Middle East Technical University

Northern Cyprus Campus

Department of Computer Engineering

Senior Project and Seminar Design

CNG 492

Course Project

A Mobile Application to Teach Turkish Sign Language

Final Report of TMT

Advisor: Assist. Prof. Dr. Yeliz Yeşilada

Instructor: Assist. Prof. Dr. Islam Elgedawy

Enes Bükte - 1543172

Musa Gök - 1543248

Nihal Yağmur Aydın– 1727700

ABSTRACT

This report aims to explain the senior project which concerns with developing an android application by using client-server approach to teach Turkish Sign Language. This application will be used by both hearing impaired people and people who wish to learn Turkish Sign Language. This project is designed as education tool rather than being a vocabulary so that every people can enhance their abilities by testing themselves or learning new words of vocabulary. It consists of three main parts which are android application, client-server and avatars. Client-server approach is used to retrieve videos from the server instead of downloading them to the phones since downloading process occupies memory space in android phones. In addition to that, this project uses avatars for signing words instead of videos that are made by human. The work that we achieved consists of creating android GUI, lectures and education tools in client part; creating some avatars; creating a web service to handle requests through database, GUI to manipulate data in server part and connecting android with our web service to play video on android phone.

1. INTRODUCTION

This project intends to create an android application which aims to teach Turkish Sign Language both hearing impaired people and people who wish to learn Turkish Sign Language. It is an education tool for teaching Turkish Sign Language rather than just being a usual vocabulary. This project benefits from avatars for signing words and uses android application for playing videos that has avatars in it. Avatars are played on server instead of android phone. This project has three main parts which are as follows: creating android application, avatars and servlet by using client-server approach.

PART 1: Android Application

Android application is needed to be created since mobile applications are more popular in these days. People tend to spend time with their mobile phones in every occasion. Especially when waiting for meetings in the waiting room or waiting a bus in a bus station etc. One of the concerns of many people is using time efficiently, so developing an android application for smart phones is an important issue when it comes to need of human beings. In our project, android application development is used for creation of GUI. We benefit from Android API of java to develop our application and handling user I/O. Android application has two parts which are vocabulary search and education tools.

PART 2: Avatars

In our project, avatars are used for signing words in Turkish Sign Language. For creation of avatars, Vcom3D [1] is used. Existing libraries of Vcom3D are for American Sign Language, which we don't use in our project. When it comes to other programs, we thought of using Blender [2]. However, after importing a model by using a program called "Make Human" [3] into Blender, we found that it was quite time consuming and not as sophisticated as Vcom3D. The another program called Vsign Builder and Vsign Player [4] also exist for creating avatars, whereas this program's quality of resolution and also avatar model is not as good as Vcom3D. Vcom3D allows us signing words with high quality by using our mouse and exporting them as media player files and we convert them to 3gp before putting to server.

PART 3: Client-Server Side

Client-server approach is used in our program for saving memory space in android phones. Restful web service with MySQL connection is used to serve videos to the android client. Also, to make the work of server side client easier, GUI in JSP is created to upload videos to the server, view, insert data into and delete data from the database. Many API's of Java programming language which include File Upload, Persistence, Rest, JDBC and MySQL connectivity are used in server side. In addition to that, some HTML is used inside JSP pages, together with java codes, to make program outlook better.

2. PROBLEM STATEMENT

There are some applications for hearing impaired people that are developed by some universities such as Bosporus University [5] and Koç University [6]. However, these applications are created as desktop or online applications and they cannot be used by mobile phones. Moreover, they are created as vocabularies rather than being education tools. These vocabularies don't use any avatars. Human videos are used for signing words which causes this application to be inflexible for some applications of computer science such as natural language processing. With the help of avatars, sequence of videos can be concatenated and creating sentences could be possible for future work. Because of these reasons, Teach Me Turkish project is created.

3. SOLUTION: THEORETICAL ADAPTION

We applied software engineering principle and rules in our theoretical adaption. First of all, we thought about the general structure of our project and did some estimation:

- inputs : 17
- outputs: 8
- inquiries :40 – retrieval from server
- logical files : -
- external interface files: -

UTFP = 17*6+ 5*7+ 6*40 = 377(high complexity)

Influence Multiplier = 0.7

ATPF = 377 * 0.7 = 263.9

LOC = 263.9* 55 = 14514.5

Due to Cocomo model 81', our project is embedded.

KDSI = Language size / 1000

KDSI = 14514.5/1000 = 14.514

MM = 3.6 * (14.514) ^1.2 = 89.21

4. SOLUTION: ANALYSIS

Our project has 3 parts, Android programming part, Avatar Creating part, and Server part. Our android program has to connect the server and play the avatar videos which we created on the avatar creating program. We firstly implemented a httpservlet in our first project. However, it wasn't working fine when there will be lots of videos in our project. Therefore, in the server part, Restful web service which has connection with database is coded to return json formatted data to the android client. Also, for helping the programmer in server side, java server pages (jsp) is used to upload videos to the server, search the database table content, insert and delete data in database. These changes are reflected on the web service output, when the client requests data by using name of the video files as keyword from the server program.

Our Functional Requirements;

1. The system should allow users to choose language of the program.

2. The system should allow users to learn the details of the program.

3. The system should allow users to get some help about the details of the program

4. The system should allow users to search vocabulary

4.1 The system should allow users to search by categories.

4.2 The system should allow users to search by name

5. The system should allow users to see the avatars of the searched items

6. The system should allow users to see the details of the processing time of the retrieved item.

7. The system should allow users to pass from one page to another page

8. The system should allow users to use educational tools such as exams and lectures.

8.1 The system should allow users to perform the exams

8.1.1. The system should allow users to choose exam types such as multiple choice, matching and flash videos.

8.2 The system should allow users to see the lectures related to animals and jobs

8.2.1 The system should allow users to choose the lectures of the related topics

8.3. The system should allow server side client to enter data including name, path, category and type into the database.

8.4. The system should allow server side client to see the database content.

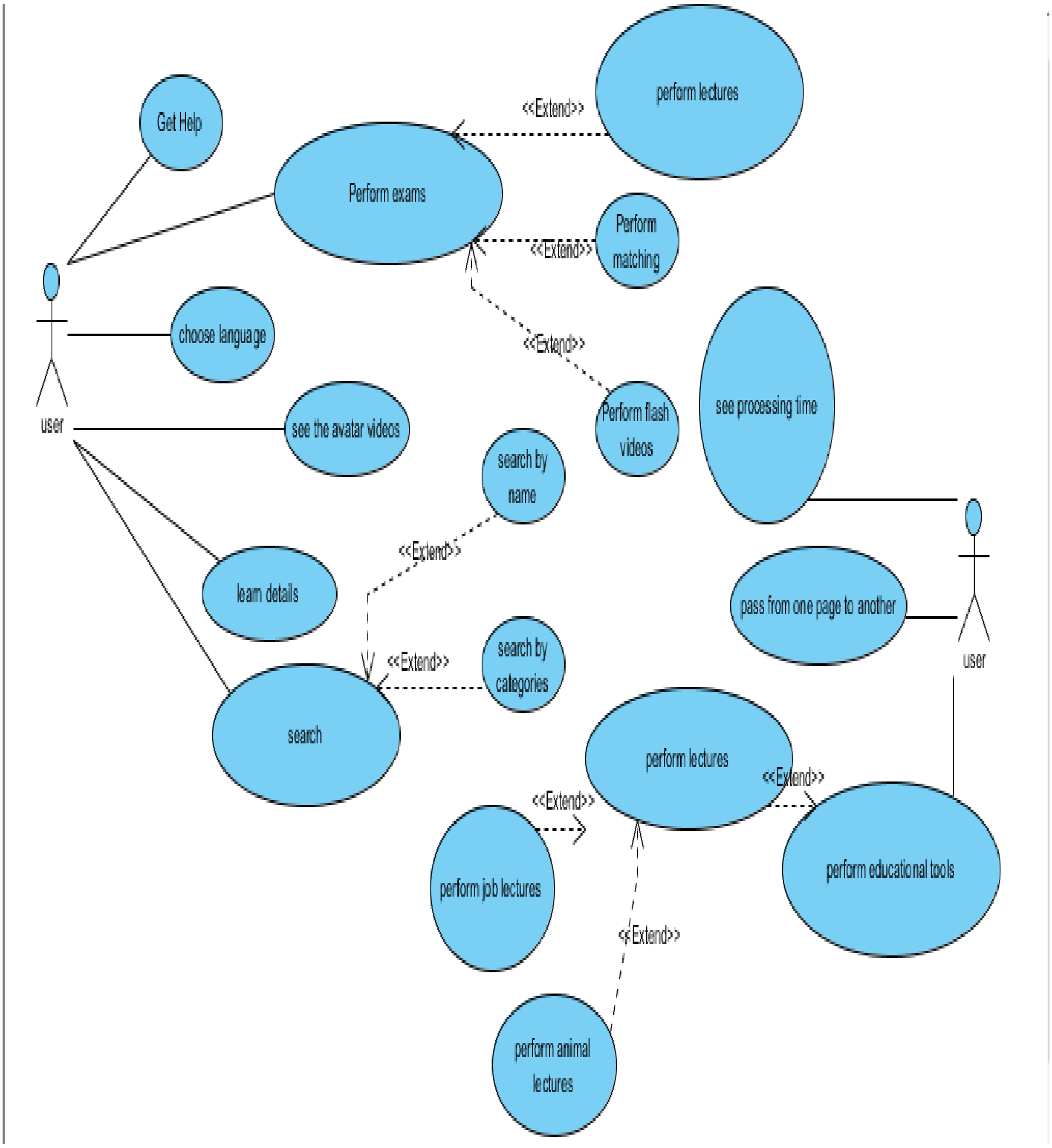
8.5. The system should allow server side client to delete data from the table.

8.6. The system should allow server side client to upload videos to the server.

5. SOLUTION: MODELING

We have designed a Sequence diagram, Architecture diagram, Use Case diagram.

Use Case Diagram:



The use case diagram above aims to understand the perspective of user side of our application. We wanted to show that what can be done by users in our application program. Entity functions are showed and some of them has extensions .When a user enters the program, h/she can able to perform these functions above.

Use Case: Vocabulary Search

Pre-condition: User chooses search type as animals.

Actor: User

Primary Scenario: Choose an animal and see the sign language

User System

1. Users clicks the vocabulary search in the system

2. The system opens the search page and shows the screen

3. After the screen opened, user chooses the type of the search as animal and clicks

4. The system shows the names of the animal such as bee, rooster etc…

5. User chooses one name on the screen and clicks

6. After clicking the button, the system shows the avatar video of the chosen animal

Use Case: Vocabulary Search

Pre-condition: User chooses search type as jobs.

Actor: User

Primary Scenario: Choose a job and see the sign language

User System

1. Users clicks the vocabulary search in the system

2. The system opens the search page and shows the screen

3. After the screen opened, user chooses the type of the search as jobs and clicks

4. The system shows the names of the jobs such as worker, advocate etc…

5. User chooses one name on the screen and clicks

6. After clicking the button, the system shows the avatar video of the chosen jobs

***Use Case*: Vocabulary Search**

Pre-condition: User chooses search by only name.

Actor: User

Primary Scenario: Choose a name and see the avatar video

User System

1. Users clicks the vocabulary search in the system

2. The system opens the search page and shows the screen

3. After the screen opened, user enters the name of the searched item and clicks the search button

4. The system shows the avatar video of the searched name

Use Case: Educational Tools

Pre-condition: User chooses exams.

Actor: User

Primary Scenario: Choose an exam type as matching

User **System**

1. The user clicks on the educational tools

2. After chooses the educational tools, system shows the exams and lectures screen

3. User chooses the exam and clicks the button

4. After clicks on the exam button, system shows the flash video exams, matching and multiple choice screen

5. User chooses the matching button and clicks

6. After clicks the matching button, system shows the matching exams

7. User chooses one of the matching exams and clicks

8. System shows the matching exam.

9. User performs the questions in the matching exam

Use Case: Educational Tools

Pre-condition: User chooses lectures.

Actor: User

Primary Scenario: Choose an lecture type as animal lectures

User System

1. The user clicks on the educational tools

2. After chooses the educational tools, system shows the exams and lectures screen

3. User chooses the lectures and clicks the button

4. After clicks on the lecture button, system shows the animal lectures and the job lectures

5. User chooses the animal lectures button and clicks

6. After clicks the animal lectures button, system shows the related lectures

7. User chooses one of the animal lectures and clicks

8. System shows the related animal lectures

9. User performs the lectures.

Use Case: Server Side Data Manipulation

Pre-condition: Server side user has permission to access the database on which restful web service is created.

Actor: Server Side User

Primary Scenario: First server side user has avatars in a folder at his/her computer.

User System

1. The user opens the start page of the jsp files.

2. User is prompted to either choose to upload a video and continue to the process of the insertion, or view the content of table, having an option to delete any entry.

3.The user chooses the video upload.

4.System prompts user to choose file

5.User chooses the file to be uploaded.
User clicks the upload button.

6.System shows user file properties such as size, name and type.

7.User continues by clicking continue button

8.System allows user to enter name, choose the type and category.

9. User inputs the data and click the submit button.

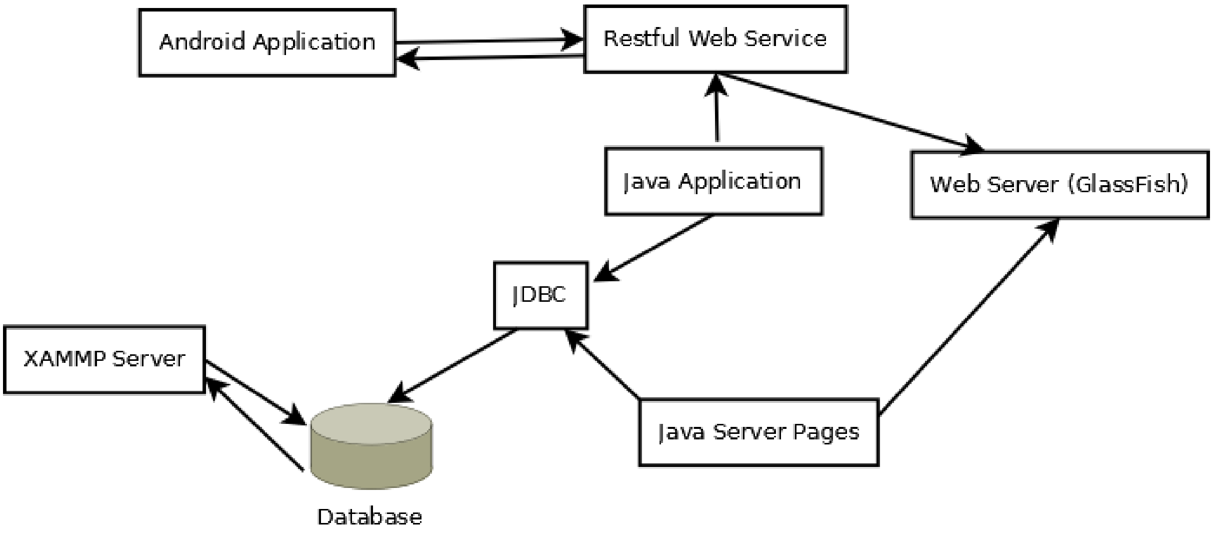
10. System tells user data is inserted

11. System provides an option to enter new entry or view table.

12.User chooses to view table by clicking view table button.

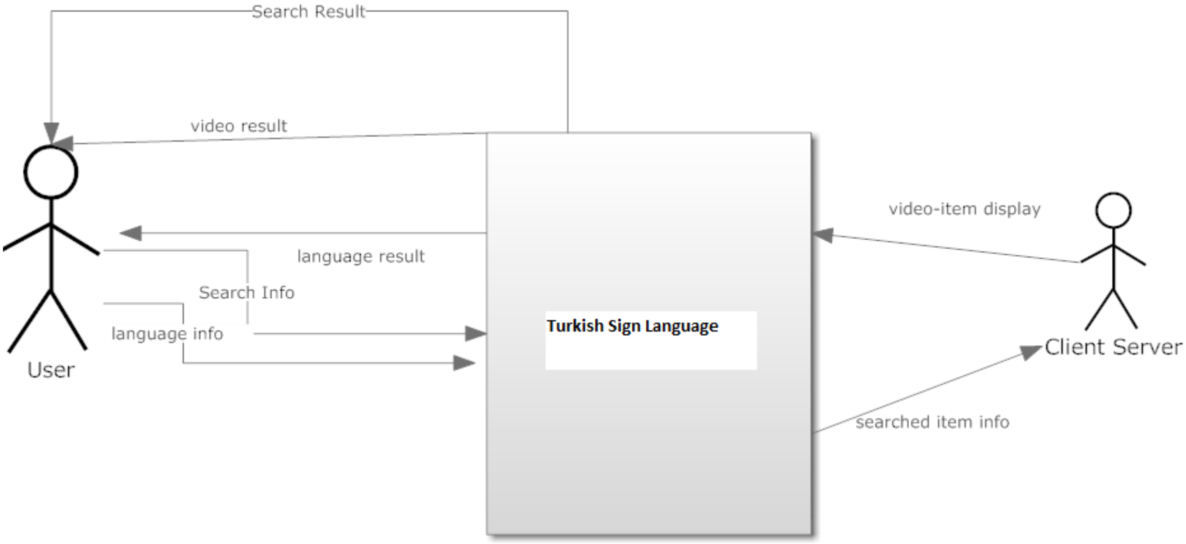
13.System displays the table content allowing user to delete any entry .

Architecture Diagram:

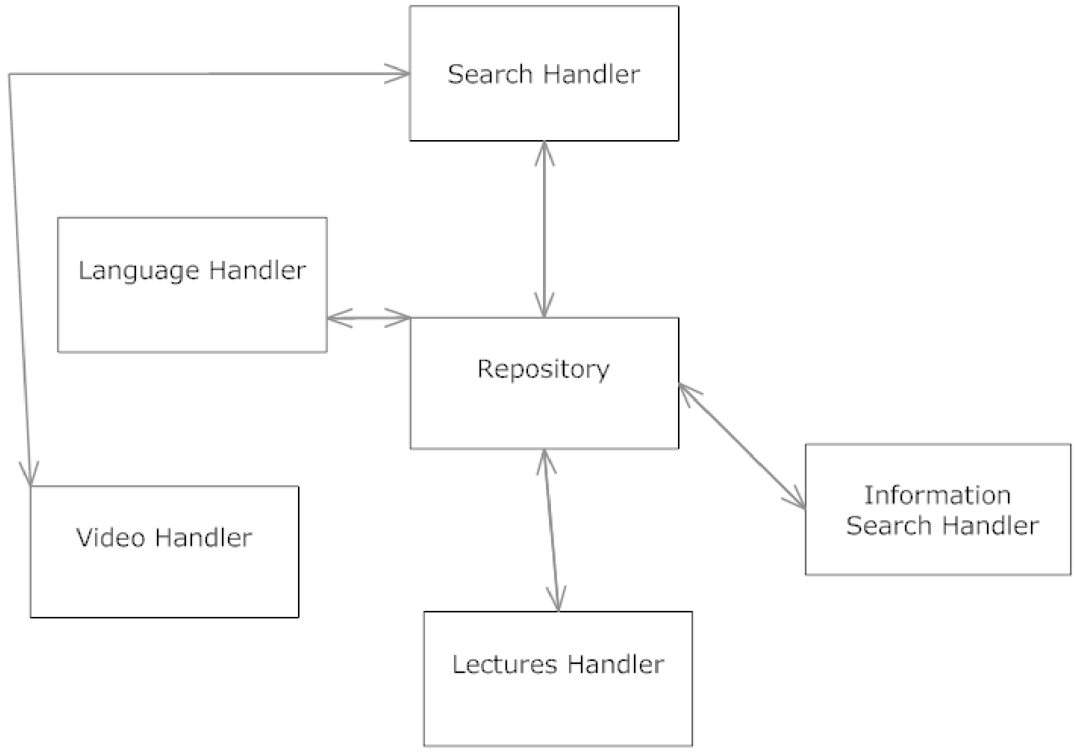


First of all, MySQL Database Server is managed by XAMMP Server. Then, through JDBC, another program handles database requests coming from web service. Finally, Restful Web Service runs on GlassFish Server together and Android client connects to the internet to make requests to the restful web service. Another Client in server side runs the JSP program which connects to MySQL database, having rights to manipulate(insert,delete, search) data in database.

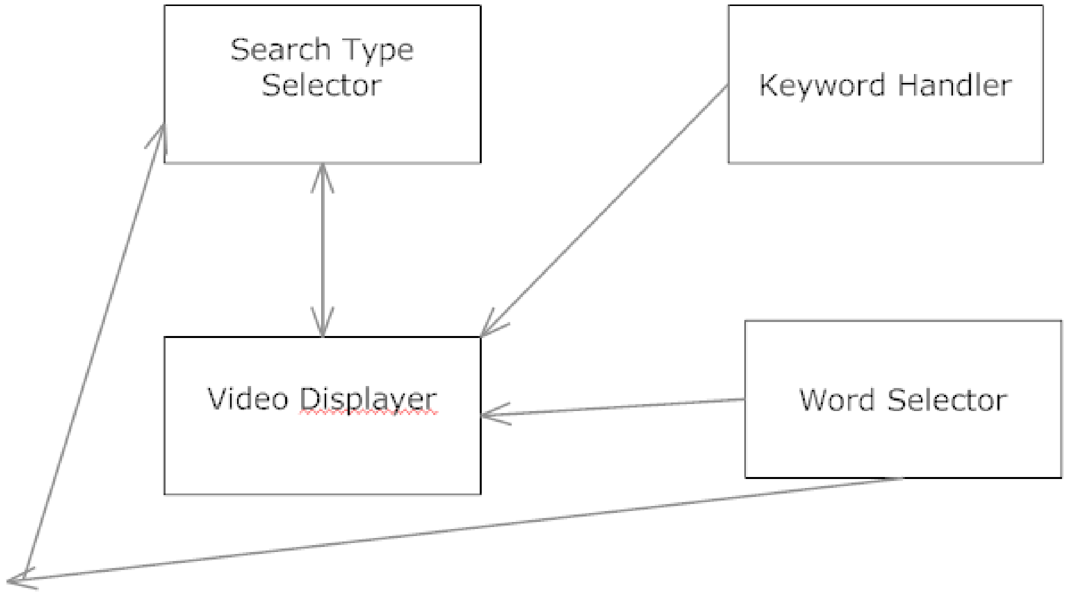
Level 0 – Context



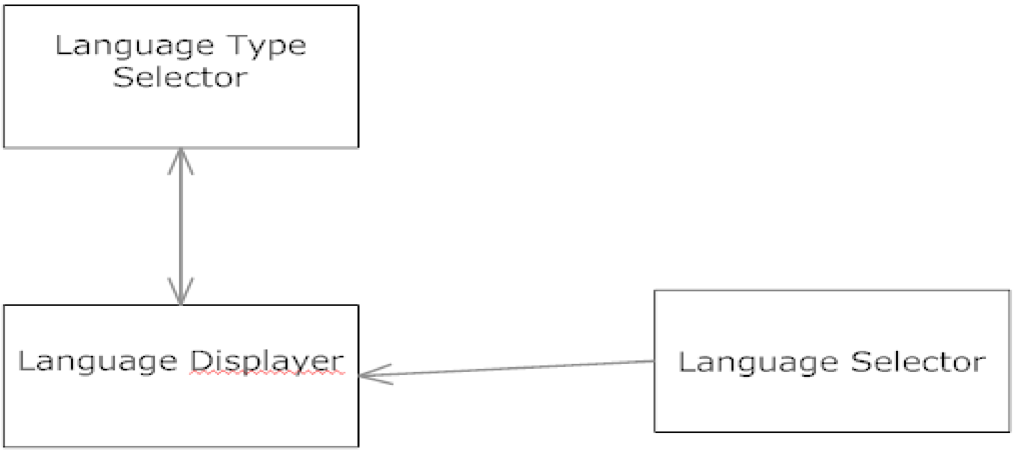
Level 1 -Repository



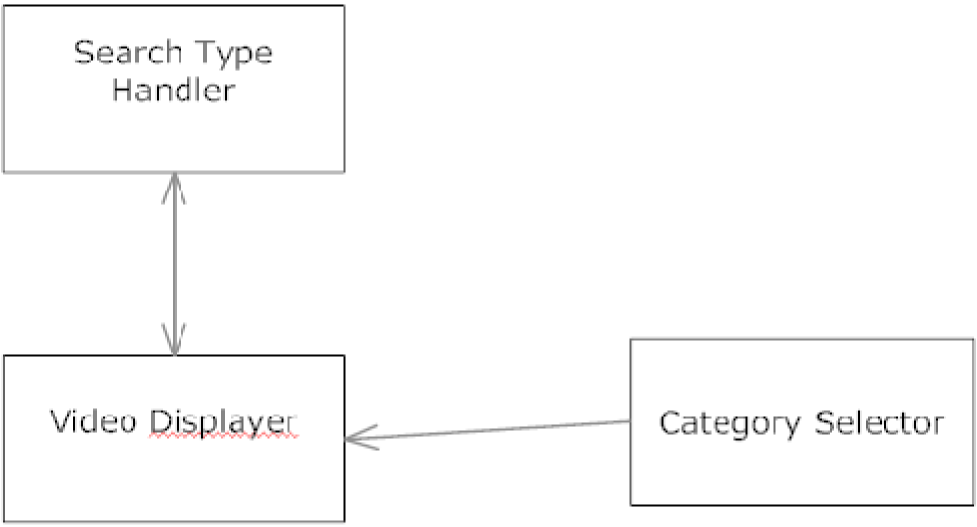
Level 2.1 Search Handler



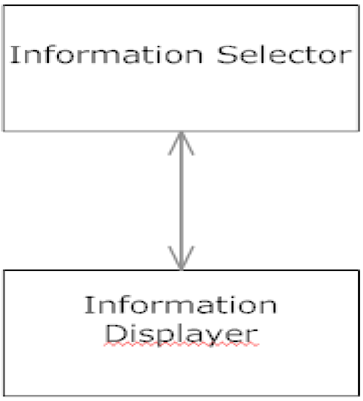
Level 2.2 Language Handler



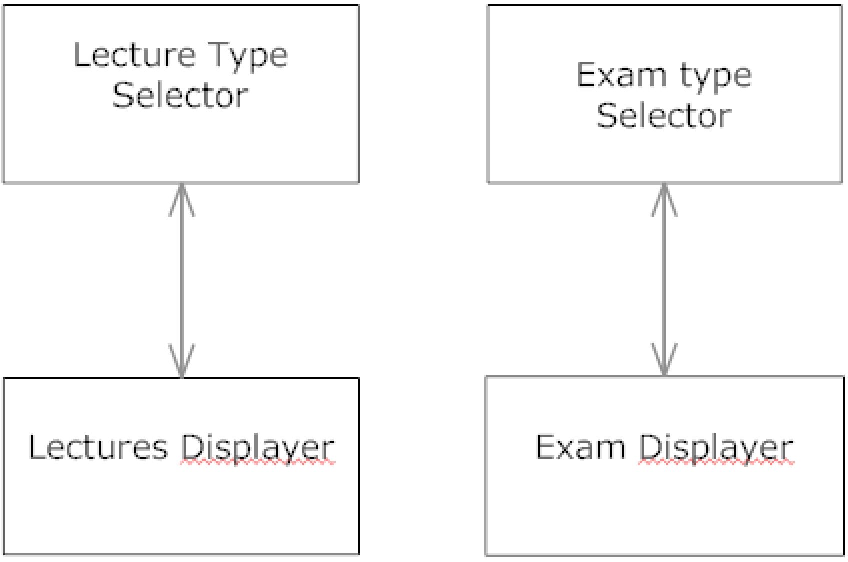
Level 2.3 Video Handler



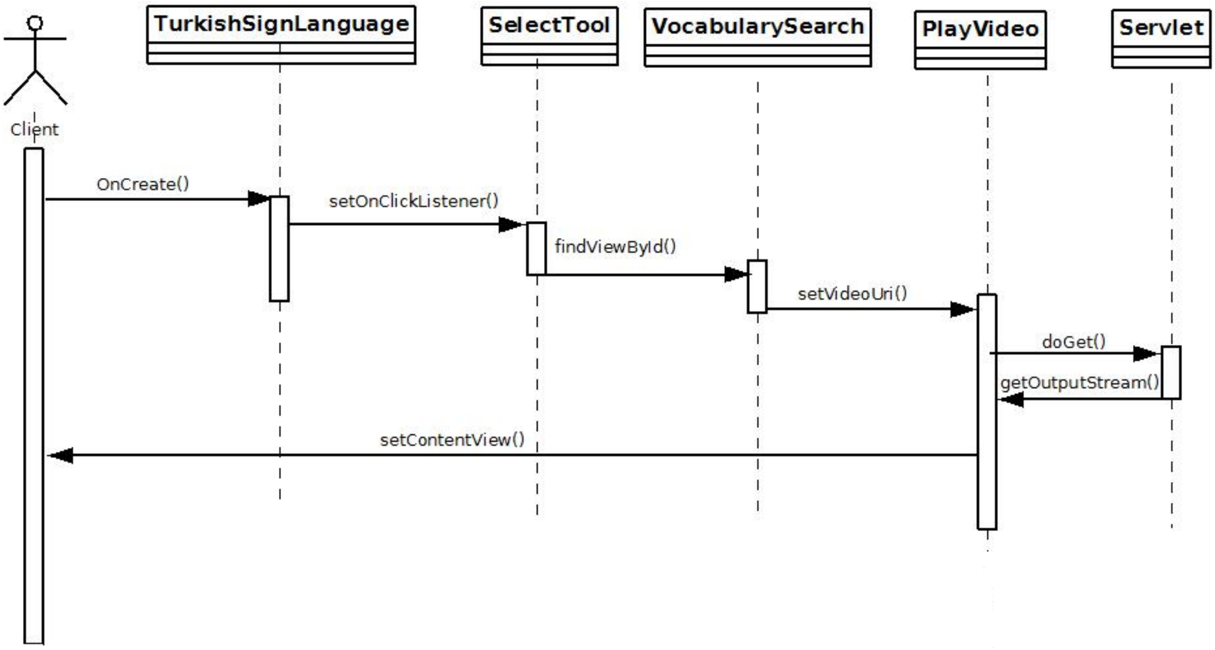
Level 2.4 Information Handler



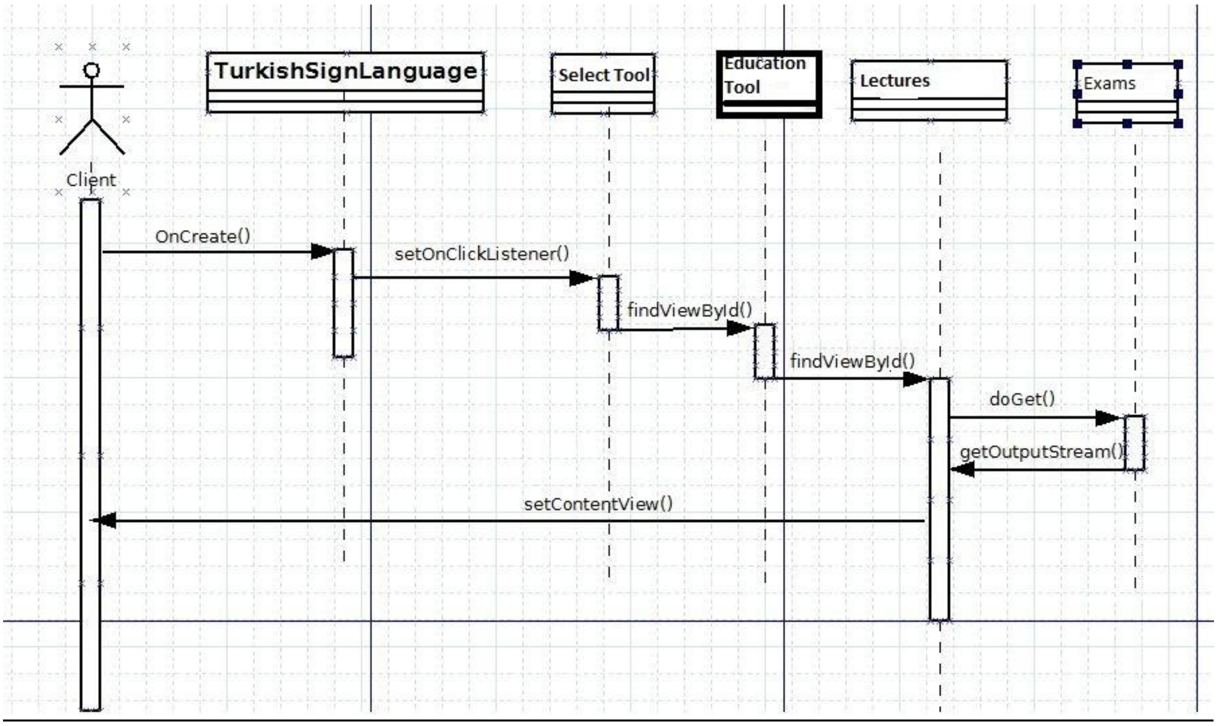
Level 2.5 Lecture Handler



Sequence Diagram for video shows:



Sequence Diagram for exams:



6.SOLUTION: Programming

Android Client Program:

We have conducted with the android programming, avatar programming and the server programming on the Java. Firstly, we started with the creating a simple android programming on java and we use the emulator on java API to see how it can be working. Then we improve our program with the new classes to open new page on android program. And we add video viewing code in to it to view videos on the new page.

Video Creation Program:

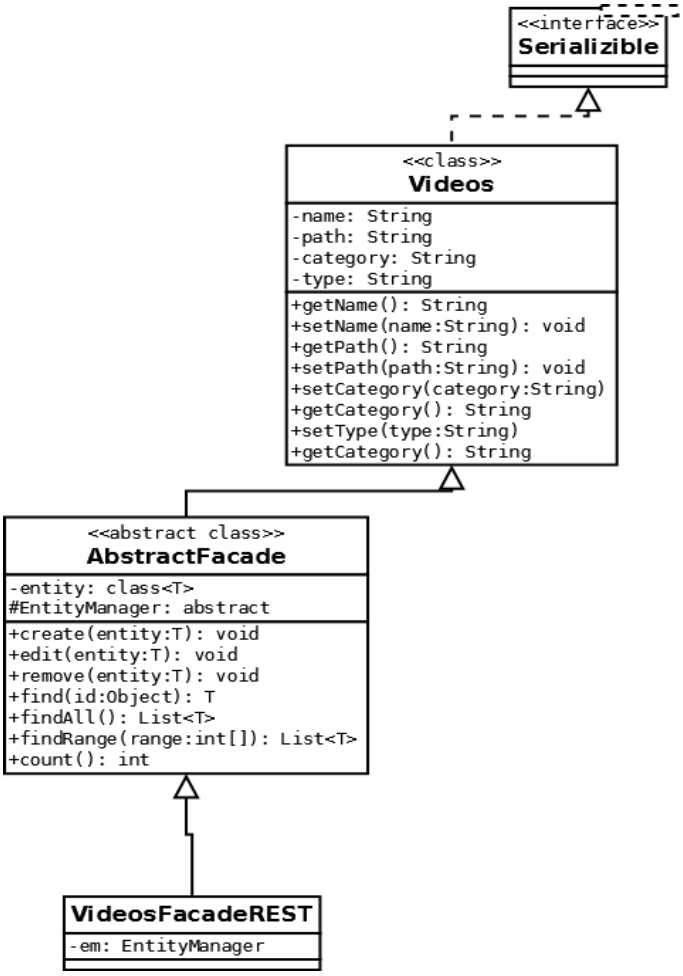
Then we have used the Vcom3D for creating avatars, there is a library on that program. While creating new avatars we use this library to create videos, while creating the videos on that library we changed the arms of the avatar person to implement sings.

Server Side Program: Web Service + MySQL +JDBC+ XAMMP Server + Glassfish Server

There are many programs running on server side. First of all, XAMMP server is used to make MySQL connection inside the program. For starting and stopping mySQL server, such server was needed to manipulate it. Afterwards, videos class is created having setter and getters of database entries. Another class from database called AbstractFacade is coded, to make queries from the database inside the program. Then, the other class which has this AbstractFacade class as super class was written as a web service, called VideosFacadeREST, to handle request(GET,POST, DELETE) from the android client. In that program, JSON format is returned when GET requests come. Restful web service is used inside the program, since it is the most suitable web service to be used from android phones.

Restful API is used inside the program and it is manually added to the library path to make program work. Jersey archive 1.17 were added to the library path by downloading it from the java web site for libraries.

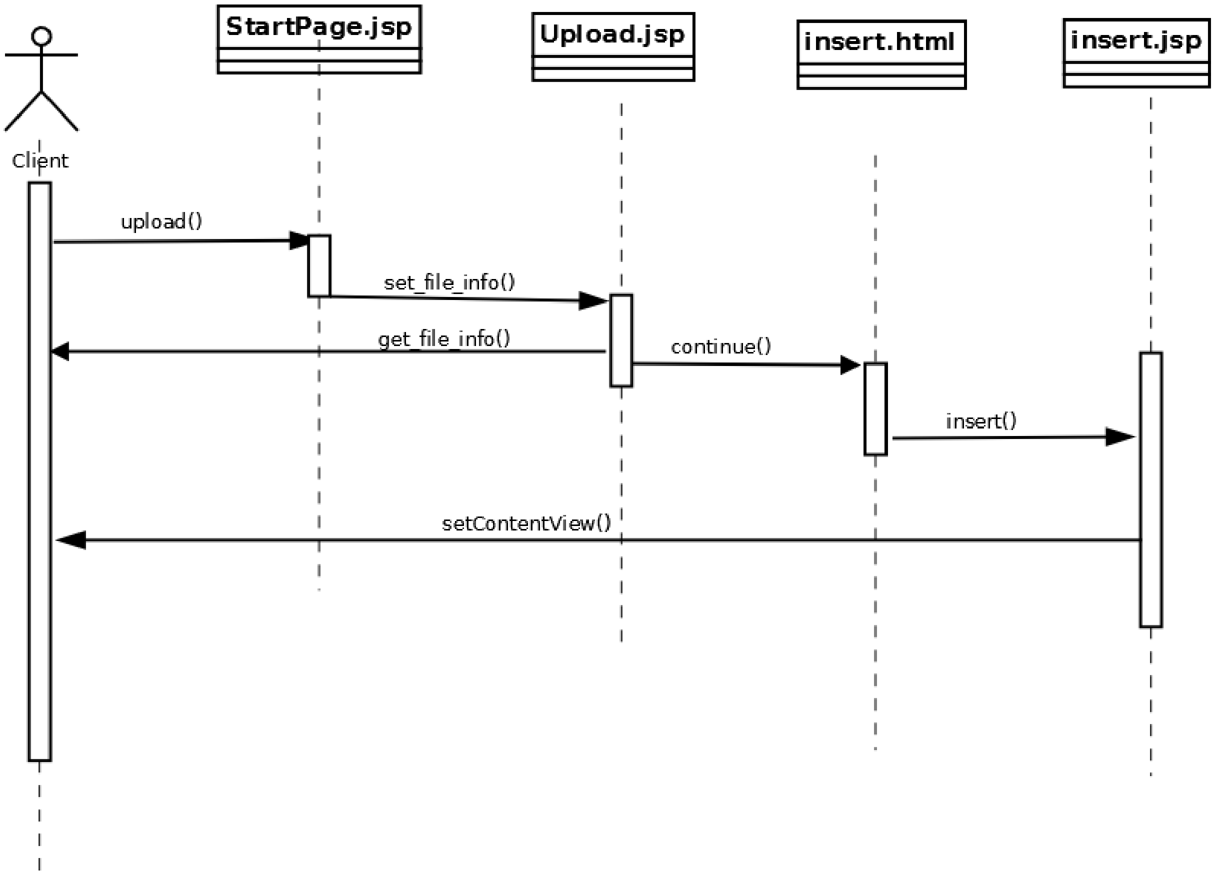
Also, Glassfish Web Server, XAMMP server and MySQL database server needs to be downloaded first to make this program work.



Server Side Program: Glassfish Server + Java Server Pages+ MySQL+JDBC

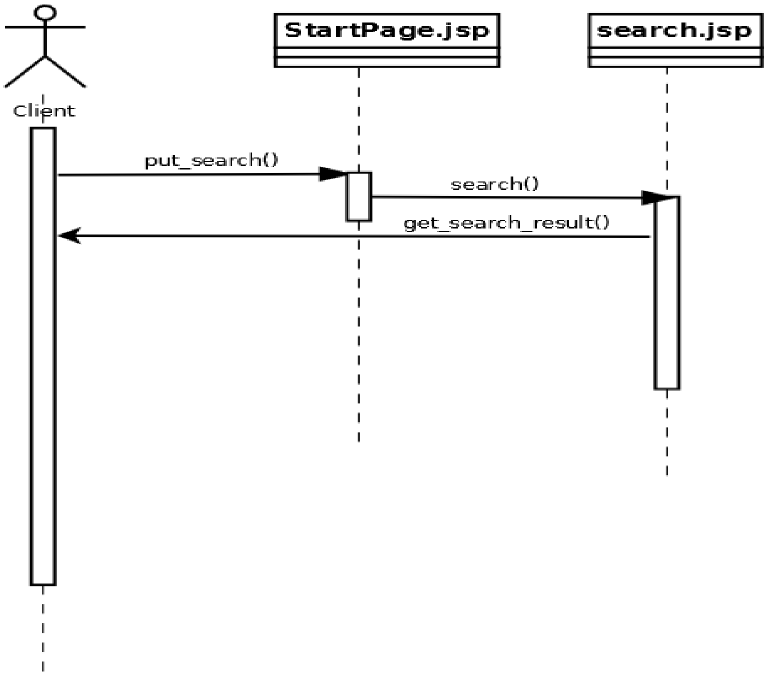
The other program were created in JSP, to make the work of server-side programmer easier. There are different files in JSP and each file calls each other. Database connection through mySQL is done in JSP. Server side user can upload the video to the server, then he/she can also insert data into the database, delete data from the database or view the content in the database. The changes are seen on web service output. Several API's are added to the library path such as mysql connectivity and file upload to make java server pages work correctly. Some HTML also used to format outlook of the program in JSP together with java.

1) Inserting data to the Database from the Server

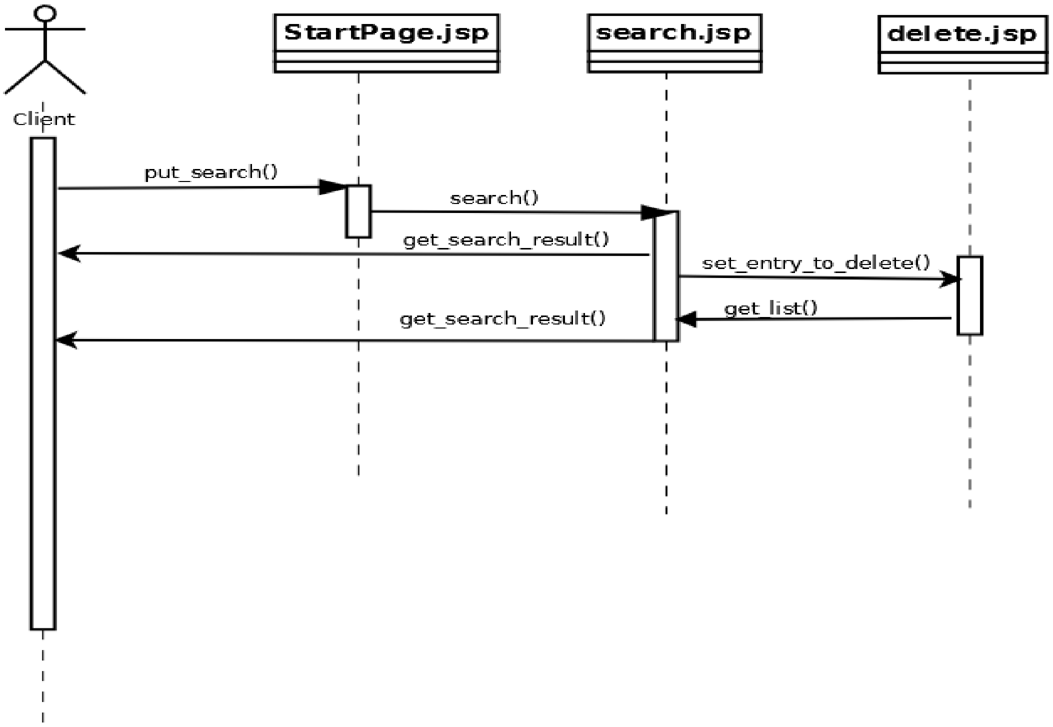


As it can be seen on the figure, server side client begins the program by uploading video. Afterwards, it gets the file content which includes information regarding size, name and type of the file. Afterwards, user continues and insert necessary information to the program to make it inserted into the database. Actually, user just enters the keyword that the file name matches and path is generated inside the program, making user less involved in entering data. User also sees a drop box which allows user to select type and category with mouse clicks, instead of entering the text. Then, the program inserts these into the database and user in server side is notified that the data is inserted into the program. Afterwards, user has two options; one option for entering new data into the database and other option for viewing the table and/or delete the data.

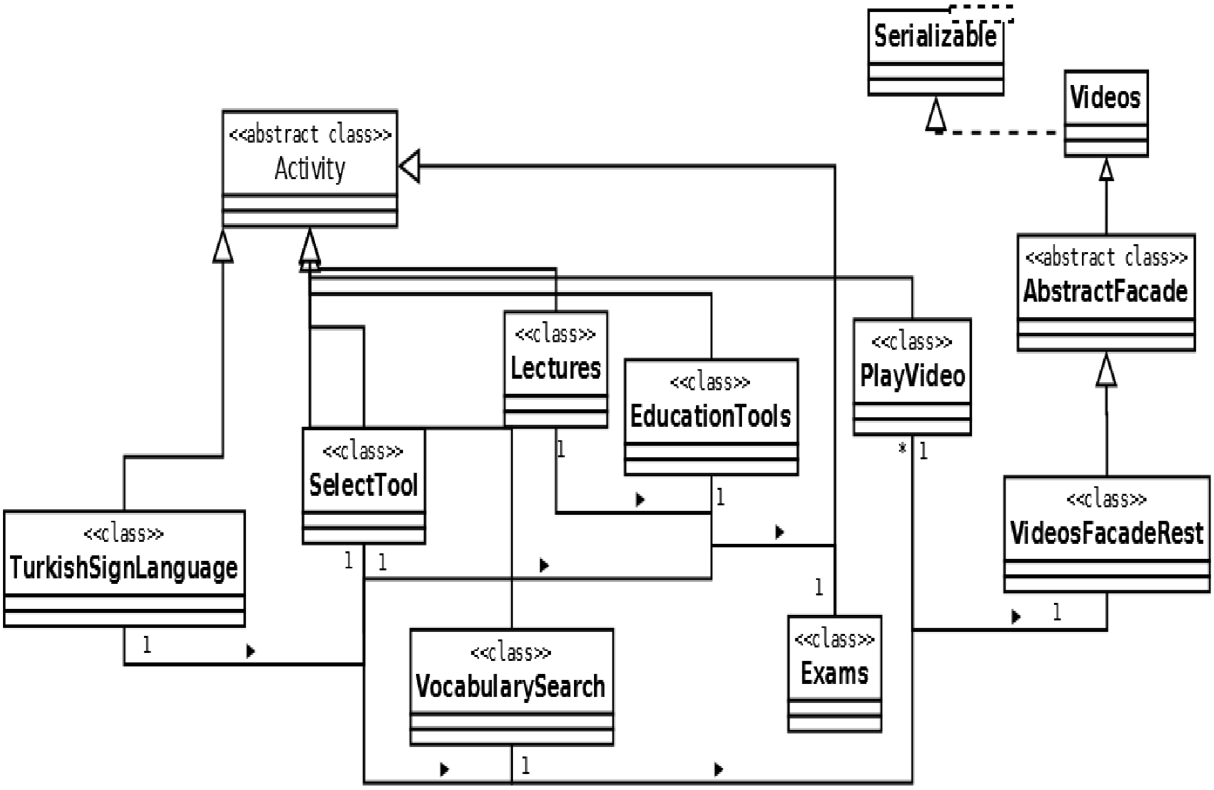
2) Search Page (Server Side)

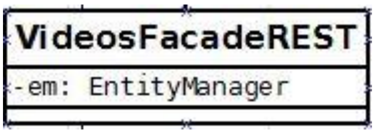


3) Delete Page (Server Side) : When the user pushes the search button, the table view is seen for the user. Afterwards, user can choose an entry to be deleted from the table from the page. This search page is updated as the user deletes a record from the table by clicking delete button next to the entry.

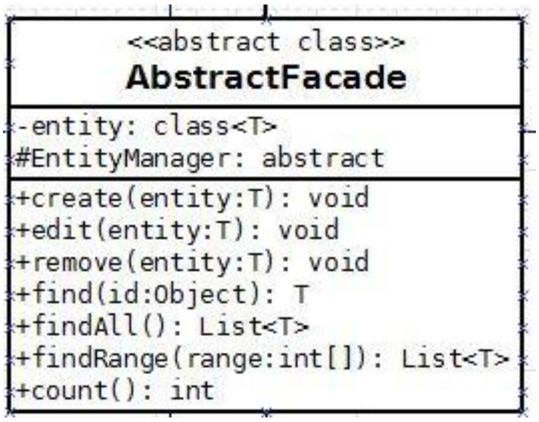


Class Diagrams: We showed that other classes are extending to the Activity class

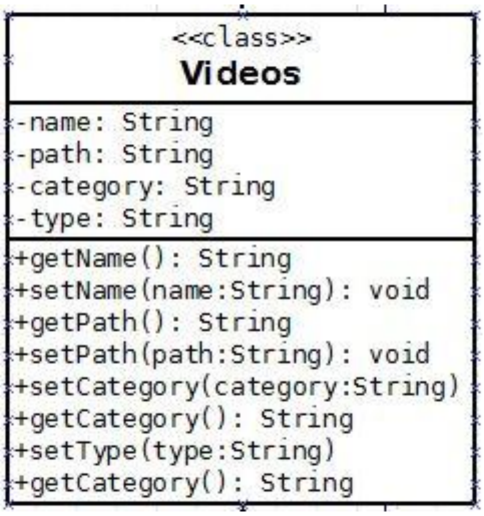




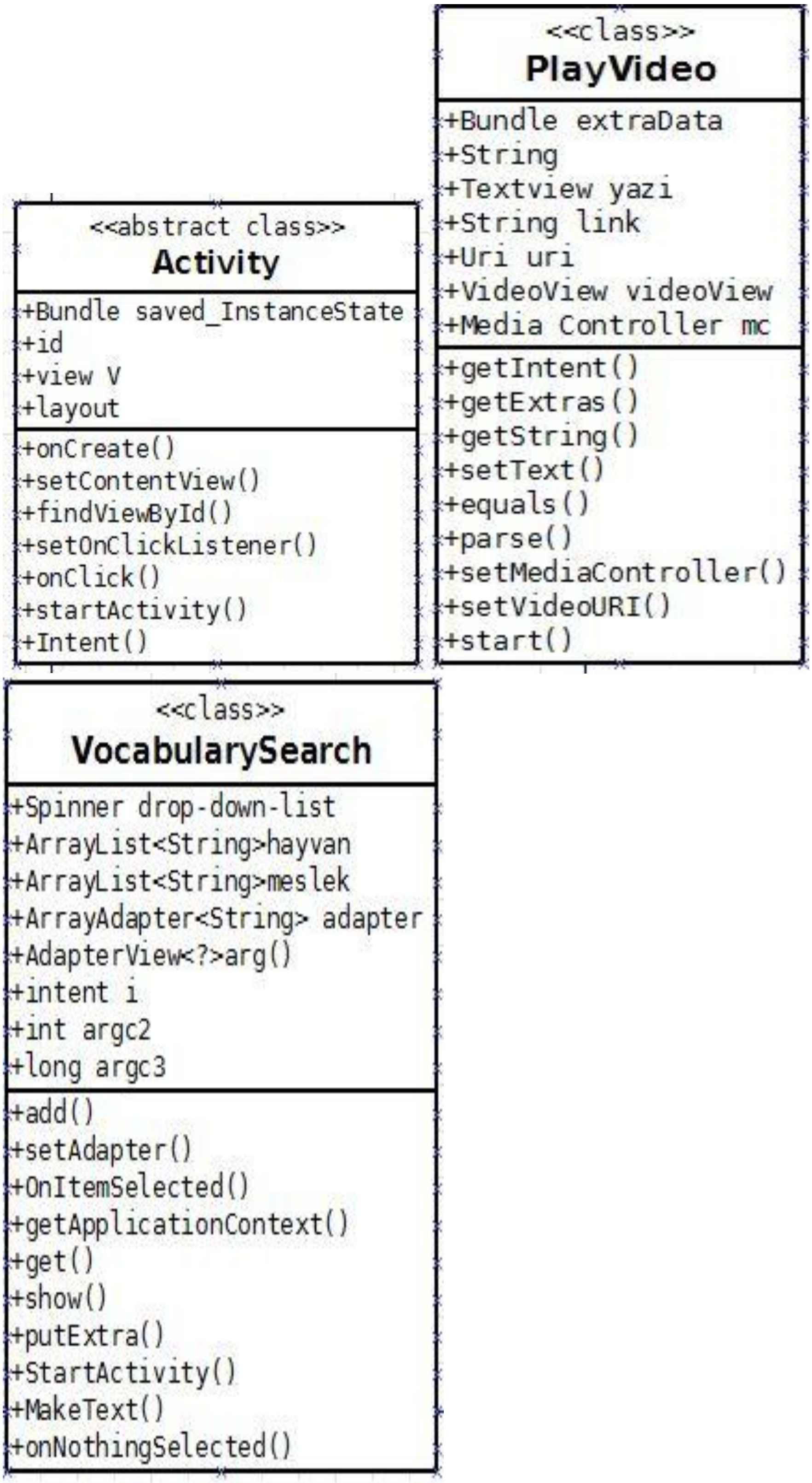
```
VideosFacadeREST
-em: EntityManager
```



```
<<abstract class>>
AbstractFacade
-entity: class<T>
#EntityManager: abstract
+create(entity:T): void
+edit(entity:T): void
+remove(entity:T): void
+find(id:Object): T
+findAll(): List<T>
+findRange(range:int[]): List<T>
+count(): int
```



```
<<class>>
Videos
-name: String
-path: String
-category: String
-type: String
+getName(): String
+setName(name:String): void
+getPath(): String
+setPath(path:String): void
+setCategory(category:String)
+getCategory(): String
+setType(type:String)
+getCategory(): String
```



<<abstract class>>
Activity

+Bundle saved_InstanceState
+id
+view V
+layout

+onCreate()
+setContentView()
+findViewById()
+setOnClickListener()
+onClick()
+startActivity()
+Intent()

<<class>>
PlayVideo

+Bundle extraData
+String
+Textview yazi
+String link
+Uri uri
+VideoView videoView
+Media Controller mc

+getIntent()
+getExtras()
+getString()
+setText()
+equals()
+parse()
+setMediaController()
+setVideoURI()
+start()

<<class>>
VocabularySearch

+Spinner drop-down-list
+ArrayList<String>hayvan
+ArrayList<String>meslek
+ArrayAdapter<String> adapter
+AdapterView<?>arg()
+intent i
+int argc2
+long argc3

+add()
+setAdapter()
+OnItemSelected()
+getApplicationContext()
+get()
+show()
+putExtra()
+StartActivity()
+MakeText()
+onNothingSelected()

7. SOLUTION: Testing

After, we implement all programs, we install this application on the android phone and we can see the program works correctly. It connected to server and plays all videos on the server. And all pages are can see on the phone. Therefore it is working correctly; there is not any problem on any part of the program.

Use Case: Vocabulary Search

1. Validate that vocabulary search screen appears on the screen

2. Validate that user clicks on the vocabulary search button

3. Validate that system shows the vocabulary screen

4. Validate that user clicks on the types of the search

5. Validate that system shows the correct search type names on the screen

6. Validate that according to the type of the search, correct video has appeared

7. Validate that user fills the search button with correct search name

8. Validate that processing monitor Works correctly.

Use Case: Educational Tools

1. Validate that educational tools screen appears on the screen

2. Validate that user clicks on the educational tools button

3. Validate that system shows the educational tools screen

4. Validate that user clicks on the educational tool types such as exams and lectures

5. Validate that system shows the correct educational tools types on the screen

6. Validate that according to the type of the educational tools, correct items has appeared

7. Validate that under exams type, system shows the multiple choice, matching and flash videos buttons

8. Validate that under lecture type, system shows the job lectures and animal lectures

9. Validate that user chooses the lectures correctly

10. Validate that system shows the lectures’ types

Use Case: **Server Side Program for Data Manipulation and Video Upload**

1. Validate that file upload is done to the correct place in server.

2. Validate that file has mp4 extension.

3. Validate that user enters the same name for keyword to be searched as the same name as file name without extension.

4. Validate that entries entered by user are inserted into the database correctly.

5. Validate that deletion option actually deletes the chosen data from the database.

6. Validate that changes in the database occurs in JSON formatted data returned by server to the client.

Testing: Restful Web Service +MySQL connection works properly.

1. Deploy and run the web service.

2. Create a test client and see the output of the web service.

3. Validate that server gives response as (Status: 200(OK))

4. Validate the output of the web service by sending GET request from test client.

5. Validate that JSON formatted data includes the database table entries correctly.

6. Validate that count works correctly inside the program showing the number of entries.

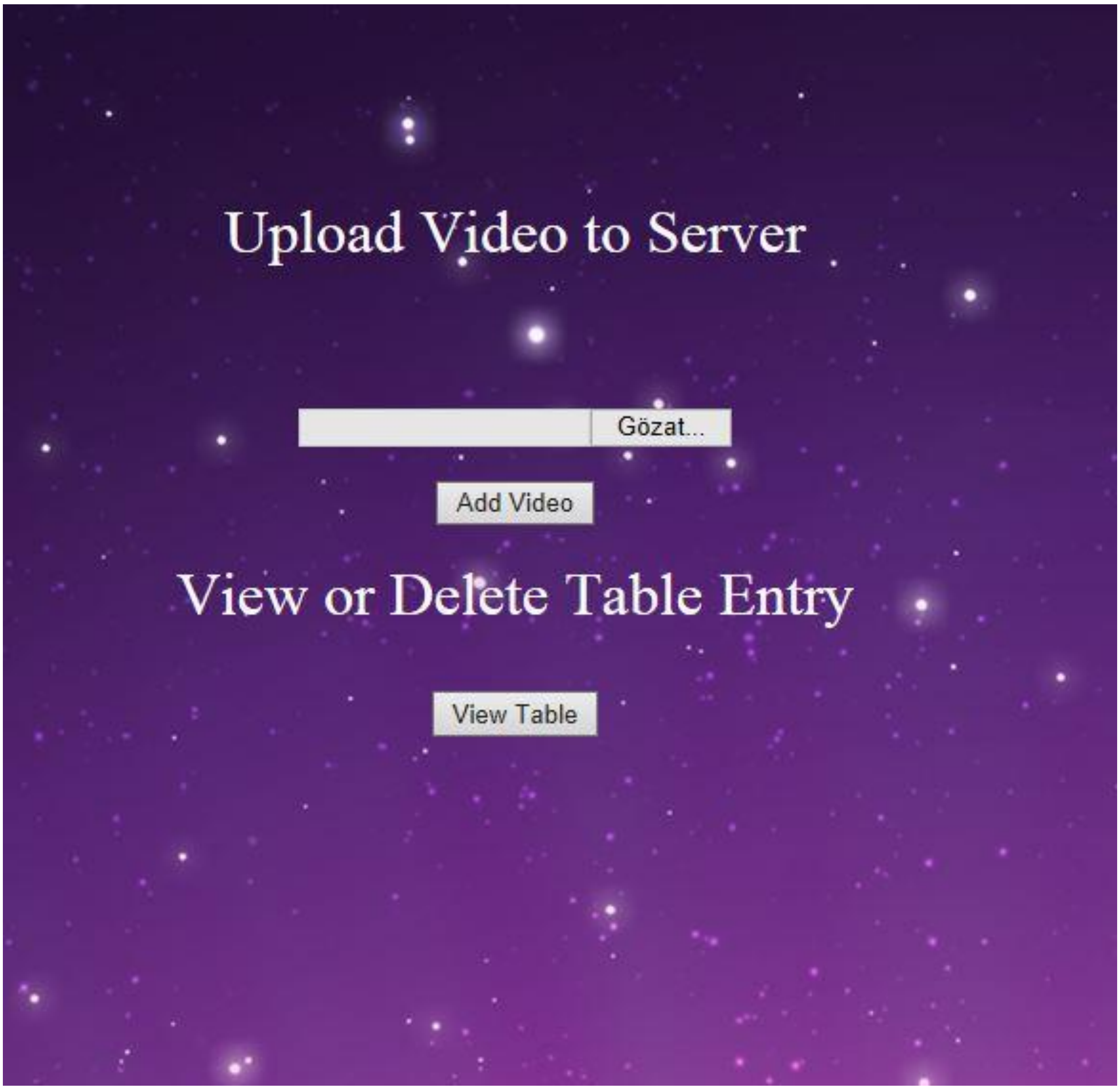
TEST RESULT from RESTFUL WEB SERVICE- Json Formatted Data

[{"name":"ARI","type":"tek","path":"http://10.144.141.82:8080/SeniorProject/videos/ari.mp4","category":"hayvan"},{"name":"ASCI","type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/asci.mp4","category":"meslek"},{"name":"ASKER","type":"tek","path":"http://10.144.141.82:8080/SeniorProject/videos/asker.mp4","category":"meslek"},{"name":"AVUKAT","type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/avukat.mp4","category":"meslek"},{"name":"AYI","type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/ayi.mp4","category":"hayvan"},{"name":"BAKAN","type":"tek","path":"http://10.144.141.82:8080/SeniorProject/videos/bakan.mp4","category":"meslek"},{"name":"DOKTOR","type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/doktor.mp4","category":"meslek"},{"name":"FARE","type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/fare.mp4","category":"hayvan"},{"name":"FIL","type":"tek","path":"http://10.144.141.82:8080/SeniorProject/videos/fil.mp4","category":"hayvan"},{"name":"GENERAL","type":"tek","path":"http://10.144.141.82:8080/SeniorProject/videos/general.mp4","category":"meslek"},{"name":"HAKEM","type":"tek","path":"http://10.144.141.82:8080/SeniorProject/videos/hakem.mp4","category":"meslek"},{"name":"HAKIM","type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/hakim.mp4","category":"meslek"},{"name":"HEMSIRE","type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/hemsire.mp4","category":"meslek"},{"name":"HINDI","type":"tek","path":"http://10.144.141.82:80

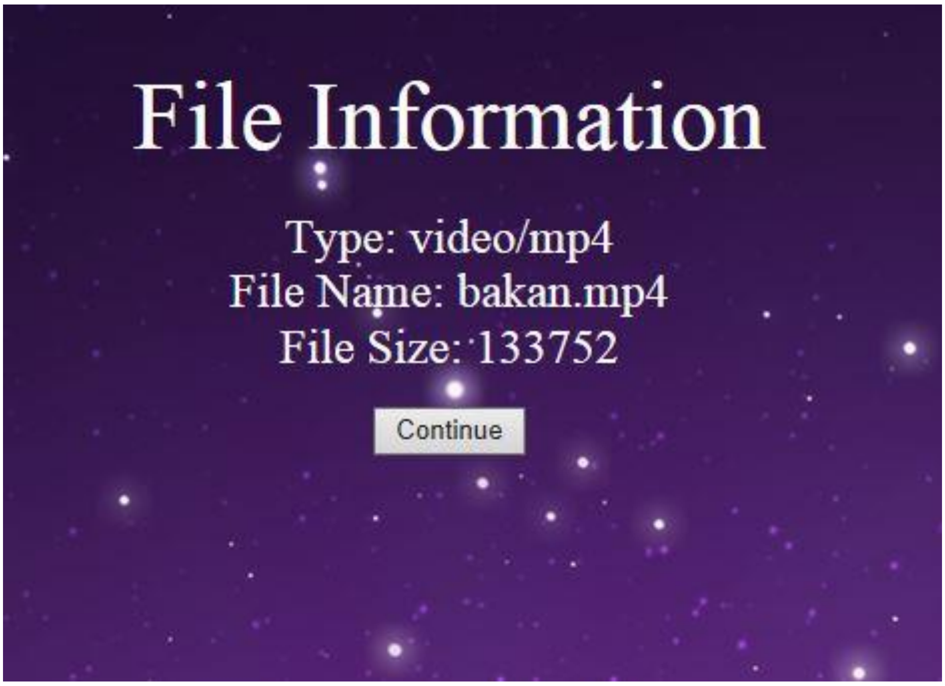
80/SeniorProject/videos/hindi.mp4","category":"hayvan"},{"name":"INEK","type":"tek","path":"h
ttp://10.144.141.82:8080/SeniorProject/videos/inek.mp4","category":"hayvan"},{"name":"ISCI","
type":"tek","path":"http://10.144.141.82:8080/SeniorProject/videos/isci.mp4","category":"mesle
k"},{"name":"ISIRMAK","type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/isir
mak.mp4","category":"fiil"},{"name":"KAPLAN","type":"cift","path":"http://10.144.141.82:8080/S
eniorProject/videos/kaplan.mp4","category":"hayvan"},{"name":"KAZ","type":"cift","path":"http:
//10.144.141.82:8080/SeniorProject/videos/kaz.mp4","category":"hayvan"},{"name":"KECI","type
":"tek","path":"http://10.144.141.82:8080/SeniorProject/videos/keci.mp4","category":"hayvan"},
{"name":"KEDI","type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/kedi.mp4",
"category":"hayvan"},{"name":"KELEBEK","type":"cift","path":"http://10.144.141.82:8080/Senior
Project/videos/kelebek.mp4","category":"hayvan"},{"name":"KOYUN","type":"cift","path":"http:
//10.144.141.82:8080/SeniorProject/videos/koyun.mp4","category":"hayvan"},{"name":"KUZU","
type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/kuzu.mp4","category":"hayv
an"},{"name":"MIMAR","type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/mi
mar.mp4","category":"meslek"},{"name":"POLIS","type":"cift","path":"http://10.144.141.82:8080
/SeniorProject/videos/polis.mp4","category":"meslek"},{"name":"SUBAY","type":"tek","path":"ht
tp://10.144.141.82:8080/SeniorProject/videos/subay.mp4","category":"meslek"},{"name":"TERZI"
,"type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/terzi.mp4","category":"me
slek"},{"name":"TILKI","type":"cift","path":"http://10.144.141.82:8080/SeniorProject/videos/tilki.
mp4","category":"hayvan"},{"name":"VALI","type":"cift","path":"http://10.144.141.82:8080/Seni
orProject/videos/vali.mp4","category":"meslek"},{"name":"ZABITA","type":"tek","path":"http://
10.144.141.82:8080/SeniorProject/videos/zabita.mp4","category":"meslek"}]

SERVER SIDE JSP PROGRAM SCREEN OUTPUTS

StartPage.jsp



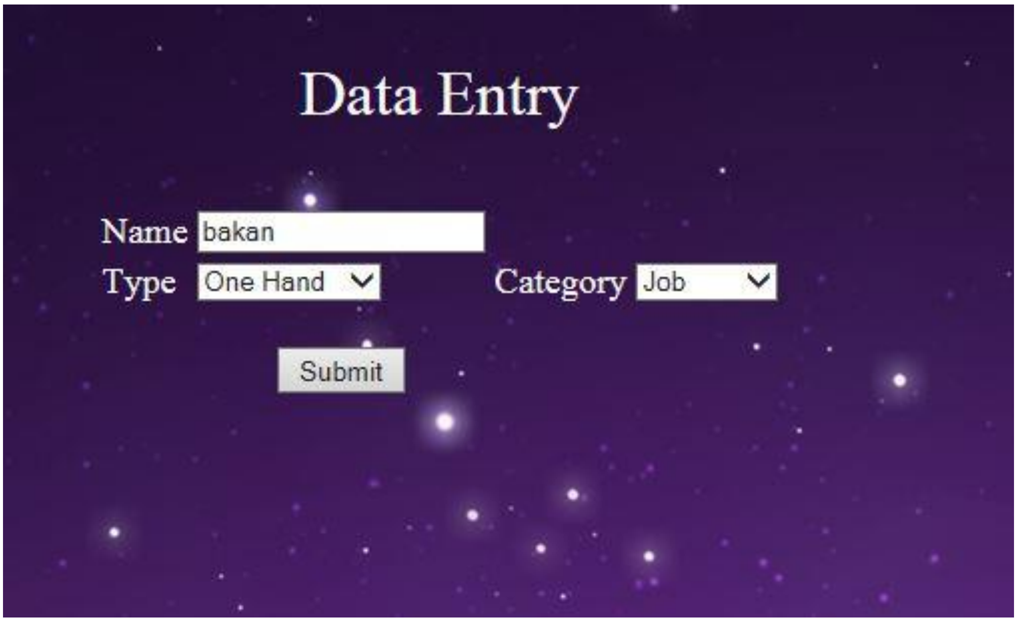
Upload.jsp



File Information

Type: video/mp4
File Name: bakan.mp4
File Size: 133752

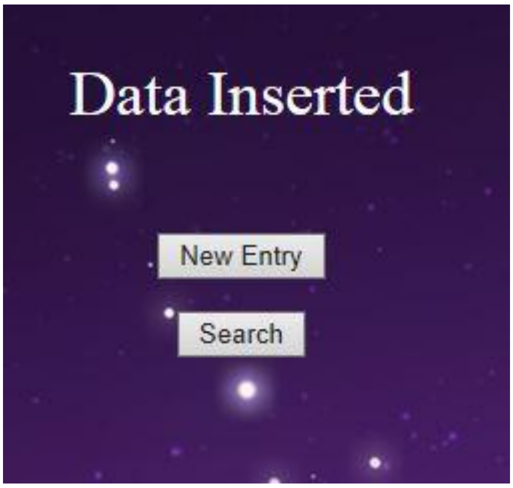
insert.html



Data Entry

Name bakan
Type One Hand
Category Job

insert.jsp



Data Inserted

search.jsp

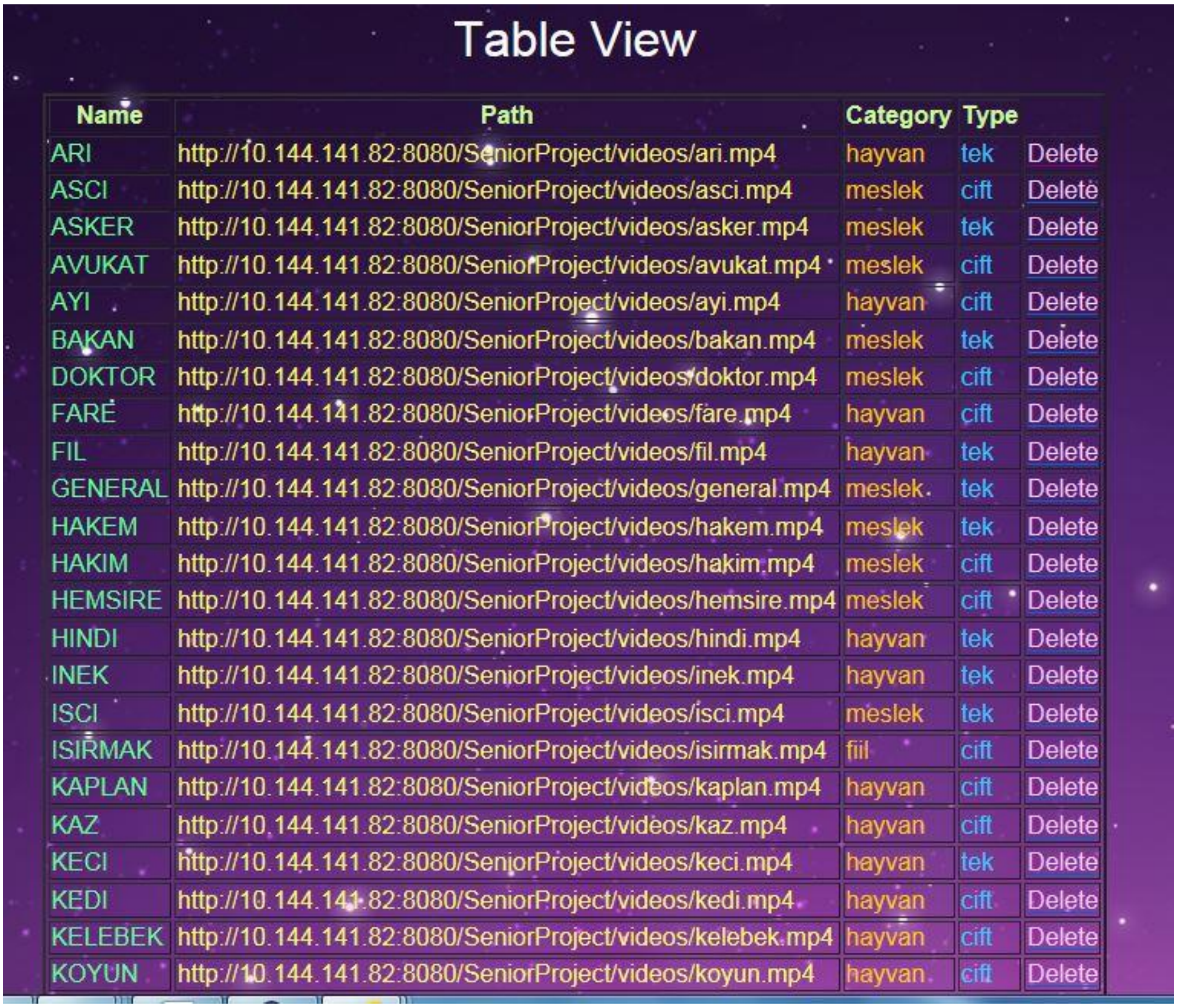


Table View

Name	Path	Category	Type	
ARI	http://10.144.141.82:8080/SeniorProject/videos/ari.mp4	hayvan	tek	Delete
ASCI	http://10.144.141.82:8080/SeniorProject/videos/asci.mp4	meslek	cift	Delete
ASKER	http://10.144.141.82:8080/SeniorProject/videos/asker.mp4	meslek	tek	Delete
AVUKAT	http://10.144.141.82:8080/SeniorProject/videos/avukat.mp4	meslek	cift	Delete
AYI	http://10.144.141.82:8080/SeniorProject/videos/ayi.mp4	hayvan	cift	Delete
BAKAN	http://10.144.141.82:8080/SeniorProject/videos/bakan.mp4	meslek	tek	Delete
DOKTOR	http://10.144.141.82:8080/SeniorProject/videos/doktor.mp4	meslek	cift	Delete
FARE	http://10.144.141.82:8080/SeniorProject/videos/fare.mp4	hayvan	cift	Delete
FIL	http://10.144.141.82:8080/SeniorProject/videos/fil.mp4	hayvan	tek	Delete
GENERAL	http://10.144.141.82:8080/SeniorProject/videos/general.mp4	meslek	tek	Delete
HAKEM	http://10.144.141.82:8080/SeniorProject/videos/hakem.mp4	meslek	tek	Delete
HAKIM	http://10.144.141.82:8080/SeniorProject/videos/hakim.mp4	meslek	cift	Delete
HEMSIRE	http://10.144.141.82:8080/SeniorProject/videos/hemsire.mp4	meslek	cift	Delete
HINDI	http://10.144.141.82:8080/SeniorProject/videos/hindi.mp4	hayvan	tek	Delete
INEK	http://10.144.141.82:8080/SeniorProject/videos/inek.mp4	hayvan	tek	Delete
ISCI	http://10.144.141.82:8080/SeniorProject/videos/isci.mp4	meslek	tek	Delete
ISIRMAK	http://10.144.141.82:8080/SeniorProject/videos/isirmak.mp4	fiil	cift	Delete
KAPLAN	http://10.144.141.82:8080/SeniorProject/videos/kaplan.mp4	hayvan	cift	Delete
KAZ	http://10.144.141.82:8080/SeniorProject/videos/kaz.mp4	hayvan	cift	Delete
KECI	http://10.144.141.82:8080/SeniorProject/videos/keci.mp4	hayvan	tek	Delete
KEDI	http://10.144.141.82:8080/SeniorProject/videos/kedi.mp4	hayvan	cift	Delete
KELEBEK	http://10.144.141.82:8080/SeniorProject/videos/kelebek.mp4	hayvan	cift	Delete
KOYUN	http://10.144.141.82:8080/SeniorProject/videos/koyun.mp4	hayvan	cift	Delete

Configurations for Server Side Programs

Restful web service, jsp, MySQL connection

1. First of all, MySQL server(port:3306) and XAMMP server needed to be installed into the computer.

2. After that, web application from java web project needs to be chosen in Netbeans. While creating the project, server needs to be defined. If Glassfish server is not installed, it needs to be installed via interface that comes. Afterwards, it is chosen.

3. Then, we need to manage database connection to connect to MySQL server.

4. Going through services->database->register MySQL server.

5. After that, new window opens and username and password with correct port name that the server is running must be written. A table inside the MySQL server firs needs to be created including name, path, category and type as varchar variables being name as primary key.

6. Under admin properties, XAMMP server's exe files for start and stop needs to be put to the appropriate place. (XAMMP server needs to be manually started afterwards)

7. Then, we connect to the database by connect option came from right clicking on server connection. After we connect, we can refer to the table in database from our program.

8. Then, we need to add Rest libraries to our projects for web service run. For jsp files run, we need to add other files such as file upload etc. to the library path under projects options. (This library files are in SeniorProject/web/build/WEB-INF/lib path.)

9. Then, program files can be copied to source files (for java files) and web pages (for jsp files).

10. Videos are put under web pages folder as a folder.

11. Images folder is also put under web pages, since jsp files use an image as a background inside the program.

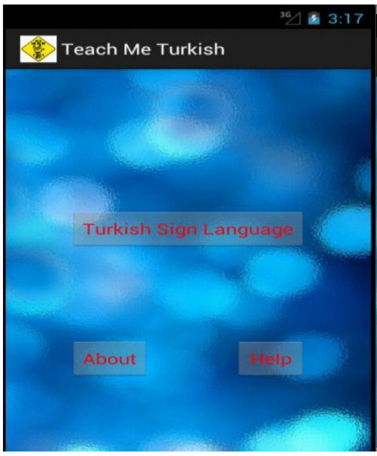
12. The information regarding the path of web service will be in web.xml after built. It can also be manually configured. In our project, it is /resources.

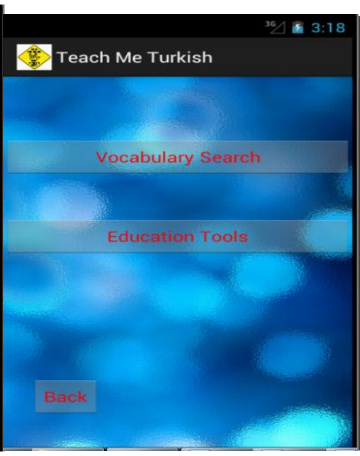
13. Build, deploy and run the project.

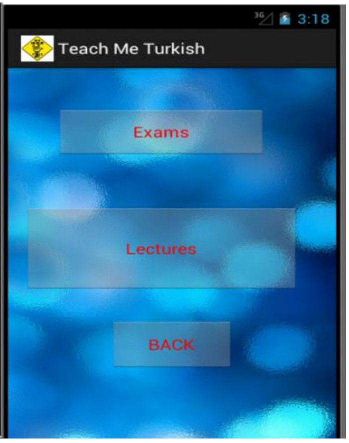
In addition to all of these, MySQL server needs to be configured to give rights to update and delete operations externally by changing server options. If it is not configured, jsp files cannot access the database to make deletions. Since the username, database name, password and file path for videos changes from computer to computer, these changes must be reflected on the upload, insert, search and delete.jsp files. Also, since the path is generated by the program based on keyword, user may need to change the web address that links the pages inside the code and any other path in the programs for the usage of other users.

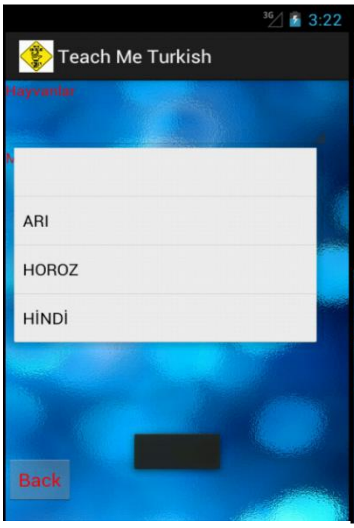
ANDROID PROGRAM SCREEN OUTPUTS:

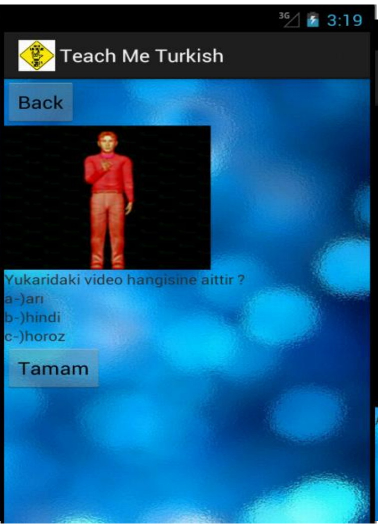
We also tested our application on Android emulator whether is working correctly or not, below our full **screenshots:**



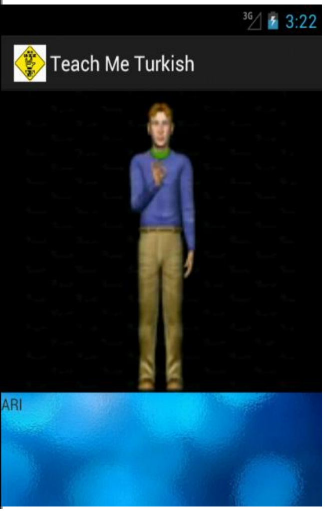








Yukaridaki video hangisine aittir ?
a-)arı
b-)hindi
c-)horoz



8. CONCLUSION and FUTURE WORK:

In conclusion, a database is created behind the web service and web service was connected to android client to make requests from the database through web service. Also, a GUI is created in server side to manipulate data in server and database. In addition to that, android application is developed, containing lectures and education tools.

Program is installed to the android phone, tested and works with no problem.

This program can be extended to different applications such as natural language processing or development of new applications for hearing impaired people.

REFERENCES

[1] http://www.vcom3d.com/

[2] http://www.blender.org/

[3] http://www.makehuman.org/

[4] http://www.vsign.nl/

[5] http://www.cmpe.boun.edu.tr/pilab/tidsozlugu/

[6] http://turkisaretdili.ku.edu.tr/en/wordlist.aspx