server is responsible to check visitor's credentials. He is sending queries to our database and determines if login data are correct. If login data are correct, then user role is specified. Our system supports several levels of security according to four kind of roles: role of gamer, role of administrator, role of secure user and role of tester. Each role interact with the system in different way. Depending of visitor roles server is displaying predefined web pages. For administrator role he is displaying administrator pages, for gamer role he is displaying gamer pages, for secure user secure user web pages and for tester role web pages of tester. If login data are not correct, visitor have to provide system with correct data or to register. Additionally, if visitor is not remembering his credentials can retrieve them using his email. Finally, server is receiving queries with gamers score's from each game, saves them in database and sending them back to client as XML messages Figure 4.1.

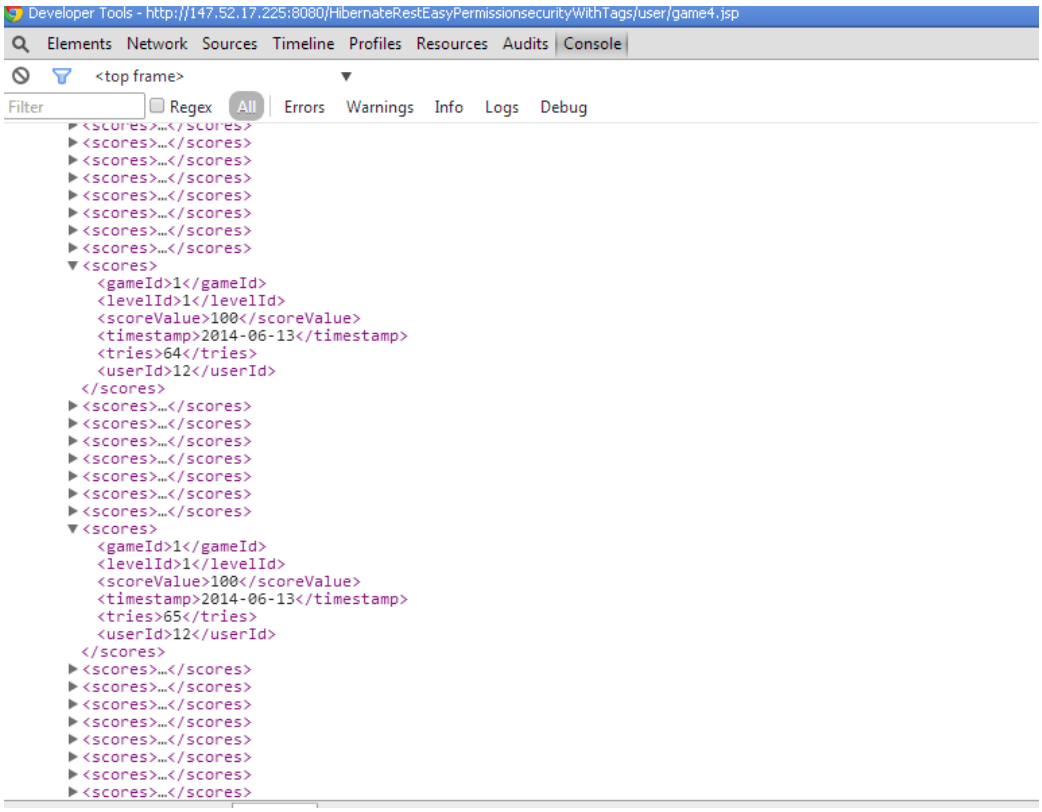


Figure 4.1: XML data messages

4.1.2 Client analysis

In client side takes place the interaction of visitor with our application. Depending on his role this interaction varies. A description of each role is given in the list below

- Role of administrator is responsible for user management. He can add, delete, edit, search for a user in our user database.
- Role of Gamer is our basic role. Server is displaying appropriate game pages to users. He can practice with our game collection, save and retrieve his scores to system database and finally study his performance through system's statistical graph data.
- Role of secure and role of tester also provides us with two extra levels of security. We are not going to focus to these extra roles as their are out of the scope of this thesis.

Also in client side our game collection exists. Games are executed in a browser as Java applets and/or as JavaScript applications. In both forms, games are taking input from microphone, estimating sound parameters, in our case pitch and intensity, calculating score values and then they are sending scores to server as HTTP requests. Finally client, receives all score values through time from server as XML message, parse it and displays embedded graphs in Web Page for further studying.

For the purposes of our thesis we are going to describe each task and sequence of actions that are supported of our system in terms of UML for better representation [51, UML]. Each service that is provided will be analyzed. Further more, the structure of our system is being presented through UML diagrams. We provide

description of our system's actors, entities that take part in our system, relationships between them, deployment diagrams, class diagrams, components diagrams, activity diagrams and sequence diagrams.

4.1.3 Actor description

For our system description we match each role with an actor with same responsibilities.

4.1.4 Use case diagrams

The use case diagram of our system is shown below. We are displaying each actor and it's possible use cases. Also we provide a short description, trigger conditions, goals, preconditions and failure states of each use case respectively to each role.

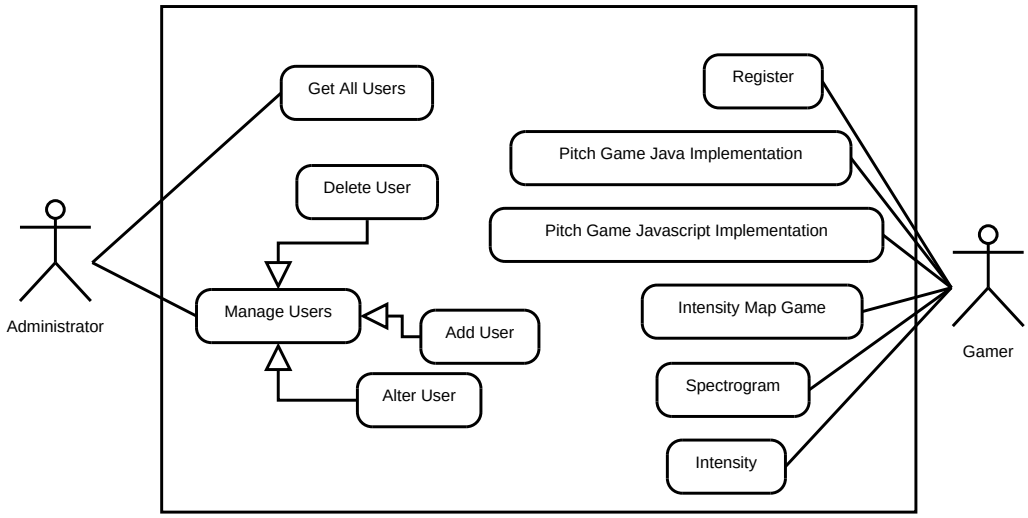
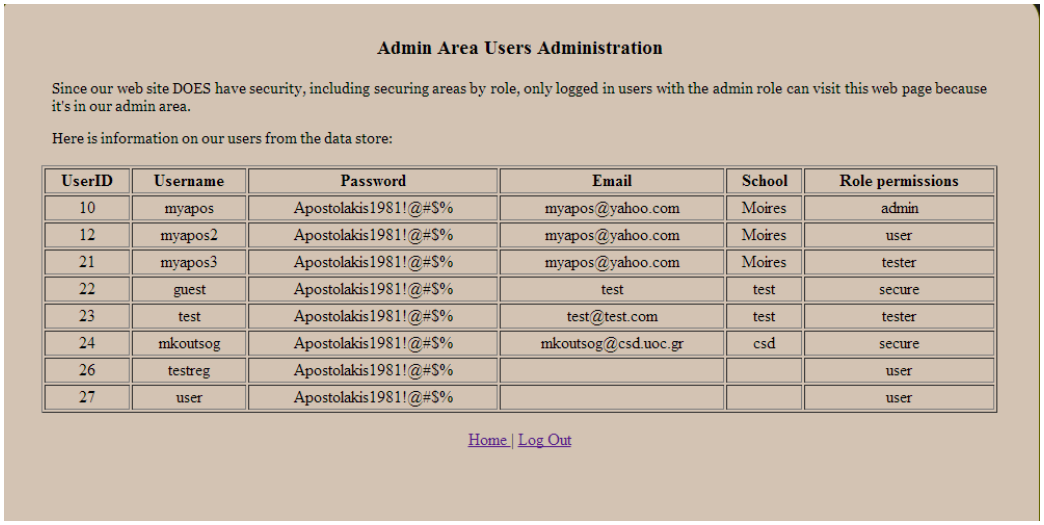


Figure 4.2: Use Case Diagram

Use case preconditions: For all use cases an internet connection has to exist between client and server. Also, the visitor has to enter his credentials or to register in our system. Finally, for spectrogram option and JavaScript applications the user has to run game applications with Google Chrome browser. For Java Game the visitor has to install Java in his System with our certification for enabling microphone access.

Administrator:

- Get All Users: Visitor of web site enters administrator data. Then server displays administrator's web pages to visitor. Administrator can review all user data from server's database. After checking data administrator can return to home page or logout from web site.
- Manage Users: Visitor of web site enters administrator data. Then server displays administrator's web pages to visitor. Administrator gets access to user data. He can modify user data, add and delete user. These operations are described below:
  - Add user: Administrator enter's user's data that he desires to create and press "Add User" button. If user does not exist in database then a new user is created with the role that is selected. Success page of adding user is displaying. If the user already exists, then the user has to try again and choose different username. Failure of adding web page is displayed



Admin Area Users Administration

Since our web site DOES have security, including securing areas by role, only logged in users with the admin role can visit this web page because it's in our admin area.

Here is information on our users from the data store:

UserID	Username	Password	Email	School	Role permissions
10	myapos	Apostolakis1981!@#$%	myapos@yahoo.com	Moires	admin
12	myapos2	Apostolakis1981!@#$%	myapos@yahoo.com	Moires	user
21	myapos3	Apostolakis1981!@#$%	myapos@yahoo.com	Moires	tester
22	guest	Apostolakis1981!@#$%	test	test	secure
23	test	Apostolakis1981!@#$%	test@test.com	test	tester
24	mkoutsog	Apostolakis1981!@#$%	mkoutsog@csd.uoc.gr	csd	secure
26	testreg	Apostolakis1981!@#$%			user
27	user	Apostolakis1981!@#$%			user

Home | Log Out

Figure 4.3: Get All Users printscreen

from server. After adding anew user to the database, administrator can return to home page or logout from web site.

- Delete user: The administrator has to know the username of user. He enters user's username and presses "Delete User" button. A delete operation in our database is happening. If everything is ok then a success web page is displayed. If the user does not exist in the web page, the administrator has to try again. After deleting the user from the database administrator can return to home page or logout from web site.
- Alter user: The administrator has to know the username of user. He enters user's username and presses "Alter User" button. An alter operation in our database is happening. If everything is ok then a success web page is displayed. If the user does not exist in the web page, administrator has to try again. After altering the user's data from database, administrator can return to home page or logout from web site.

Gamer:

- Pitch Game Java Implementation: The visitor of web site enters Gamer's data. In next step he selects Pitch Game with Java implementation. After that he selects pitch from pop up list. This value is the pitch that Gamer wishes to train with. When he selects pitch and presses OK, then graphical user interface of pitch game is displayed. The Gamer has to try to land the starship on the asteroid only by changing the pitch of his voice. The starship is looping over the space until starship lands on asteroid. If the Gamer succeeds then a second pop up window displays with several options. Gamer can select to play again, stop or study graph results. In each option, game is executing again or Gamer can logout of our system. If he selects to see graph results then the client is communicating with server in order to send his score. Score for a single game execution is calculated with the following equation, where "numberOfTries" is the number of passages of starship through screen.

$$score = 100/numberOfTries \tag{4.1}$$

After the Gamer's scores is sent to the server, server responds with all previous scores that he achieved from the beginning of his registration in our system. Because we want to display performance per day, average value of

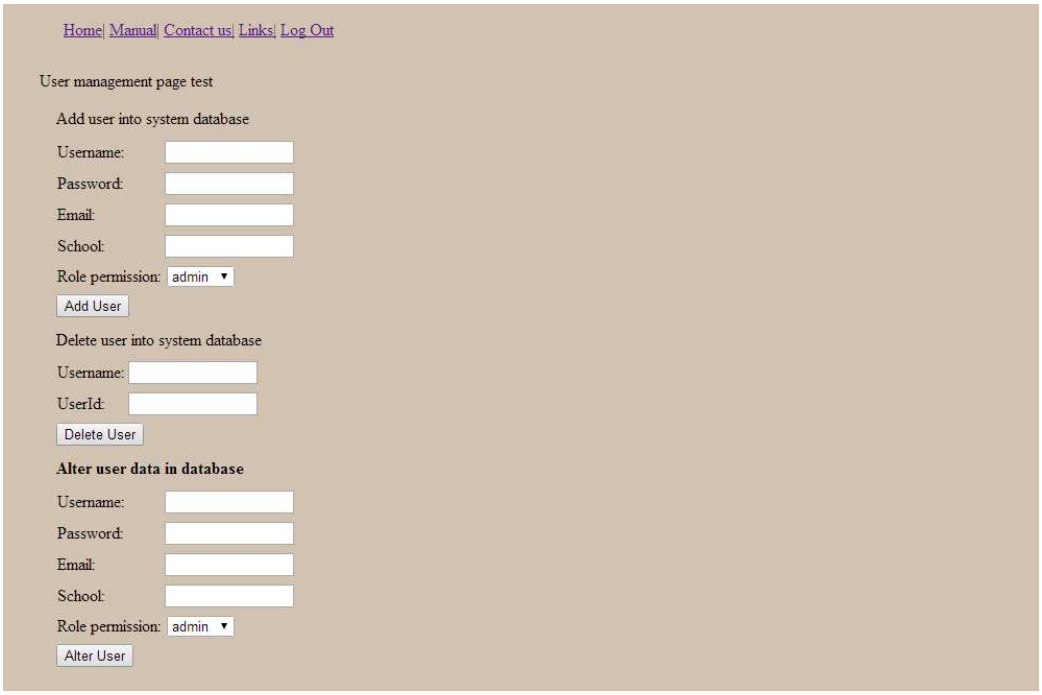


Figure 4.4: Manage users printscreen

game executions per day is calculated. As final step, average values per day are being displayed as a pop up window to the Gamer (Figure 4.6).

- Pitch Game JavaScript Implementation: The visitor of the web site enters Gamer's data. In next step he selects Pitch Game with JavaScript implementation. After that, the system asks the Gamer to allow access to microphone. Next the Graphical User Interface is displayed where the Gamer has several options. He can drag and drop the asteroid to position he wishes to train. Position of asteroid stands for pitch height. However, the Gamer can calibrate maximum and minimum pitch that he produces with his voice. This option is implemented as an extra feature in order to cover all varieties of the Gamer's voice. Usually children's have a more high frequency voice from adults. Despite this fact, an adult Gamer can use our system too with this option. Predefault values are introduced. After setting game's configuration the Gamer can actually play pitch game and try to land the starship on the asteroid only by changing the pitch of his voice. Same scenario as in Java implementation exist here too. Score is calculated respectively to equation 4.1. After landing starship on the asteroid, client is sending the score of game to the server and the server responds with the score values of old game executions. Average value per day is also calculated and results are displayed in an embedded graph in our web page. After studying the graph results he can logout from system (Figure 4.7).

- Intensity Game: The visitor of the web site enters the Gamer's data. In next step he selects the Intensity Game. After that, the system asks the Gamer to allow access to microphone. Next the Graphical User Interface is displayed where the Gamer has several options. He can drag and drop the asteroid to the position he wishes to train. The position of asteroid stands for intensity value (Sound Pressure Level). Nevertheless, the Gamer can calibrate maximum and minimum SPL he can produce with his voice. This option is implemented as an extra feature in order to cover all the varieties of the Gamer's voice. Predefault values are introduced. After setting the

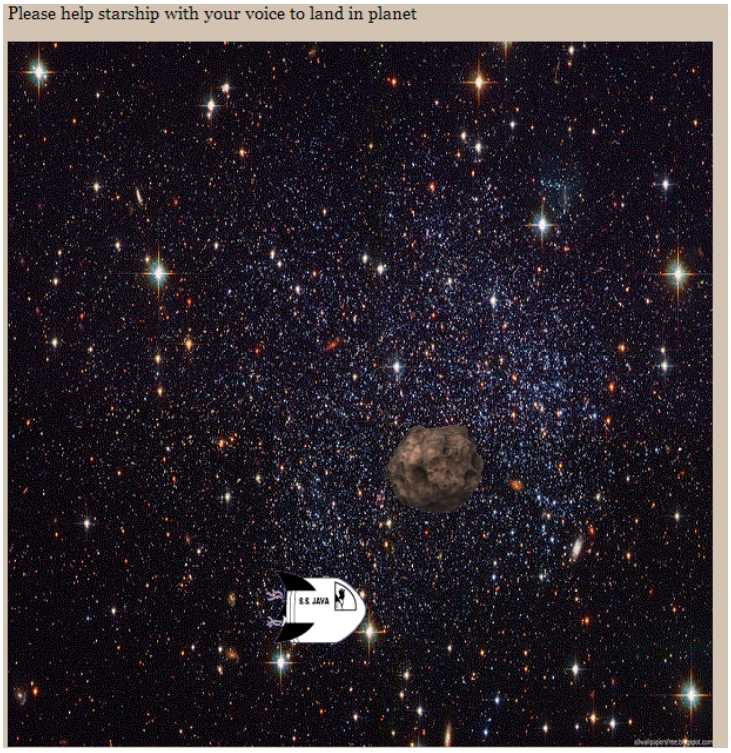


Figure 4.5: Pitch Game - Java implementation printscreen

game's configuration, the Gamer can actually play the intensity game and try to land the starship on the asteroid only by changing the loudness(SPL) of his voice. Same scenario as in pitch games exist here too. Also, score is calculated respectively to equation 4.1. After landing the starship on the asteroid, the client is sending the score of the game execution to the server and the server responds with score values of old game executions. Average value per day is also calculated and the results are displayed in embedded graph in our web page. After studying graph results he can logout from system (Figure 4.8).

- Intensity Map Game: The visitor of the web site enters the Gamer's data. In next step he selects the Intensity Game. After that, the system asks the Gamer to allow access to microphone. Next the Graphical User Interface is displayed where the Gamer has several options. The Graphical User Interface is consisting of three asteroids in different positions. Each position of asteroids stands for intensity value (Sound Pressure Level). Several combinations of asteroids in several predefined heights are available through the form of loudness exercises. The Gamer can select and practice with them. Also, he can calibrate with the maximum and the minimum SPL he can produce with his voice. This option is implemented as an extra feature in order to cover all varieties of Gamer's voice. Predefault values are introduced also. After setting game's configuration Gamer can actually play intensity map game and try to land starship on each asteroid only by changing the loudness(SPL) of his voice. In this case game scenario is different from previous games. Gamer has to land spaceship in each asteroid. Here, the spaceship is not looping over space. The spaceship is passing through the space just once. So the Gamer have only one try. Score is calculated respectively by the equation 4.2 where "scoreFactor" is the ratio "numberOfAsteroidsLanded" divided by "numberOfAsteroidsInMap"

Figure 4.6: Player performance - Pitch game Java implementation printscreen

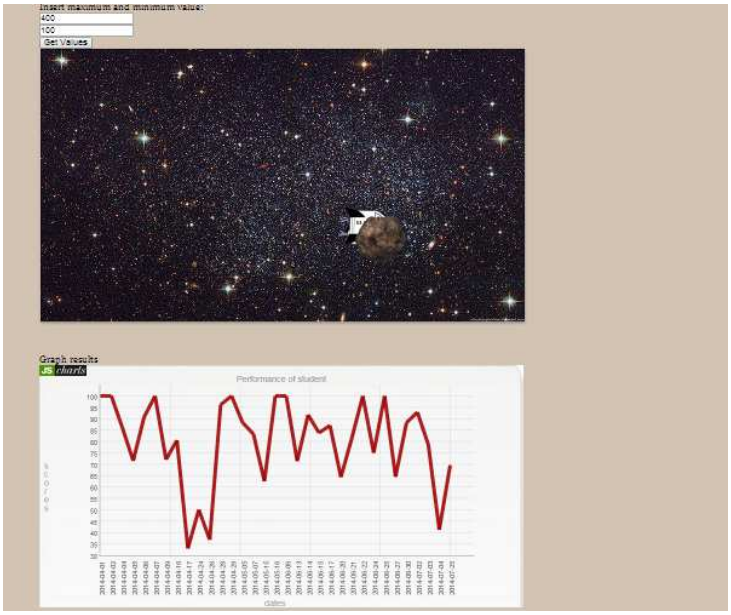


Figure 4.7: Pitch Game - JavaScript implementation printscreen

$$score = 100 * scoreFactor \tag{4.2}$$

After score calculation, the client is sending the score of the game execution to the server and the server responds with the score values of the old game executions. the average value per day is also calculated and the results are displayed in an embedded graph in our web page. After studying the graph results he can logout from system (Figure 4.9).

- Spectrogram: The visitor of the web site enters the Gamer's data. After that, the system asks the Gamer to allow access to microphone. Next the Graphical User Interface is displayed the where Gamer has several options. The Graphical User Interface is consisting of special section where the spectrogram of Gamer's voice is being drawn. Also, there is another section where reference spectrogram for several vowels and consonants are being available. The Gamer has to produce several phonemes such as /α/, /ε/, /φ/, /ι/, /o/,

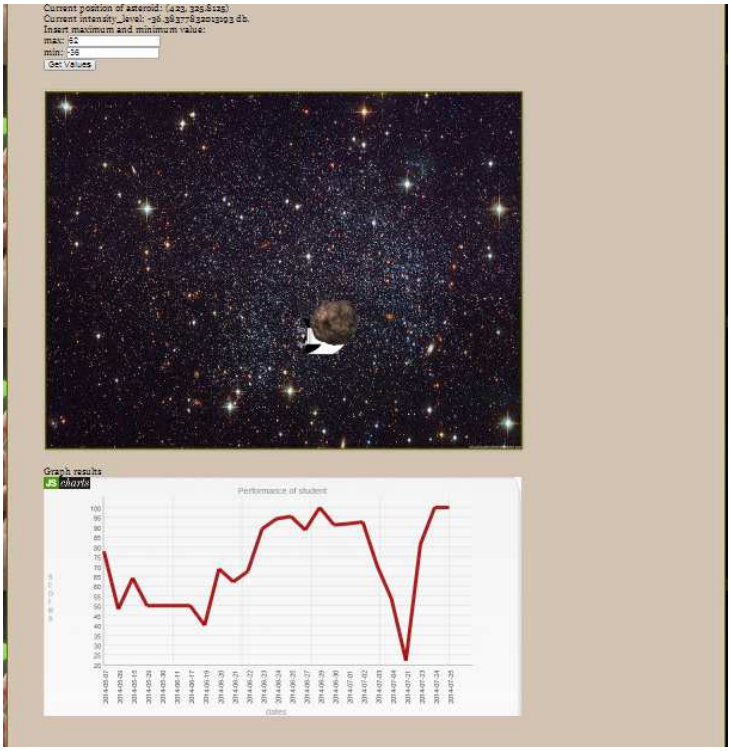


Figure 4.8: Intesity Game - JavaScript implementation printscreen

/o/, /υ/, /σ/, /ϑ/, /ζ/, compare them with the reference ones, and record his notes. Then can press "try again" for a new game effort. We have to note though that the whole process is better to be executed with the supervision of voice specialists in order to estimate differences in results and to instruct the Gamer how to pronounce the phonemes so that the Gamer's spectrogram matches the reference spectrogram. After comparing the spectrogram graph the Gamer can logout from the system (Figure 4.10).

4.1.5 Package diagram

4.1.5.1 Client package diagram

In this section we present package diagrams for client side (Java implementation).

In Figure 4.11 are displayed all packages who exist in client side and how they communicate with each other. Each package has different functionality and serves a different purpose. For instance, package Rocket is responsible to draw Graphical User Interface and to manage the animation of the spaceship. Also, is responsible for the communication with the server when a game target is accomplished. Similarly, package chart is used to display performance graphs to the user, and packages Loudness and PitchDetector to calculate intensity and pitch of voice input respectively.

4.1.5.2 Server package diagram

In this section we present package diagrams for server side.

In Figure 4.12 are displayed all packages who exist in server side, how they communicate with the client user interface and the database. Here is a short description of each package and it's functionality.

- Servlet. This package contains all necessary classes for user management. Supported actions are "add a user", "delete a user", "alter user's data",

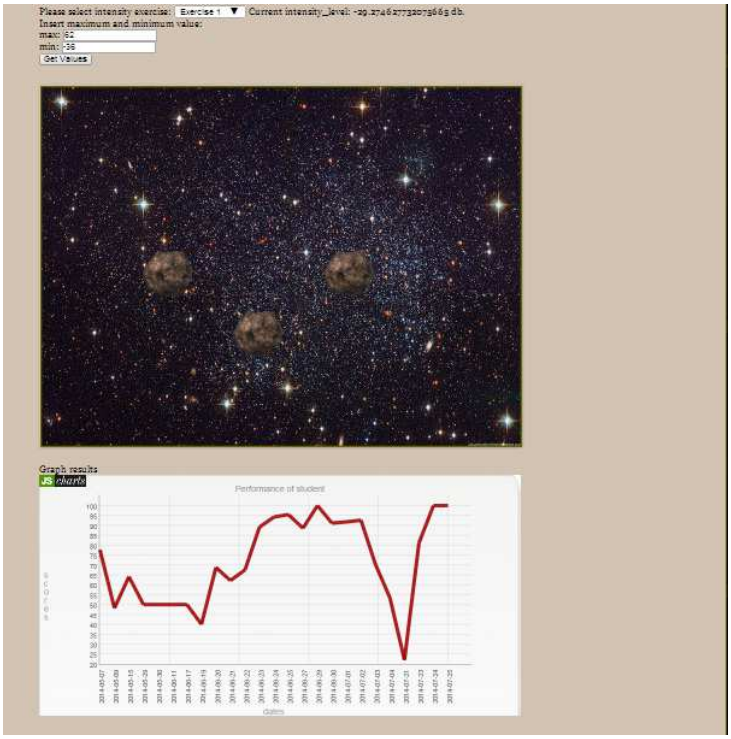


Figure 4.9: Intesity map Game - JavaScript implementation printscreen

"retrieve all user data from database", "send email" if someone has forgotten his password, "log in", "log out" and "register".

- Apache Shiro. This package contains all necessary classes for user data authentication and verification.

- RestEasy. This package contains all classes which are needed by the server in order to communicate with client. RestEasy package is an implementation of REST architecture and is used in order to deploy our RESTFul web service. Scores data are serialized in xml messages and are sent back to client side.

- HibernateModel. This package contains all classes which are needed by the server in order to model our E-R database schema using hibernate framework. All tables, constraints and relationships of our database are modelled. Hibernate is responsible to communicate with our database using HQL language. Hibernate offers to our application an extra layer of abstraction as we could replace easily our MySQL database with another, without having to modify our code.

Also, in server side exists our database schema which contains all of our data which are used by our web application. These are personal information of users, their scores etc. For more details see section "Database schema E-R diagram".

4.1.6 Class diagrams

In this section we present class diagrams for server and client side. Client side, refers to java implementation of pitch game. Every class diagram, represents classes and associations for each package. The classes who are depicted to have no associations, for each package either provide an independent functionality to the system so it is not required to be used from another classes or they are used by classes who belong to different packages. Finally, we provide general class diagrams

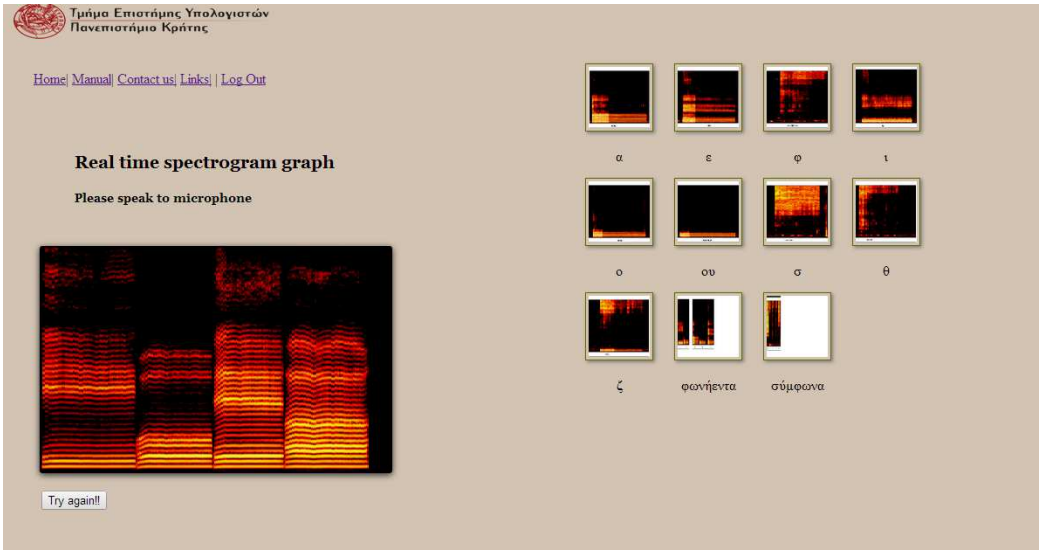


Figure 4.10: Real time spectrogram printscreen

where it is represented every association between all classes of our system regardless of what package every class belongs to.

4.1.6.1 Client class diagram

In this section we present class diagrams for client side (Java implementation - pitch game) .

In Figure 4.18 are displayed all classes from all packages which are used in client side. Each class diagram for each package are described in next diagrams.

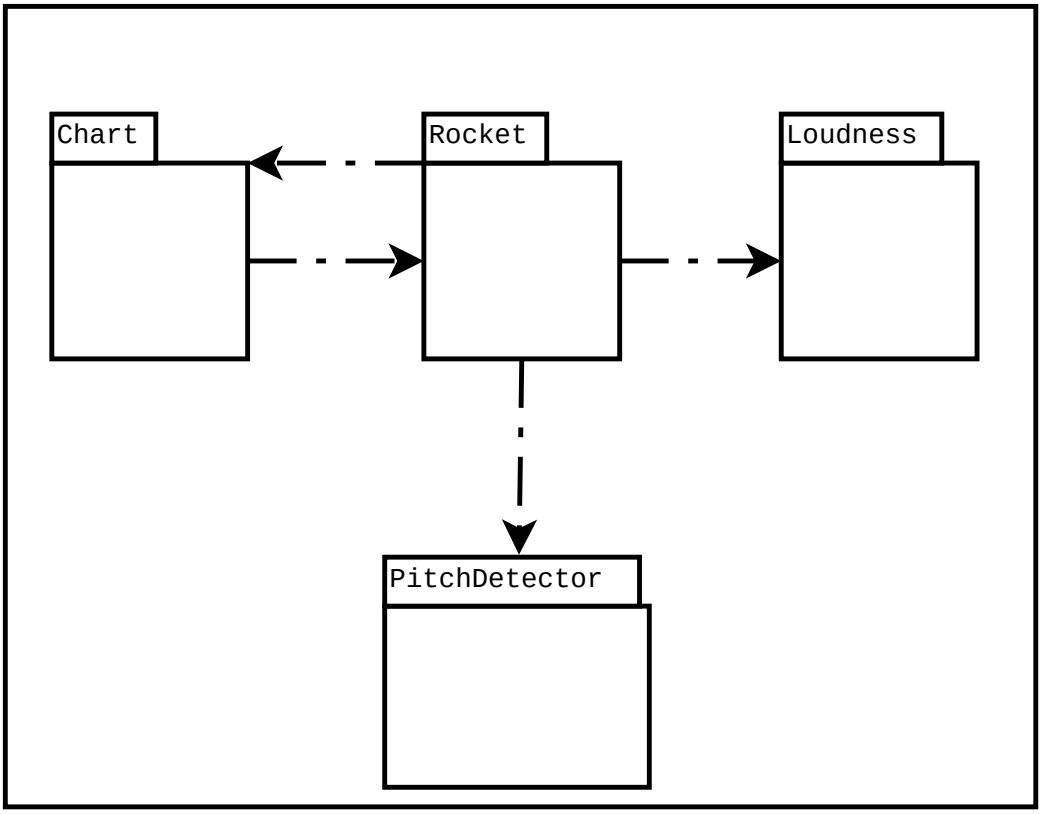


Figure 4.11: Client Package Diagram

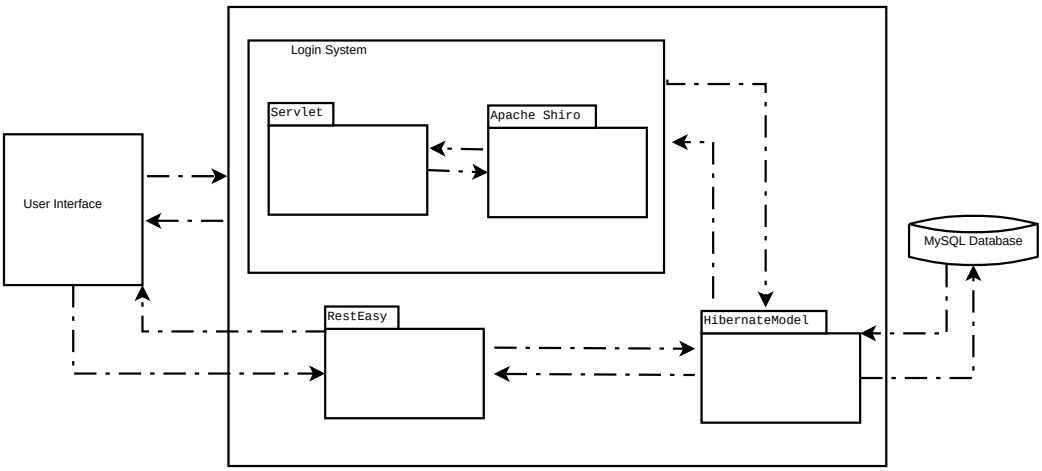


Figure 4.12: Server Package Diagram

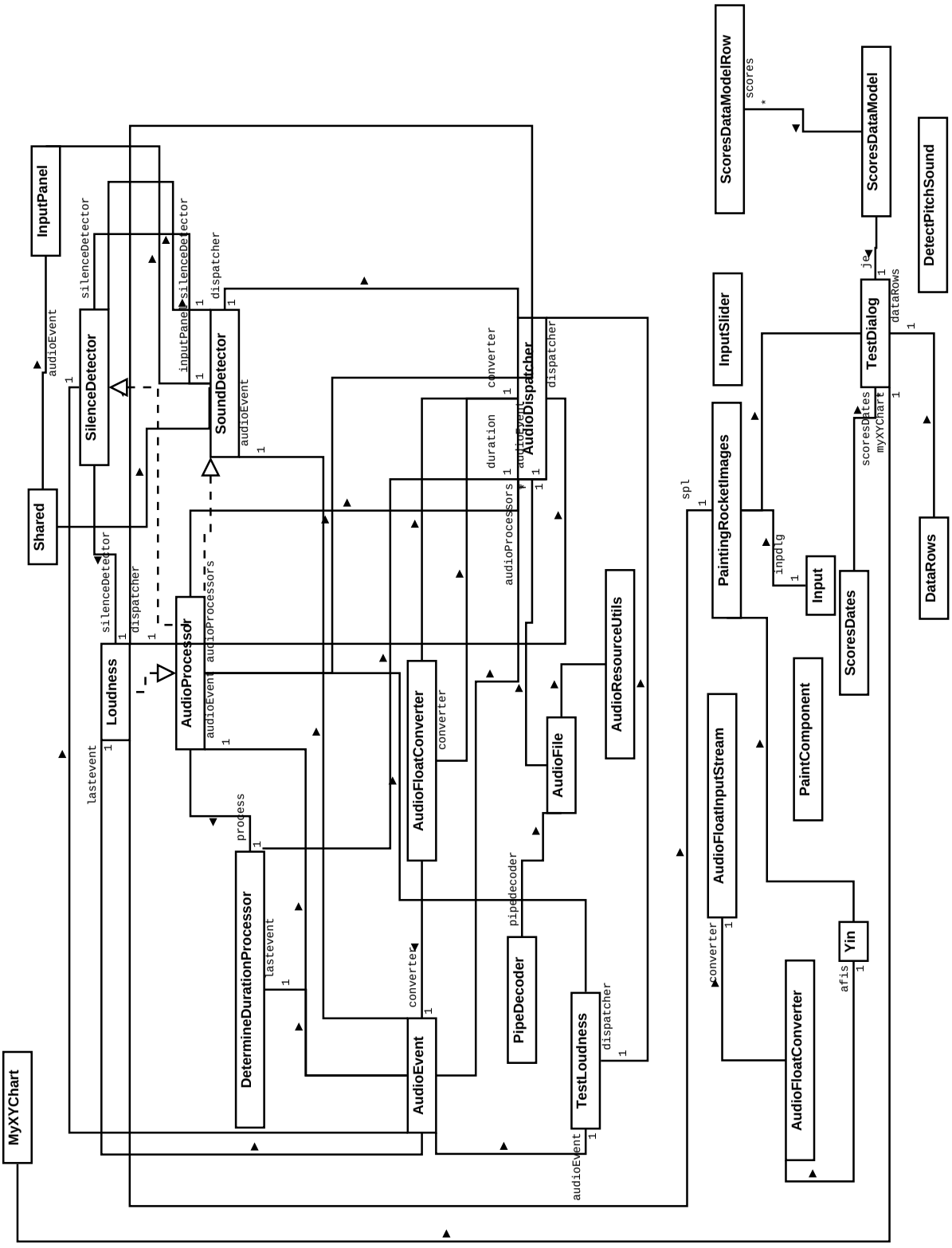


Figure 4.13: General Client Class Diagram

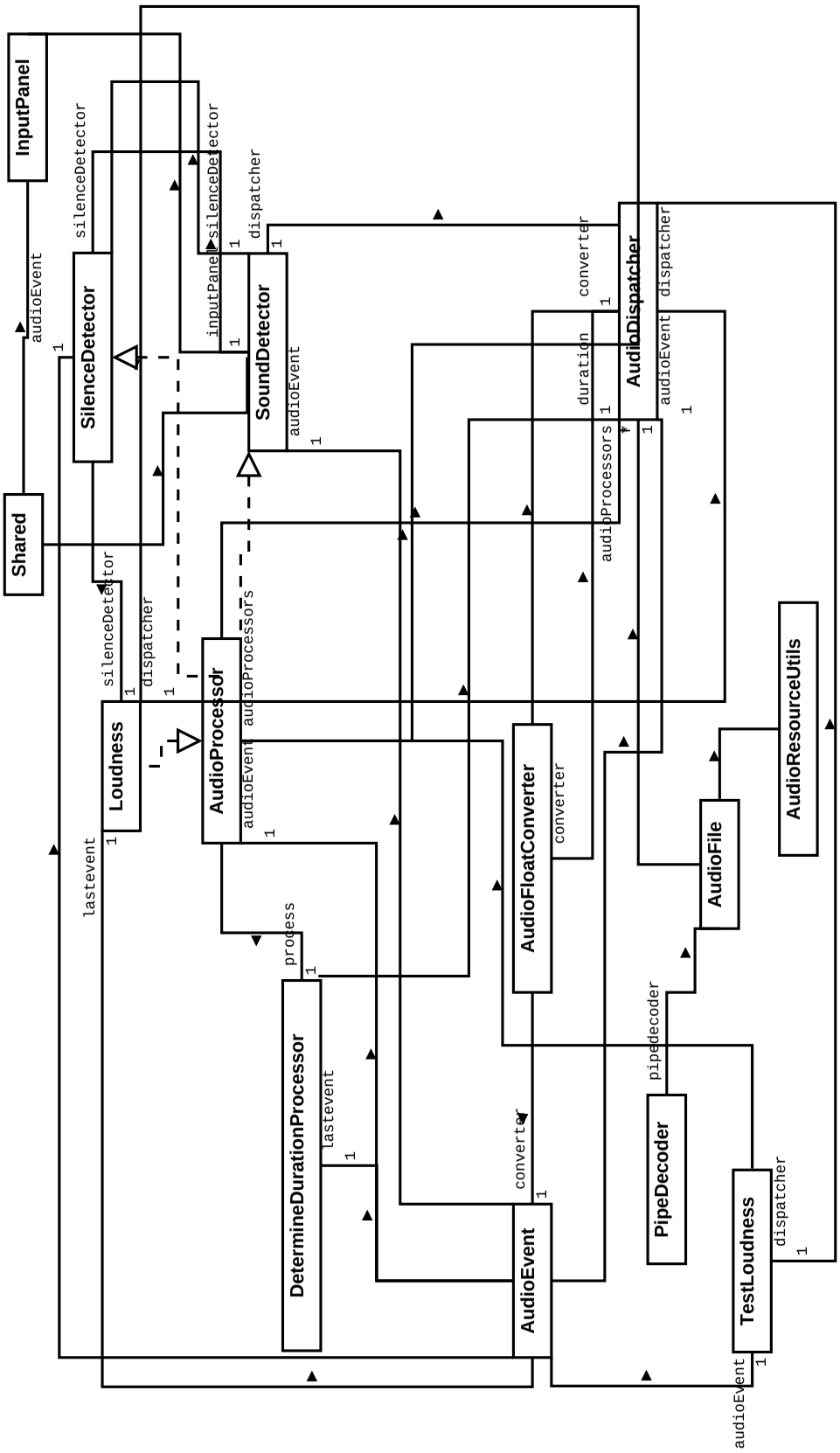


Figure 4.14: Loudness Class Diagram

In Figure 4.14 are displayed all classes from package "Loudness". These classes are used to calculate intensity of input signal. Basically, the sound pressure level (SPL) of input is measured. If the result of this calculation is below a threshold, which we have already defined, then very low energy signals (noise) from the environment is ignored [52, Tarsos].

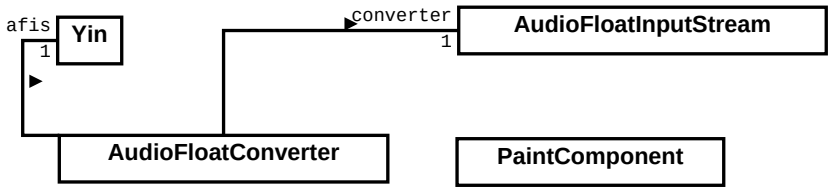


Figure 4.15: Pitch Detector Class Diagram

In Figure 4.15 are displayed all classes from package "PitchDetector". These classes, are used to calculate pitch of input signal. Input from microphone is received and processed according to YIN algorithm in order to calculate pitch [52, Tarsos].

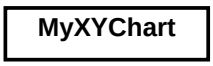


Figure 4.16: Chart Class Diagram

In Figure 4.16 are displayed all classes from package "myXYChart". These classes use JFreeChart library in order to produce performance chart. In the beginning, scores data are received as input from server. Next, data are processed and are displayed in performance graph .

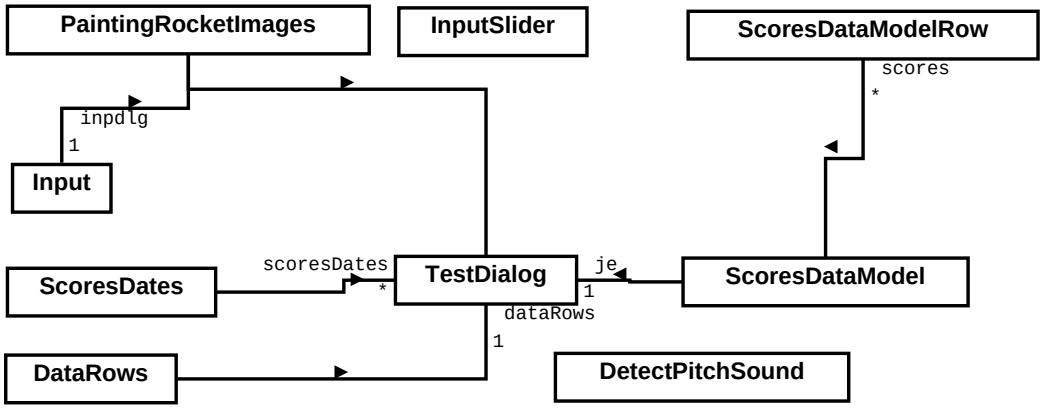


Figure 4.17: Rocket Class Diagram

In Figure 4.17 are displayed all classes from package "RocketClassDiagram". These classes, are responsible for Graphical User Interface management and for the animation of spaceship. They consist the core package of our system. Furhtermore, we can note that the class ScoresDataModel uses many objects of class ScoresDataModelRow. These classes, are used to save scores data, which are received from the server. In next step, data are sent to the package MyXYChart for graph production and display.

4.1.6.2 Server class diagram

In this section we present class diagrams for server side per package.

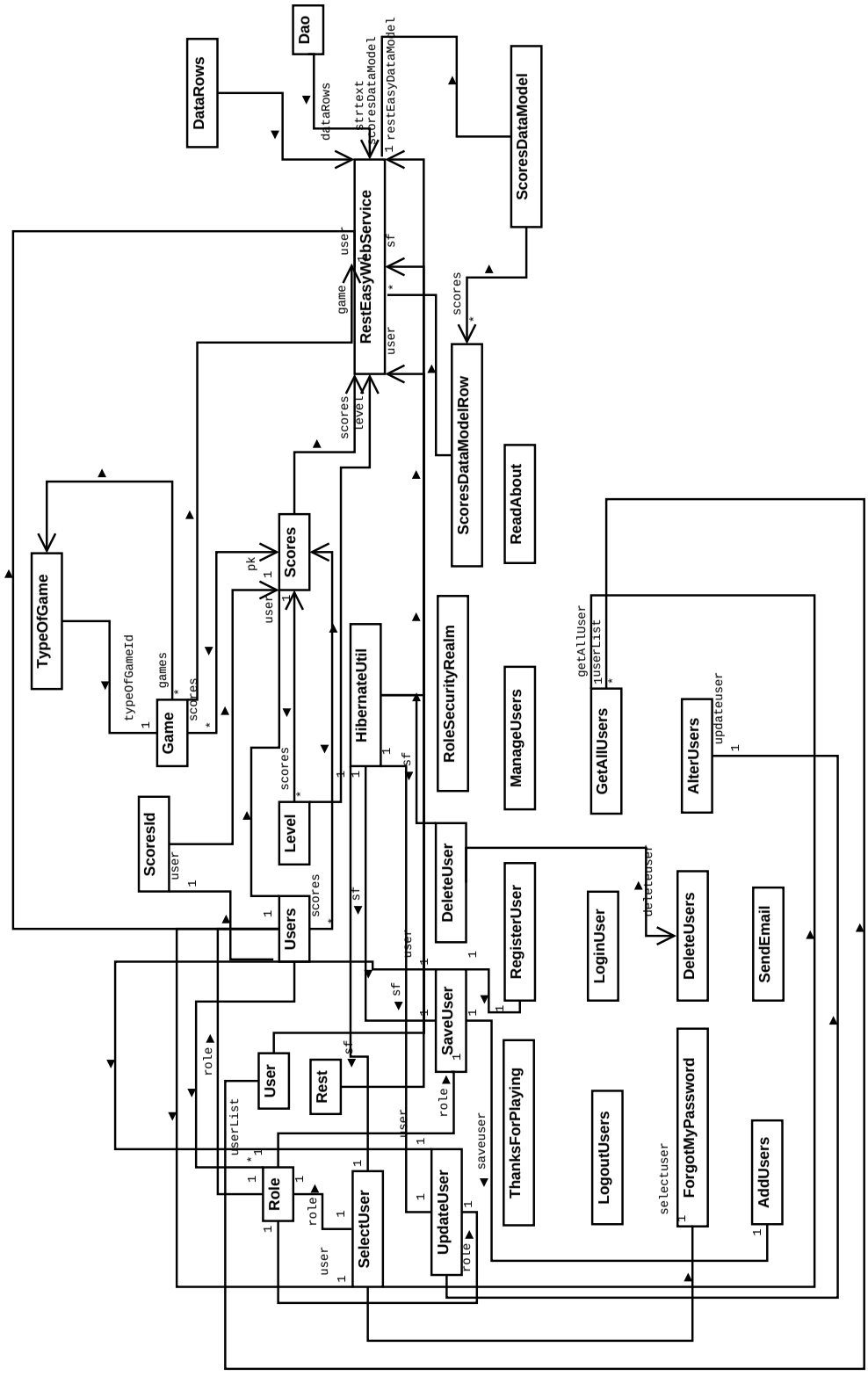


Figure 4.18: General Server Class Diagram

In Figure 4.18 are displayed all classes from all packages which are used in server side. Each class diagram for each package are described in next diagrams.

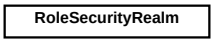


Figure 4.19: ApacheShiro Class Diagram

In Figure 4.19 are displayed all classes from package "ApacheShiro". These classes are used to authenticate and authorize user of our system. RoleSecurityRealm uses hibernate in order to retrieve user's credentials, decide the role of user and display the right web pages of our web application.

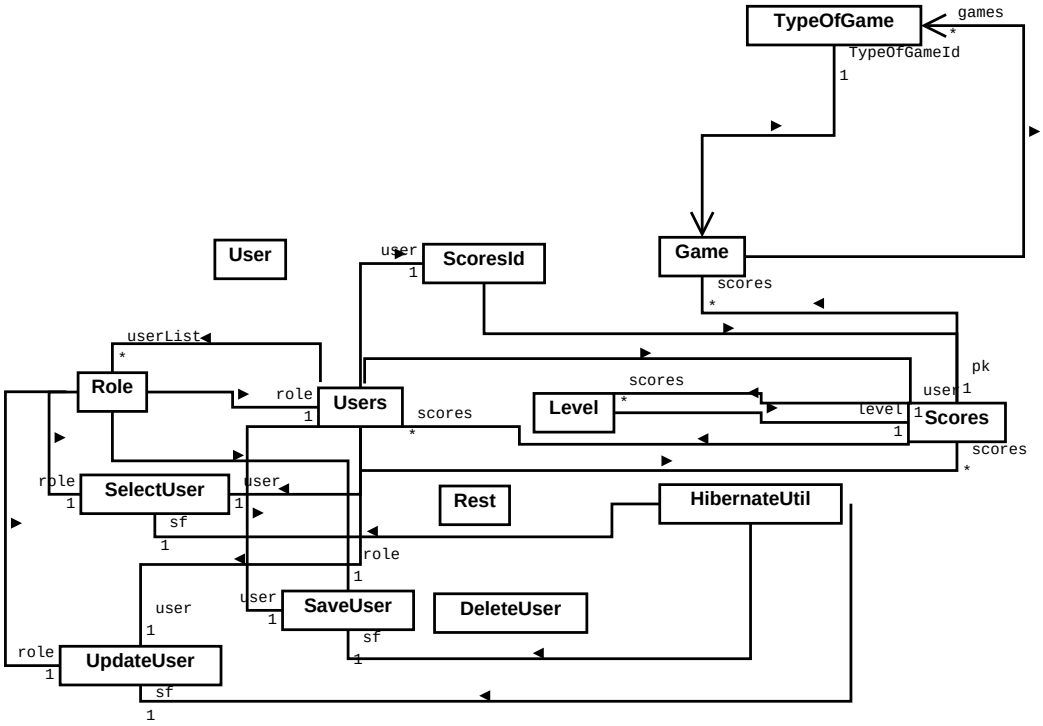


Figure 4.20: Hibernate Model Class Diagram

In Figure 4.20 are displayed all classes from package "HibernateModel". These classes are used to map entities and relationships of our database schema with Java objects. As we mentioned in previous sections, hibernate and HQL offers to us an extra layer of abstraction because it gives the developer the capability to choose free the type of database (Derby, Oracle etc) he wants to use with minimum effort and modifications of the system.

In Figure 4.21 are displayed all classes from package "RestEasy". These classes enhance server with the capability to receive and to respond to http requests. The HTTP requests include information from client such as user id, game id and scores. The scores are saved to database using hibernate and after that server is responding to client with scores of old game executions in the form of XML messages. The client receives the XML messages, processes them and displays to the user his performance graphs.

In Figure 4.22 are displayed all classes from package "Servlet". Supported actions are "add a user", "delete a user", "alter user's data", "retrieve all user data from database", "send email" if someone has forgotten his password, "log in", "log out" and "register".

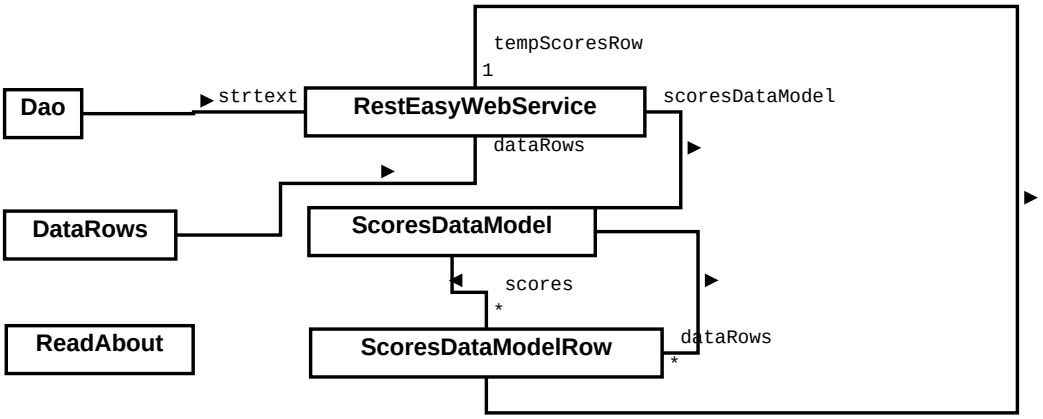


Figure 4.21: RestEasy Class Diagram

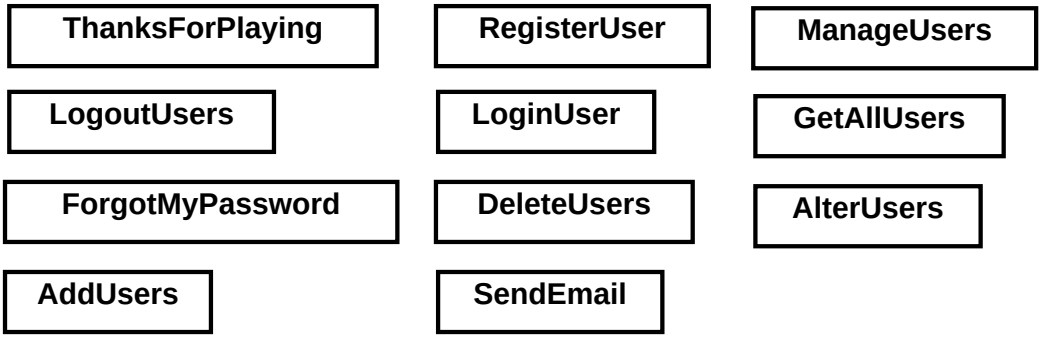


Figure 4.22: Servlet Class Diagram

4.1.7 Activity diagram

4.1.7.1 Client activity diagram

In this section we present activity diagrams for client side (Java implementation). No further description is provided as they are quite informative and self explanatory.

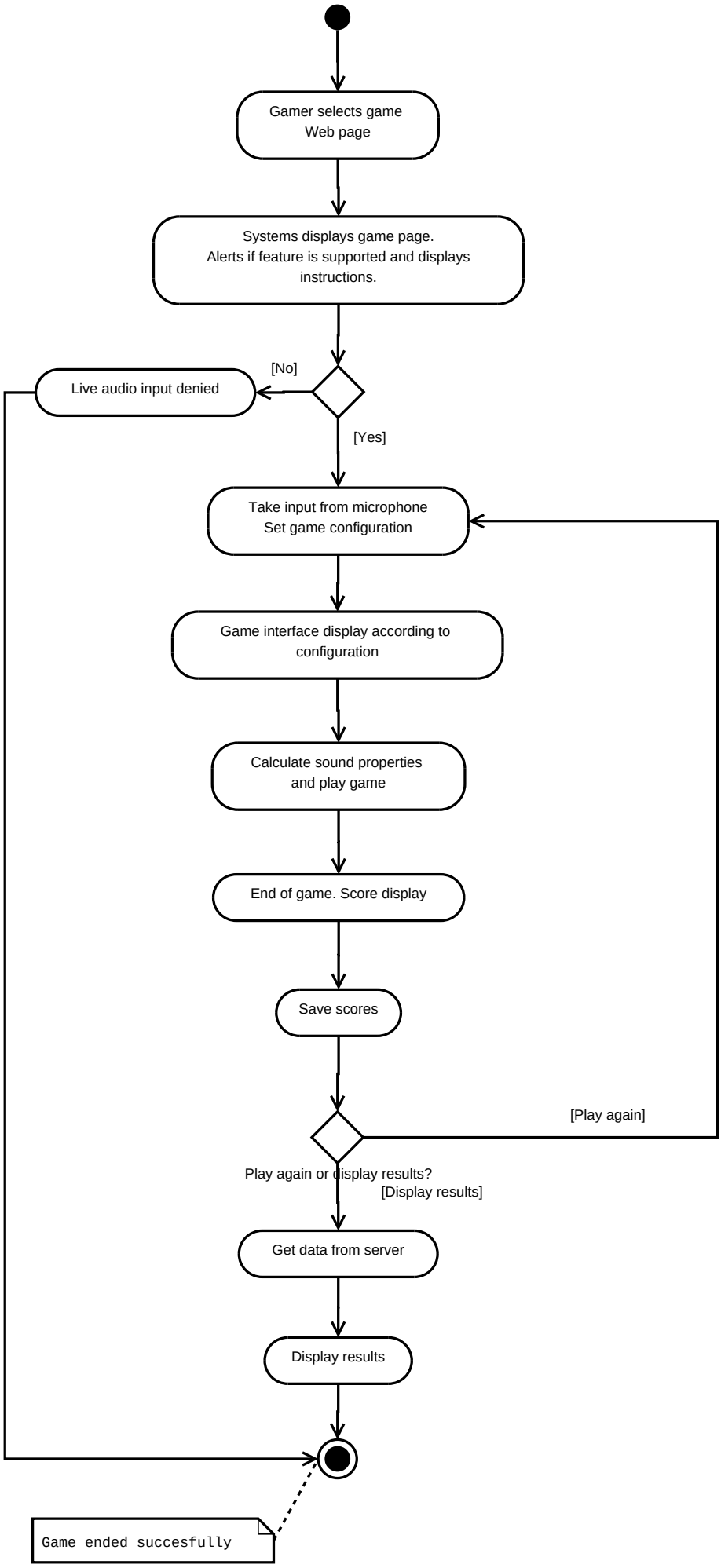


Figure 4.23: Game activity diagram

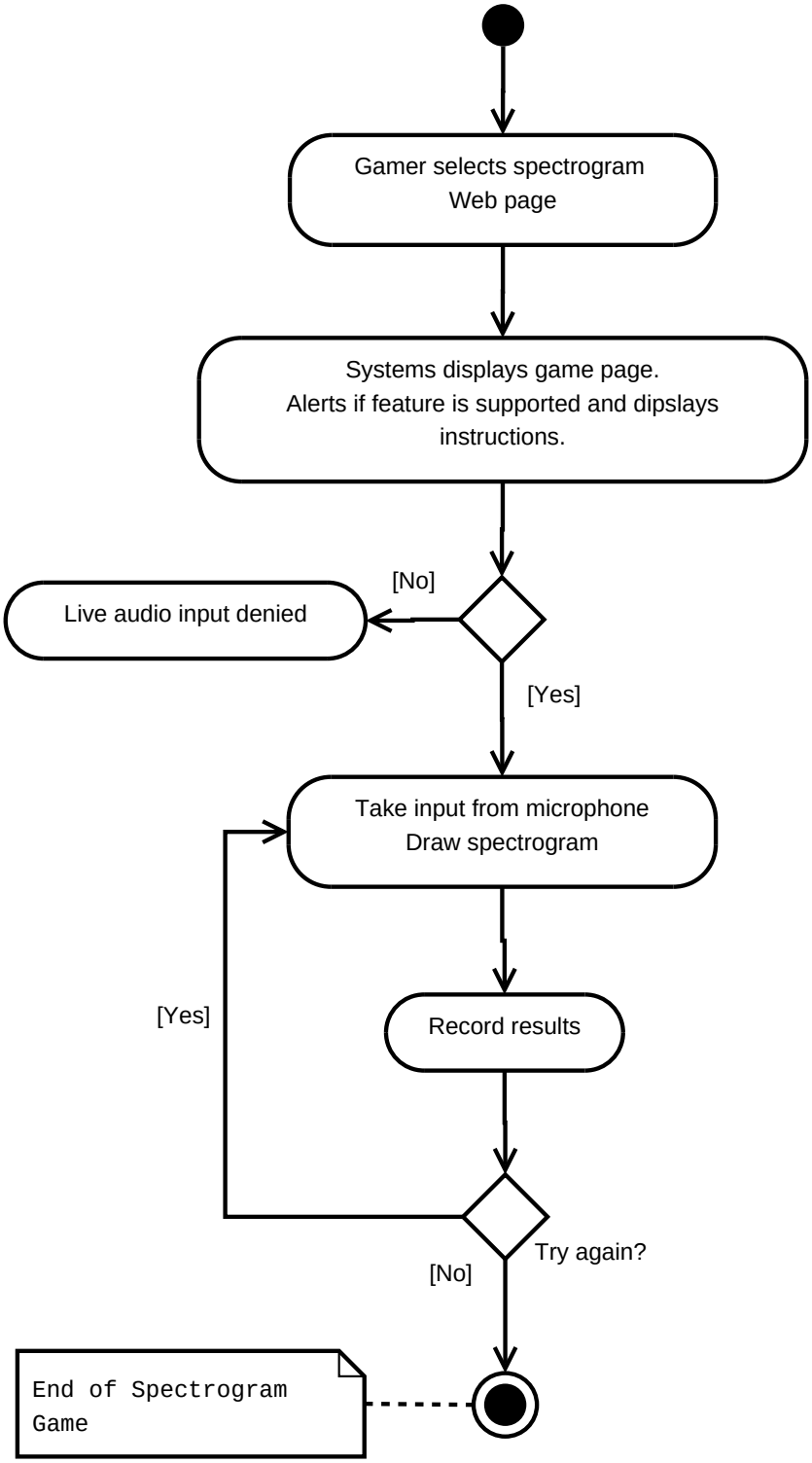


Figure 4.24: Spectrogram activity diagram

4.1.7.2 Server activity diagram

In this section we present activity diagrams for server side. No further description is provided as they are quite informative and self explanatory.

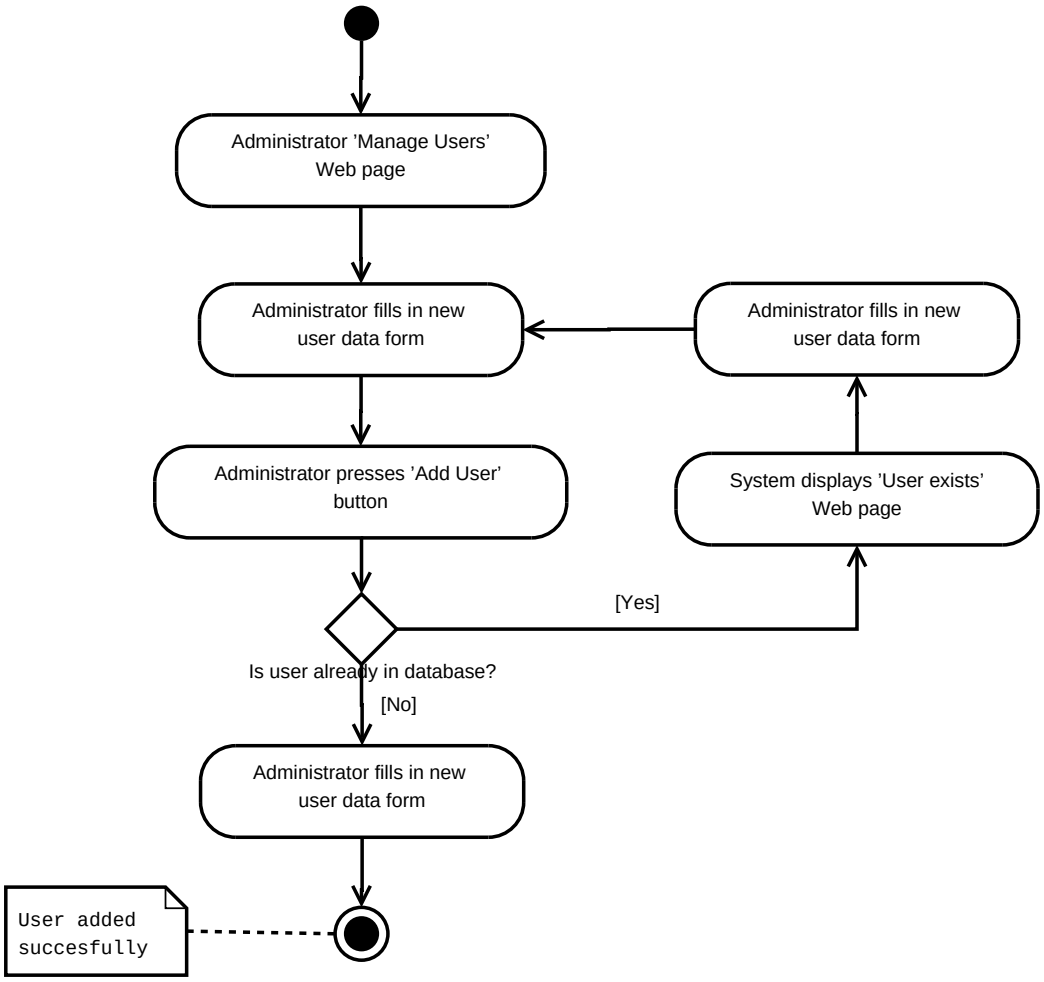


Figure 4.25: Add user activity diagram

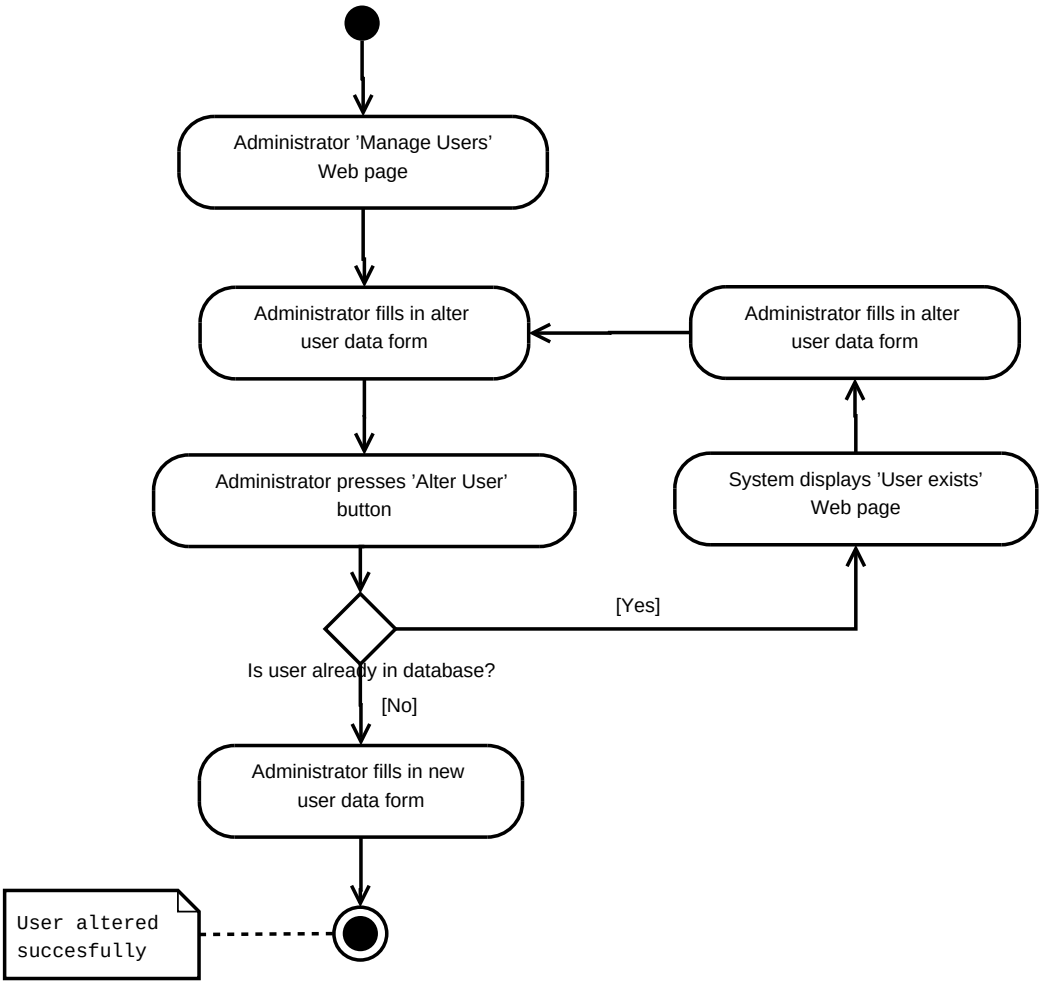


Figure 4.26: Alter user activity diagram

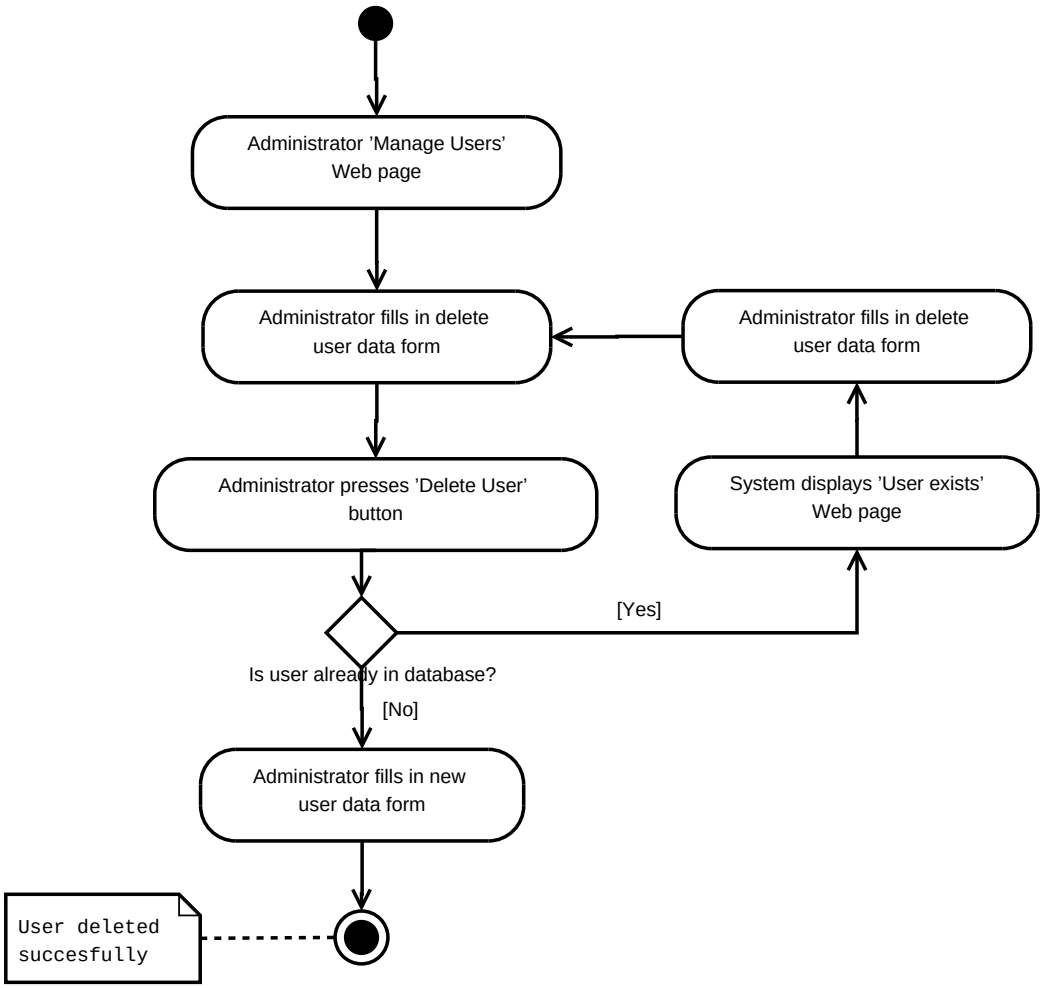


Figure 4.27: Delete user activity diagram

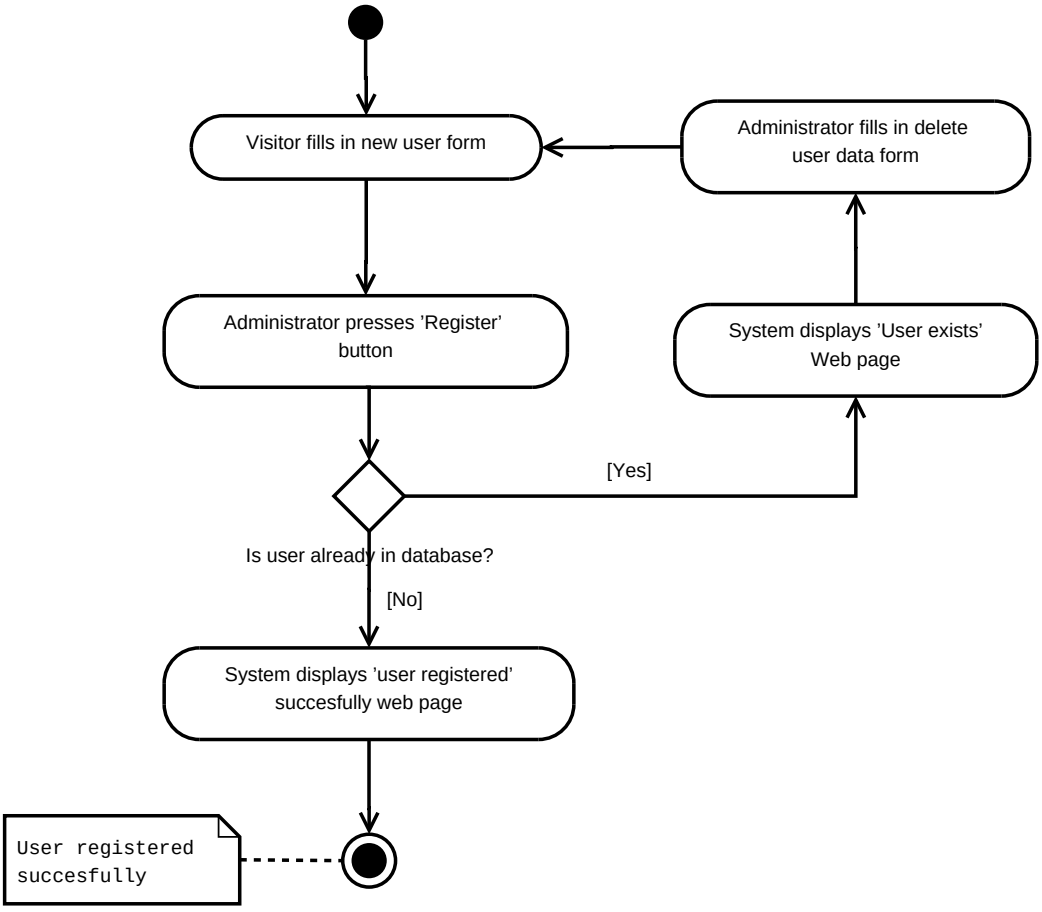


Figure 4.28: Register user activity diagram

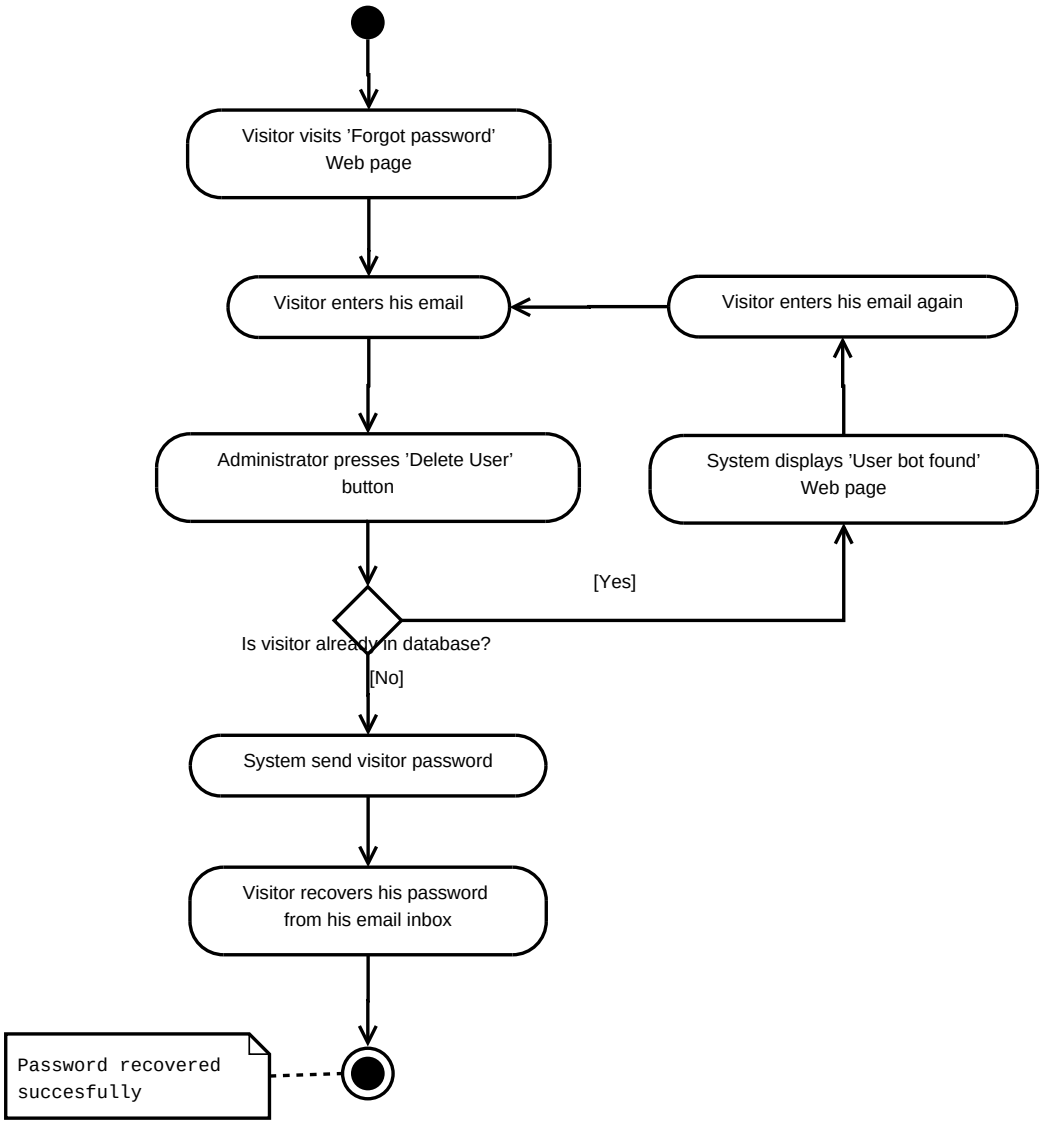


Figure 4.29: Forgot data activity diagram

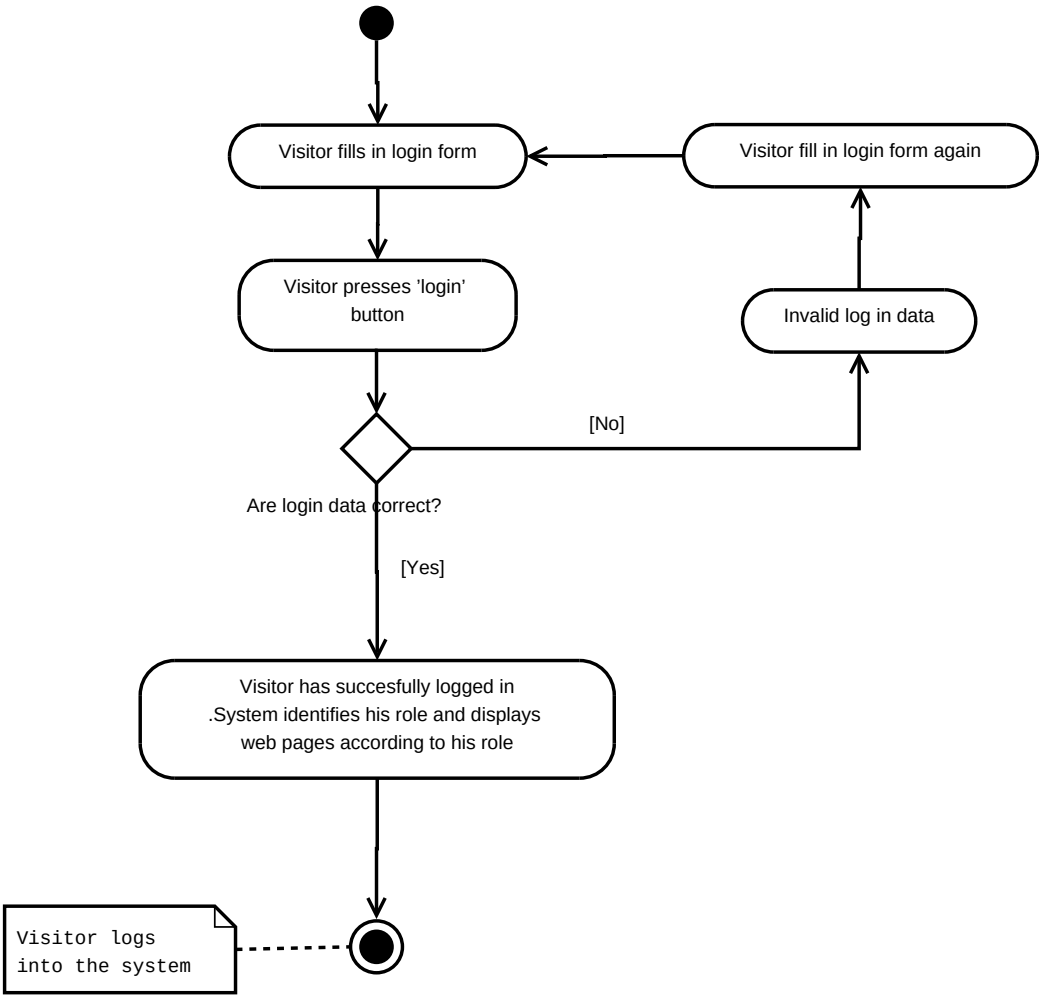


Figure 4.30: Login activity diagram

4.1.8 Sequence diagram

4.1.8.1 Client Sequence diagram

In this section we present sequence diagrams for client side (Java implementation). No further description is provided as they are quite informative and self explanatory.

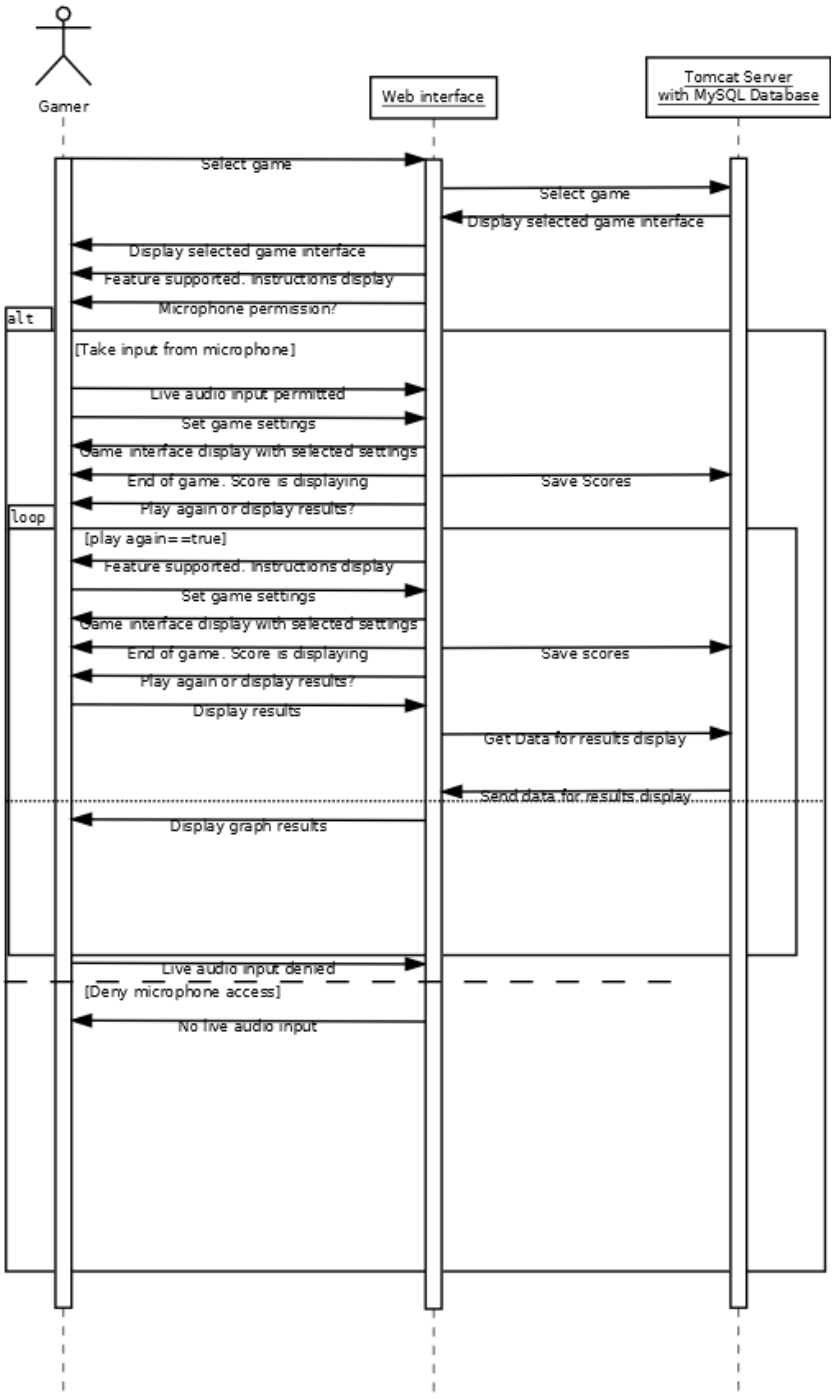


Figure 4.31: Game sequence diagram

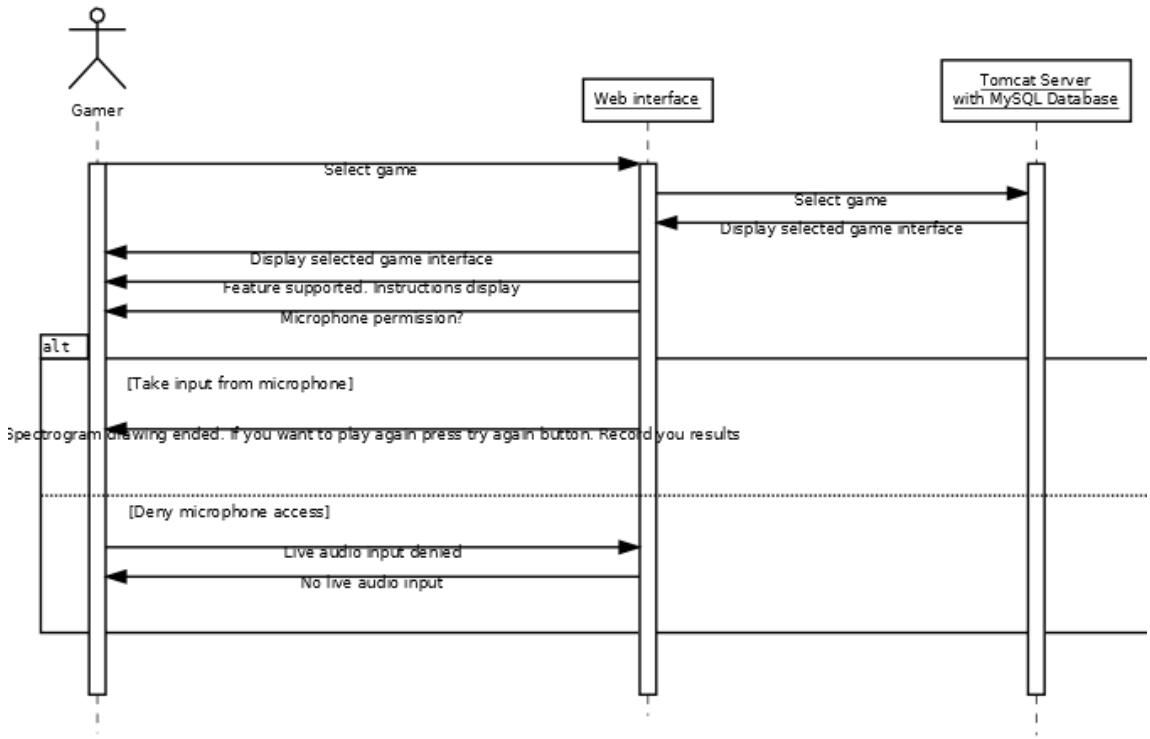


Figure 4.32: Spectrogram sequence diagram

4.1.8.2 Server Sequence diagram

In this section we present sequence diagrams for server side. No further description is provided as they are quite informative and self explanatory.

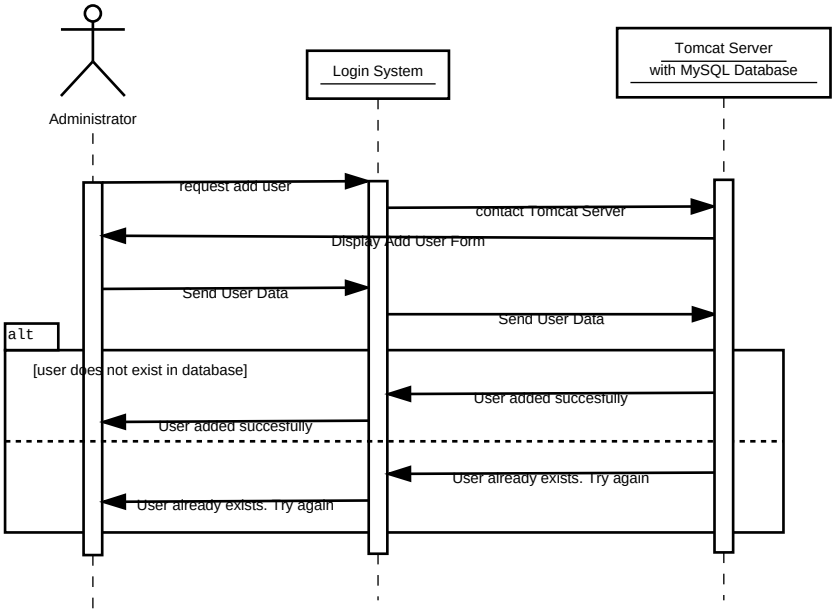


Figure 4.33: Add users sequence diagram

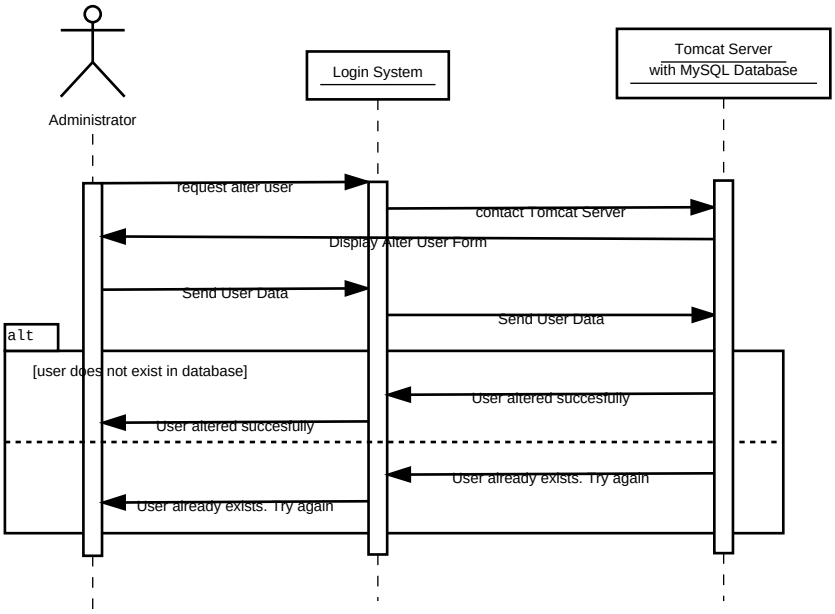


Figure 4.34: Alter user sequence diagram

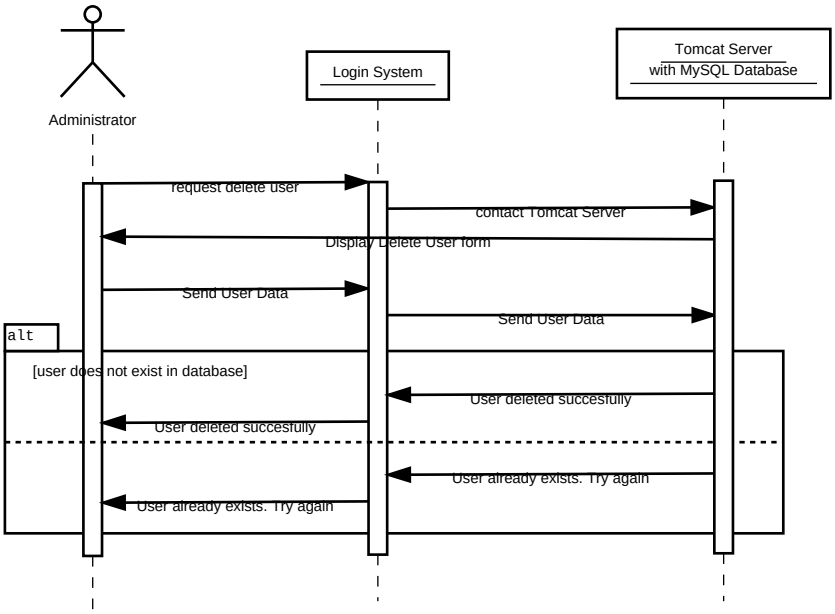


Figure 4.35: Delete user sequence diagram

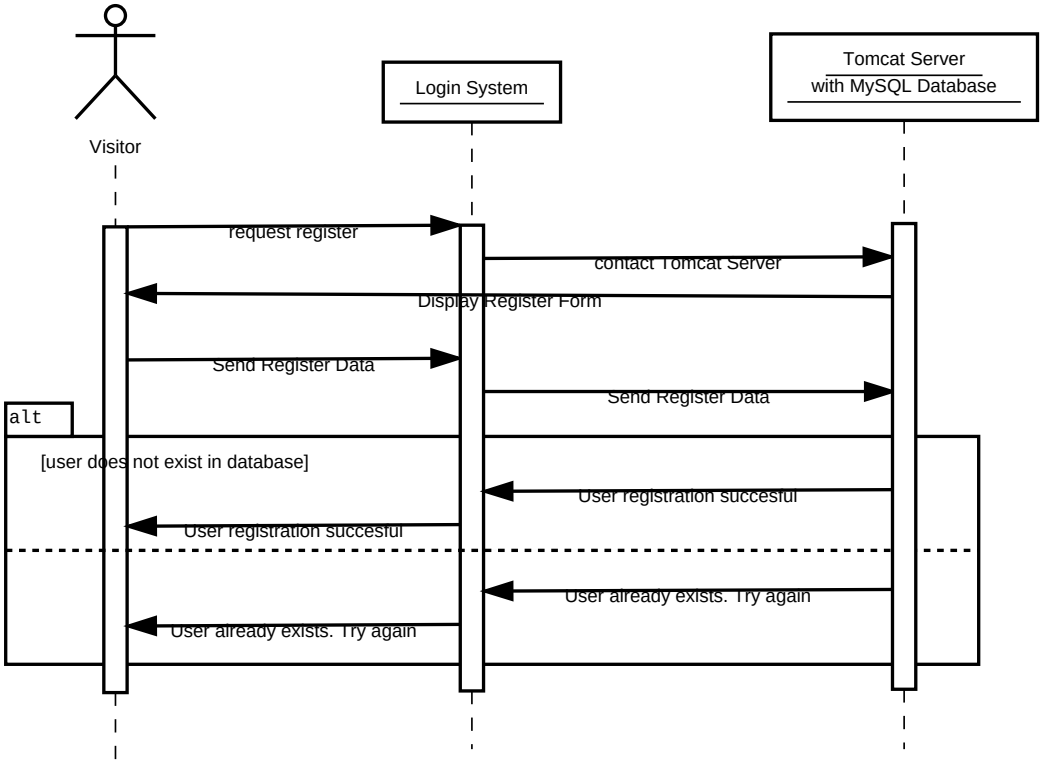


Figure 4.36: Register sequence diagram

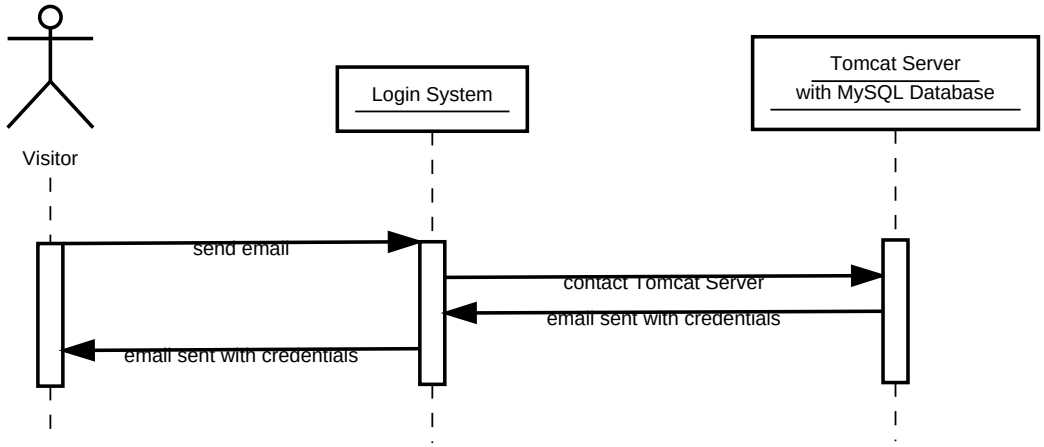


Figure 4.37: Forgot credentials sequence diagram

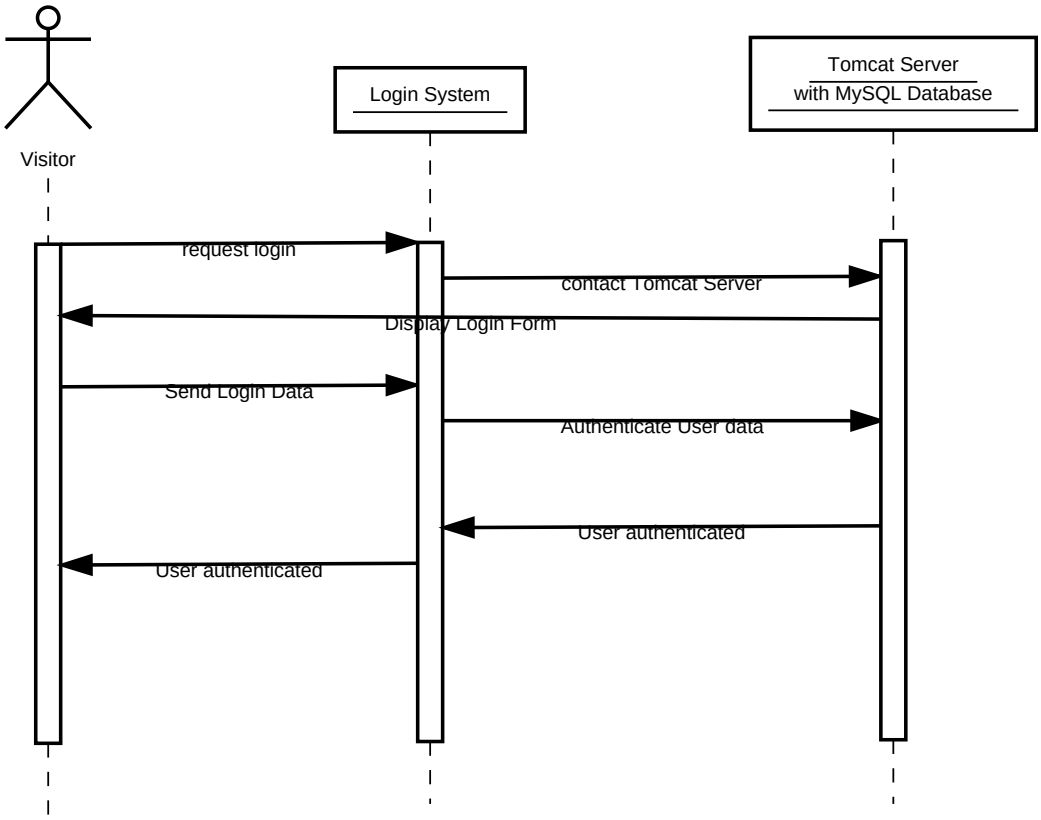


Figure 4.38: Login sequence diagram

4.1.9 Database schema, E-R diagram

In this section we present E-R diagrams for our database schema which lies into our server. We modeled all basic entities of our system and their relationships. User, roles, scores, games, levels, type of each game are depicted in Figure 4.39 as tables and relationships in our database. Each table has it's own properties and relationship which is defined from system's requirements.

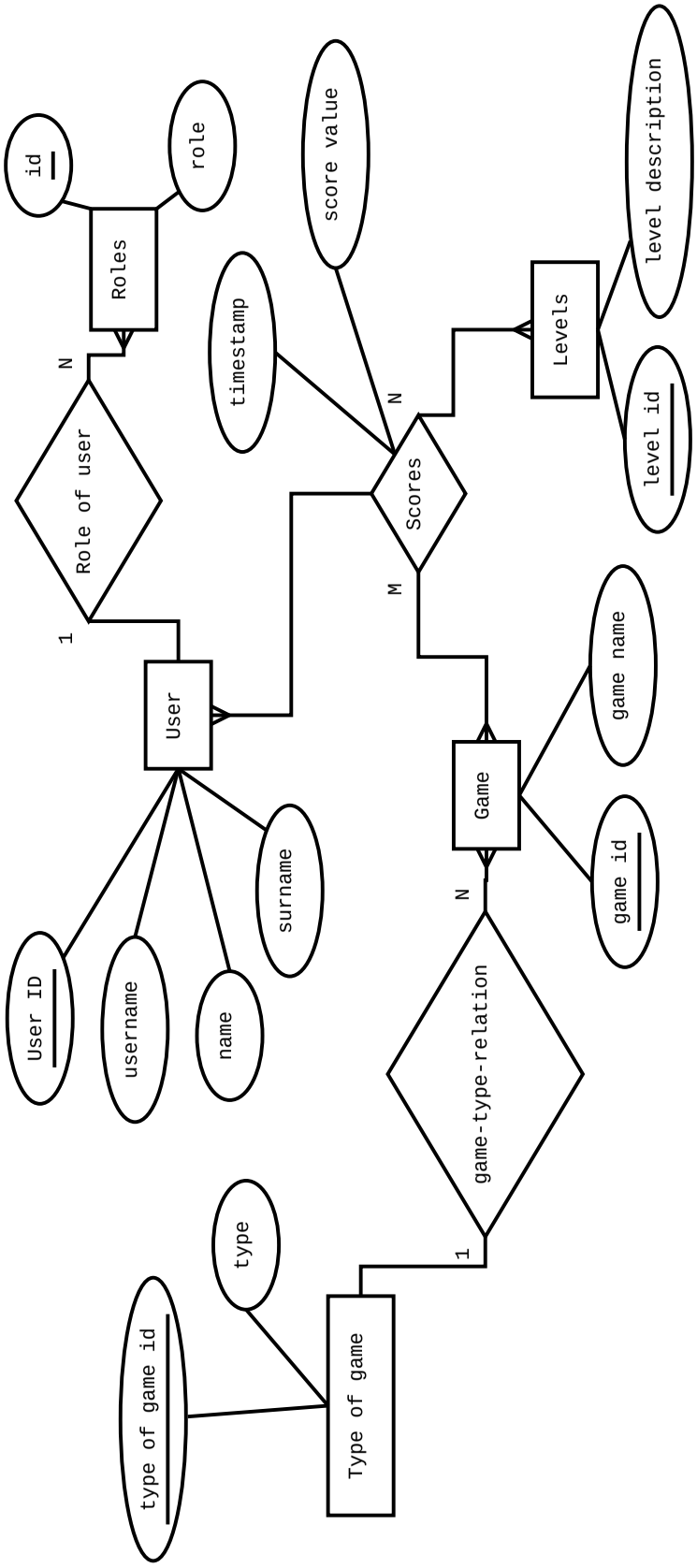


Figure 4.39: E-R diagram

A short description of each entity and relationship is given below.
Entities:

- User. In this table credentials of each user are saved. Attributes of this table are (`user_id`, username, name, surname).
- Roles. In this table roles of each user are saved. Attributes of this table are (id,role). Several roles are provided with different rights. These are (admin, secure, user).
- Levels. In this table levels of each game are saved. Attributes of this table are (`level_id`, `level_description`). Several levels are provided (easy, medium, difficult)
- Game. In this table information of each game are saved. Attributes of this table are (`game_id` , `game_name`, `type_of_game_id`). In our system three types of games are supported (spacecraft, pitch, vowel game).
- Type of game. In this table type of game is are saved. Attributes of this table are (`type_of_game_id`, type). Each game has a type. In our system three types of games are supported (intensity, pitch, vowels).

Relationships:

- Role Of User. It's a one-to-many relationship. One user can have many roles.
- Game-type-relation. It's a one-to-many relationship. One type of game can be matched to many games.
- Scores. It's a many-to-many relationship. This is a relationship between three tables (table user, table game, table levels). Many users can play many games in many levels. It is represented in our schema as an extra table. It's attributes are (id,`game_id`, `level_id`, tries, timestamp, scores). Also, scores, number of tries and date of calculation of our games are saved in this table.

In Figure 4.40 we can examine our database schema as it is represented by phpmyadmin designer tool.

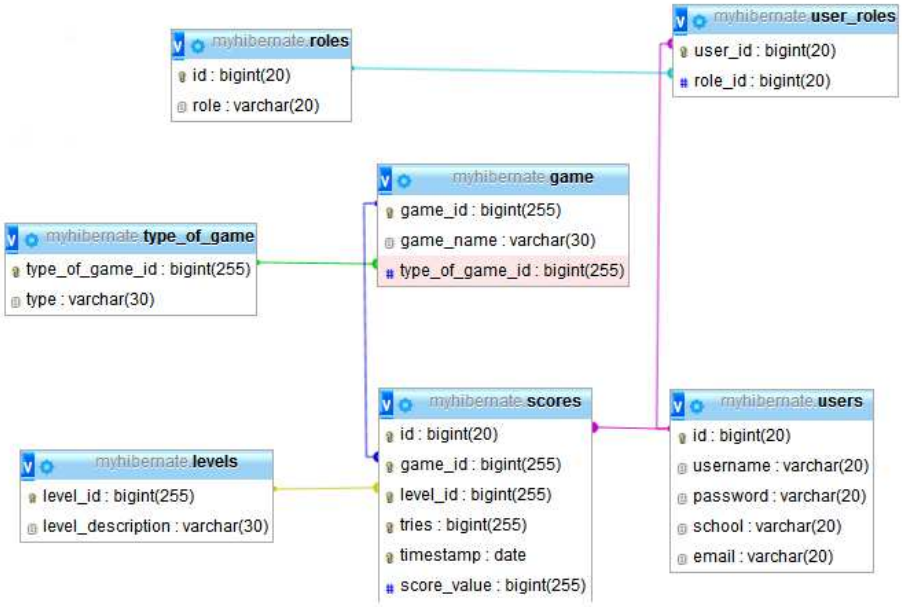


Figure 4.40: Database schema

Chapter 5

Evaluation

5.1 Introduction-Method

The evaluation of such a system are in long-term the children themselves which by practice the show or not improvement. In most cases though, the pre-evaluation of ths system is held by experienced users of speech. This kind of users could be experienced speech therapists. In order to evaluate our system we created a questionnaire for filling up by speech therapists. The evaluation gave details about the effectiveness, feasibility and accessibility of the our system in the treatment of speech by hearing-impaired children.The results of evaluation is depicted below.

5.2 Results

Evaluation questionnaire
Please answer the following questions in the range 0-10.

1. Do you think that experience with computers is necessary to be able to use this system? **8**
2. Do you think that phonetic knowledge is necessary to work with this system? **10**
3. How did the system meet with your expectations? **10**
4. Was the system easy to handle? **10**
5. Did you consider the training as meaningful? **10**
6. Were the performance graphs useful? **10**
7. Were the performance graphs easy to use? **10**
8. How was the system from a pedagogical point of view? **10**
9. Do you think that game interaction with the children is easy?(Pitch:**10**, Intensity:**10**, Articulation:**8**)
10. Do you think that visual feedback of games is easy to understand by children? **8**
11. Was the game reliable in terms of giving consistent and correct feedback? **10**
12. For which age group is this kind of game appropriate? **5** +
13. Do you think that children are motivated to train with the system? **10**

14. Did you miss the possibilities to train something? If yes, what? **No. The tool serves it's purpose.**

15. Would you like to see more games in our system? **yes**

16. Would you like to see more extensions in our games? **yes**

17. How important is the remote access to the system, to speech therapy procedure? **10**

18. How speech therapy procedure could benefit from online speech therapy tools?

    - ***Accurate data on the progress and development of the program Pathologists.***
    - ***The data is accessible via internet and thus "always" available, which facilitates therapists and saves time because you do not need to take notes on the progress of the treatment program.***
    - ***Some operations can be done in the natural environment of the child via computer. The results of these activities can be discussed with parents and expedite treatment plan under the supervision of the speech therapist.***
    - ***Teachers or any other interested parties can be informed immediately of the disorder or for the development of the therapeutic program of the child, since the data is available via internet.***
    - ***When such activities through computer is through play, children have a strong incentive to engage in and participate enthusiastically in the therapeutic process.***

5.3 Discussion

As we can see in answers of evaluation questionnaire our system requires user to have experience with computers in order to be used. Also phonetic knowledge is required in order to work with this system. This is reasonable because of the existence of spectrograms where someone has to know how to evaluate them in order to be used. Also as we can notice our system is quite easy to use with pleasant pedagogical interaction and performance graphs are quite important in terms of usefulness. Furthermore, visual feedback is quite easy to understand and quite reliable. Finally in last question we can see that online speech therapy software tools are very important in speech therapy procedure because they help therapists to save time, to have better tracking performance of each child through performance graphs and for children to have strong incentive to engage in and participate in the therapeutic process.

Chapter 6

Comparison with other commercial tools

As we mentioned in our introduction the main disadvantage of the existing tools is that they are developed for commercial use. Therefore, the cost to obtain a speech therapy tool is quite high especially if it is oriented for public use (e.g in public schools for educational purposes). Moreover, these tools are not easily adaptive and flexible. As they are oriented for standalone commercial use, the update process lasts in time and costs money as most of the times to get an updated version requires to pay for the whole program again. Furthermore, none of the tools is developed for use by Greek children. These disadvantages are faced through our system. Our system provides

- Low cost
- Easy access
- Real time spectrograms through web is a new feature.
- Flexibility and adaptation to user profile
- No installation restrictions.
- Platform independence
- Low cost in memory and CPU requirements
- Versatility
- Code re-use is another positive side-effect of Web services' interoperability and flexibility. One service might be utilized by several clients, all of which employ the operations provided to fulfill different game objectives. Instead of having to create a custom service for each unique requirement, portions of a service are simply re-used as necessary.

Chapter 7

Conclusions and Future Work

7.1 Requirements and Restrictions

Basic requirements of our system is that user of our system must have access to the internet. Since we run our game through browser, our system is platform independent. User can access our system through all platforms that can run all mainstream modern browsers such as Mozilla Firefox, Google Chrome, Safari etc. Restrictions of our system are produced due to browser behaviour deviation. All browser has to run Javascript and Java. More specifically browsers have to support Javascript Web Audio API in order to take access to microphone data. This feature is present for Google Chrome browser but not in other browsers. Since Web Audio API is a new API that is growing up every day we expect other browsers to support it too. In Figure 7.1 we can see which browser versions supports Web Audio API. As someone can see most modern browsers are changing their policy and are starting to support Web Audio API.

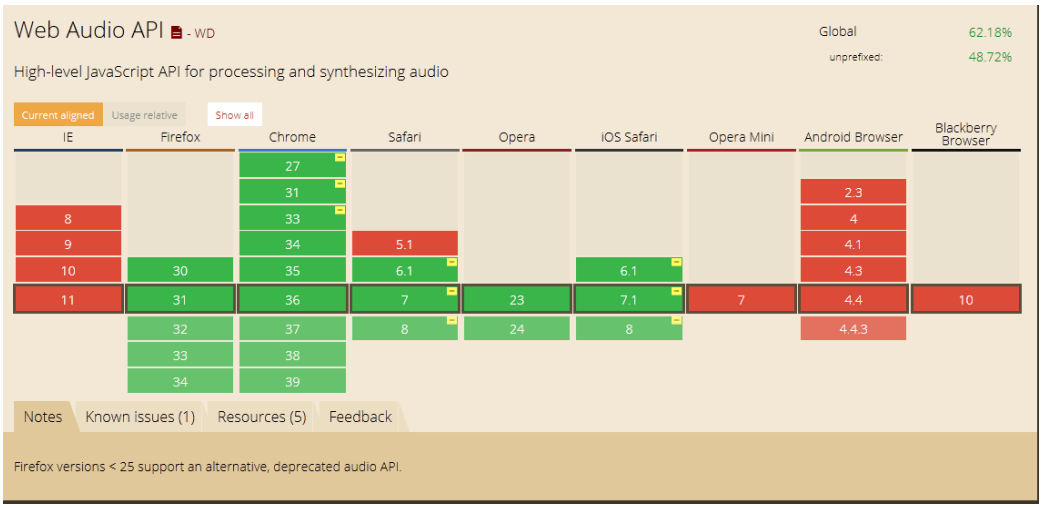


Figure 7.1: Web Audio support/browser version

Similarly we provide Java implementation for pitch game. We could also select Java for other games. Restrictions for our Java implementation exist too. These restrictions are produced since Oracle decided to change his policy towards Java applets, releasing new updates for security reasons. Until now a common scenario for development with Java Applets was:

- The developer developed Java application
- The developer was signing Java application with a self certificate

- The developer was releasing his self certificate (for example in his personal web page)
- The user was installing developer's self certificate
- The user run's Java applet

But since Java 7 Update 45 CPU this scenario was modified. Self certificate is not acceptable any more. Developers has to buy a code signing certificate from companies like Commodo or Thawte. These code signing certificates usually are expensive to buy. If a developer has not such certificate, then his application is blocked for security reasons. Latest release of Java is Java 7 Update 67 Limited Update (August 4 2014) which is blocking self signed applications.

For our project we used Java update 21 (April 16 2013)and we signed our code with our personal certificate in order for the system to allow execution. Our personal certificate was created with OpenSSL and installed in our browser. If a newer update of Java is installed then our application is blocked too. So this restriction is very serious one because converts Java into a non flexible language for an independent developer. In Figure 7.2 we can see a list of latest Java Updates

Java 7 Releases	Release Date
Java 7 Update 67 Limited Update	August 4, 2014
Java 7 Update 65 CPU	July 15, 2014
Java 7 Update 60 Limited Update	May 28, 2014
Java 7 Update 55 CPU	April 15, 2014
Java 7 Update 51 CPU	January 14, 2014
Java 7 Update 45 CPU	October 15, 2013
Java 7 Update 40 Limited Update	September 10, 2013
Java 7 Update 25 CPU	June 18, 2013
Java 7 Update 21 CPU	April 16, 2013
Java 7 Update 17 - Special Update[3]	March 4, 2013
Java 7 Update 15 CPU - Special Update	February 19, 2013
Java 7 Update 13 CPU	February 1, 2013
Java 7 Update 11 CPU[2]	January 13, 2013
Java 7 Update 10 Limited Update	December 11, 2012
Java 7 Update 9 CPU	October 16, 2012
Java 7 Update 7 - Special Update[1]	August 30, 2012
Java 7 Update 6 Limited Update	August 14, 2012
Java 7 Update 5 CPU	June 12, 2012
Java 7 Update 4 Limited Update	April 26, 2012
Java 7 Update 3 CPU	February 14, 2012
Java 7 Update 2 Limited Update	December 12, 2011
Java 7 Update 1 CPU	October 18, 2011
Java 7 Release	July 28, 2011

Figure 7.2: Java updates releases

7.2 Implementation issues and time-restrictions

The suggested implementation roadmap of the project is consisted of two independent parts. The first part is the development of the applet and Javascript

games and the second part is the development configuration and deployment of the application server. These two parts could be either developed by a single team sequentially or by two teams in parallel.

7.3 Extensions Future work

Based on the time limitations, this thesis focused on the development of the Java applet and HTML5 - Javascript games initially and the integration to Apache Tomcat in a later stage. Also, three speech properties are incorporated in our development. On future collaboration, more games could be developed in order to test more speech properties and more statistical graphs. These could be included in extended collection of browser games where a bunch of speech properties could be tested in order to give even more possibilities to children. These could be apart for the above mentioned

- Speech waveforms
- Prosody
- Speech rate
- Spectrograms
- Phoneme pronunciation
- Articulation and coarticulation

7.4 Conclusions

The main focus of this thesis was to create an online, 24-hour, non-commercial and educational platform that will help children with hearing problems to train their voice. This system is intended to be used by children of ages 5-12 with the presence of speech therapists. Firstly we collected data about existing speech therapy software tools and we studied types of feedback that could help us to achieve our purpose. In second step we took advantage of this bibliography research and developed games which could help children train with voice intensity and voice pitch. Also, real time spectrograms are being drawn which are useful for visual identification of consonants, aiming on training children with visual feedback. Scores of each user are saved in our server and special graphs can be produced in order to follow user's performance. Evaluation is being provided by speech therapists.

Bibliography

[1] From Wikipedia the free encyclopedia. Rubella. http://en.wikipedia.org/wiki/Rubella. 3

[2] Klara Vicsi. Computer-assisted pronunciation teaching and training methods based on the dynamic spectro - temporal characteristics of speech.dynamics of speech production and perception p. divenyi (ed.). *IOS Press,*, 374:283–306, June 2006. 3, 12, 15, 16, 25

[3] Rafi Shemesh. Hearing impairment: Definitions, assessment and management. http://cirrie.buffalo.edu/encyclopedia/en/article/272/. 5

[4] Moores. Educating the deaf: Psychology, principles, and practices (5th ed.). *Boston: Houghton Mifflin*, 2001. 5, 7

[5] Marshchark. Raising and educating a deaf child: A comprehensive guide to the choices, controversies, and decisions faced by parents and educators. *New York: Oxford University Press*, 1997. 6

[6] Cruickshanks KJ. Prevalence of hearing loss in older adults in beaver dam, wisconsin. *American Journal of Epidemiology*, pages 148:879–886, 1998. 6

[7] From Wikipedia the free encyclopedia. Lip reading. http://en.wikipedia.org/wiki/Lip_reading. 7, 8

[8] Dorothy Clegg. The listening eye: A simple introduction to the art of lip-reading, methuen and company. 1953. 8

[9] Adam Schembri. Understanding auslan: How do children learn sign languages. *Australian Association of the Deaf Inc AAD Outlook*, 14 Issue 4:3, May 2005. 9

[10] Heidi Hanks. How to teach the f sound and v sound. http://mommyspeechtherapy.com/?p=1870. 9

[11] Articulate Technologies. Speech buddies. http://www.speechbuddy.com/slps/provider-program. 9

[12] Madeline Hayes. Tongue placement exercises for speech therapy at home. http://voices.yahoo.com/tongue-placement-exercises-speech-therapy-home-3914210.html?cat=25. 9

[13] Maxine Eskenazi. An overview of spoken language technology for education. *Speech Communication*, 51 Issue 10:832–844, 2009. October. 11

[14] Special Needs Systems. Overview of speechviewer iii. ftp://ftp.boulder.ibm.com/sns/spv3/spv3supt.htm. 12

[15] Sakshat Virtual Labs. Estimation of pitch from speech signals. http://iitg.vlab.co.in/?sub=59&brch=164&sim=1012&cnt=1. 12

[16] Bernstein J. and Christian B. For speech perceptions by human or machines, three senses are better than one. *Proc ICSLP*, pages 1477–1480, 1996. October. 12

[17] Markham D and Nagano Madesen Y. Proceeding of the international conference on spoken language processing. pages 1473–1476, 1996. October. 12

[18] Macquarie University. Waveform definition. http://clas.mq.edu.au/speech/acoustics/waveforms/speech_waveforms.html. 12

[19] From Wikipedia the free encyclopedia. Prosody. http://en.wikipedia.org/wiki/Prosody_%28linguistics%29. 13

[20] Mark W Pellowski. Speech-language pathologists knowledge of speaking rate and its relationship to stuttering. *Contemporary Issues in Communication Science & Disorders*, 37:50, March 2010. 13

[21] From Wikipedia the free encyclopedia. Spectrogram. http://en.wikipedia.org/wiki/Spectrogram. 13

[22] David J Ertmer. How well can children recognize speech features in spectrograms? comparisons by age and hearing status. *Journal of Speech*, 47 Issue 3:484, June 2004. 14, 15

[23] Peter Ladefoged and Keith Johson. *A Course in Phonetics 6th ed.* 2010. 15

[24] Massaro Dominic W Light Joanna. Using visible speech to train perception and production of speech for individuals with hearing loss. *Journal of Speech, Language, and Hearing Research*, 47 Issue 2:304, Apr 2004. 17

[25] Vicsi K Roach P Oster A Kacic Z Barczikay P Tantos A Catari F Bakcsi Zs and Sfakianaki A. A multimedia, multilingual teaching and training system for children with speech disorders. *International Journal of Speech Technology*, pages 289–300, Apr. 18, 25

[26] Inc Communication Disorders Technology. Istra 'indiana speech training aid features'. 18

[27] Eduardo Lleida Richard Rose Carlos Vaquero William R. Rodriguez Oscar Saz, Shou-Chun Yin. Tools and technologies for computer-aided speech and language therapy. 9 April 2009. 19

[28] K Vicsi and A Vary. Distinctive training methods and evaluation of a multilingual, multimodal speech training system. , NOTE = 1999. 23

[29] Dominic Massaro. Computer-animated tutor for spoken and written language learning. *ICMI '03 Proceedings of the 5th international conference on Multimodal interfaces*, pages 172–175, 2003. 23, 25

[30] Dominic Massaro. Baldi youtube video. http://www.youtube.com/watch?v=p1gZodEQ2xE&list=UUujwg_Zl3HKfuhuEmUZuzRQ. 23, 25

[31] Sascha Fagel & Katja Madany. A 3-d virtual head as a tool for speech therapy for children. *INTERSPEECH*, 2008. 25

[32] The Apache Software Foundation. Apache tomcat. http://tomcat.apache.org/. 26

[33] The Apache Software Foundation. Apache shiro. http://tomcat.apache.org/. 26, 34

[34] RedHat. Resteasy. http://resteasy.jboss.org/. 26, 42

[35] Oracle. Mysql. http://www.mysql.com/. 26, 37

[36] From Wikipedia the free encyclopedia. Hibernate description. http://en.wikipedia.org/wiki/Hibernate_%28Java%29. 26, 36

[37] From Wikipedia the free encyclopedia. Pitch detection algorithm. en.wikipedia.org/wiki/Pitch_detection_algorithm. 28, 29

[38] Alain de Cheveigne & Hideki Kawahara. Yin a fundamental frequency estimator for speech and music. 9 January 2002. 29

[39] From Wikipedia the free encyclopedia. Spl. http://en.wikipedia.org/wiki/Sound_pressure. 33

[40] From Wikipedia the free encyclopedia. Xampp:wikipedia. http://en.wikipedia.org/wiki/XAMPP. 38

[41] Apache Friends. Rest description. http://www.xfront.com/REST-Web-Services.html. 39

[42] Roy Thomas Fielding. Architectural styles and the design of network-based software architectures. http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm. 39

[43] From Wikipedia the free encyclopedia. Java. http://en.wikipedia.org/wiki/Java_%28programming_language%29. 43

[44] From Wikipedia the free encyclopedia. Javascript. http://en.wikipedia.org/wiki/JavaScript. 44

[45] Chris Rogers Google. Web audio api. https://dvcs.w3.org/hg/audio/raw-file/tip/webaudio/specification.html#introduction. 44

[46] From Wikipedia the free encyclopedia. Html5. http://en.wikipedia.org/wiki/HTML5. 44

[47] From Wikipedia the free encyclopedia. Jsp. http://en.wikipedia.org/wiki/JavaServer_Pages. 45

[48] From Wikipedia the free encyclopedia. Xml. http://en.wikipedia.org/wiki/XML. 45

[49] From Wikipedia the free encyclopedia. Css3. http://en.wikipedia.org/wiki/CSS3#CSS_3. 45

[50] From Wikipedia the free encyclopedia. Clientserver. http://en.wikipedia.org/wiki/Client%E2%80%93server_model. 47

[51] From Wikipedia the free encyclopedia. Uml. http://en.wikipedia.org/wiki/Unified_Modeling_Language. 48

[52] Joren Six. Tarsos, a modular platform for precise pitch analysis of western and non-western music. http://0110.be/tags/Java. 60