

**STUDENT INFORMATION AND ACTIVITY
SYSTEM**

BY

MURAD CENGİZ

BACHELORS DEGREE

EPOKA UNIVERSITY

2014

STUDENT INFORMATION AND ACTIVITY SYSTEM

BY

MURAD CENGIZ

Thesis submitted to Faculty of Architecture and Engineering, Epoka University, in
fulfilment of the requirement for the degree of bachelor science

May 2014

DEDICATION

To my dear family for all their support and guidance.

ABSTRACT

Faculty of Architecture and Engineering

Advisor: Igli Hakrama

Pre-College educations tend to be dull when it comes to information and tracking of the student, technologies that provide such services ease our daily lives and saves resources.

Information is a really important aspect of our lives, if you have information about something you can have a clearer understanding on what you should do about it.

Missing such information on students be it the parent, or teacher – in forms of feedback-supervisors are missing a core information about the student during the time of their life which child's are most effected in. If a supervisor is missing such information, and the student doesn't have an idea of what his/her actions will cause in the long run, it may already be too late when the effect leaves a mark on the student. An action should be done when a student performs poorly or does a good action, there should be constant motivation and a constant interaction with the student to get feedback.

However a student of that age tends to have focus and attention problems, thus it's hard to get continuous feedback from such a student. A system that keeps the focus and attention of the students and also give feedback to the supervisor and/or the students – including good/bad behavior, student grades- would solve the big problem of tracking

the students on time. The main focus of this system is to keep the attention of the students and help them co-operate with their supervisor. Such co-operation is provided making a course interactive with live sessions which provide constant teacher-student interaction throughout the course.

An important piece of information gained during the planning and designing of the system was that a bond between the teacher and the working staff was also created. Parents and teacher would each share their point of view and construct a better understanding of the student. Additional features to help them communicate isn't developed as is considered as a high priority update.

ACKNOWLEDGEMENTS

I have to thank and acknowledge my father for guiding me with his wisdom. I also have to thank my advisor and supervisor Igli Hakrama for all his help and guidance.

DECLARATION

I hereby declare that the thesis is based on my original work except for quotations and citations that I have duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Epoka University or other institutions.

Murad Cengiz

9 June 2014

TABLE OF CONTENTS

Table of Contents

DEDICATION	i
ABSTARCT	ii
ACKNOWLEDGEMENTS	iv
DECLARATION.....	v
CHAPTER 1.....	1
INTRODUCTION	1
CHAPTER 2.....	5
LITERATURE REVIEW	5
2.1 Information Systems	5
2.2 Development of an Information System.....	6
2.3 Accessibility of an Information System.....	9
2.4 Technologies Used.....	10
2.4.1 MySQL vs Oracle.....	11
2.4.2 Application vs Website	11
CHAPTER 3.....	13
Software Analysis and Design.....	13
3.1 Functional Requirements.....	13
3.1.1 Usability Requirements.....	13
3.1.2 Performance Requirements	14
3.1.3 Supportability Requirements	14
3.1.4 Security Requirements.....	15

3.2 Non-Functional Requirements	15
3.2.1 Hardware Requirements	15
3.2.2 Software Requirements	15
3.2.3 Security Requirements.....	16
3.3 User Profiles.....	16
3.3.1 Student	16
3.3.2 Parent	19
3.3.3 Teacher	21
3.4 System Design	23
CHAPTER 4.....	31
Implementation	31
4.1 Technologies Used.....	31
4.1.1 CSS3	31
4.1.2 Database structuring	32
4.1.3 Server side scripting.....	34
4.1.4 JQuery.....	34
CHAPTER 5.....	37
Conclusions and future work	37
APPENDIX A.....	38
DIAGRAMS	38
References.....	44

LIST OF FIGURES

Figure 1: System development life cycle	Pg. 8
Figure 2: Use Case diagram of a student	Pg.18
Figure 3: Use case diagram of a parent	Pg.19
Figure 4: Use Case diagram of the teacher	Pg.21
Figure 5: Activity diagram of a student	Pg.23
Figure 6: Activity diagram of grade insertion	Pg.24
Figure 7: Sequence diagram of grade insertion	Pg.25
Figure 8: Class diagram of the system	Pg.27
Figure 9: System component diagram	Pg.28
Figure 10: Deployment diagram of the system	Pg.29

CHAPTER 1

INTRODUCTION

Humans by nature, try to make their lives easier in an everlasting cycle. This extends from simply inventing small inventions, building machines and even simply learning how to live a better life. Thus technology grows, people develop new applications every passing day to make access information, making calculations, saving data and many more.

One of these type of applications were Enterprise Resource Planning (ERP) applications. ERP is a system that helps manage a business be it via storing information, sharing data amongst members of the organizations, making calculations, planning events and so forth.

A type of an organization that was in need of such a system were schools. Students need a way to access to see their grade informations, be it their quiz score, attendance, discipline score and so on. However this falls short on pre-college schools, student doesn't have the same mindset as of an adult, and depending on the student he/she may or may not care about his/her education. Thus such information may not be providing its fullest efficiency.

A student of such age requires a lot of attention, it's a time that peer grouping effects them immensely. Such relations most of the time harm the

student since negative attitude is passed along easier than the good one. A parent, a good one or a bad one, can only do as much as he can after the student leaves the school. They can't have a clear understanding/knowledge of the current state of the student on the fly. The only way for the parent to gain access to such information is 1, to either go to the school or talk to their teachers 2, contact them online via phone or email. Both of these options are not as dynamic as it could've been. A teacher and/or the parent may not be available at all times, a phone call or an email is unreliable and it would overburden the teacher with an excessive amount of phone calls made by the teacher. If a crucial event happened to a child this information should be shared in real time with the parent, since an action should be done as soon as possible, it may be too hard to fix the bad habits a student may have inherited through his/her peer group later on.

A parent should also know more about his/her child, a parent should not only know the final grade of the student but also know any grade the student might have taken before as well. If a bad grade is encountered a reaction should be made on the fly and not right before the finals and/or before a parent meeting is arranged. If a student also commits a bad behavior resulting on a loss on disciplinary score the parent should have immediate notification about such information. In the same way a student should also be motivated to do good, meaning if a student gets a good grade and/or

commits a good behavior the parent should know about that particular event as soon as possible and reward his/her child accordingly.

However we should keep in mind that a child has focus and attention issues, to be able to get required feedback about the student an active interaction should be created between the students the teacher and the parent. A class in pre-college school tends to be really dull. The lessons should be more appealing and there should be more motivation to the student so it's easier for them to co-operate with their supervisor. A child at that age has a really low focus time, a child switches focus and pays attention to different things very rapidly, it is hard to control such students and make them pay attention to the lesson. This means dull lessons such as a teacher joining the lesson explaining the lesson and leaving should not happen. This brings up solutions such as interactive classes, if course material, quizzes, assignments and others are presented in a way that is appealing to the student, there is a higher chance that the student will remain focused for a longer time.

We shouldn't pass without mentioning that we, as humans tend to forget quite often, especially students therefore if the teacher or the student have an opinion during the class about a student or a teacher, it should be noted without being forgotten, and this could also remain as evidence to certain periods of time. Such notes give better information on student's status in the school.

To be able to get such interactivity a student should be motivated to use the system, and also be motivated towards the lecture. A student should be rewarded for good, and be punished for bad behavior. These rewards and punishments should be on the moment changes, and should increase or decrease student score accordingly. If a student with high student score is not rewarded or a student which repeatedly commits bad behavior and is not punished, students will not be motivated by rewards and be effected by bad behavior of others. Such is unhealthy for the education and the students personality overall.

CHAPTER 2

Literature Review

2.1 Information Systems

Information Systems is the single most used system type throughout the world be it on schools, businesses, hotels, libraries and every other company you can think of. A dictionary definition of an information system would be: “system consisting of the network of all communication channels used within an organization” [1]. Or “an integrated set of components for collecting, storing, and processing data and for delivering information, knowledge, and digital products.” [2]. such information system is used to have clearer idea on what is going on in the company and/or organization. For example in a trading company, data such as items remaining, items being shipped, item costs, customer payments and so forth could be stored. Such information can be accessed by any related member of the information system.

These systems have shown to greatly increase the performance, success and efficiency of a company. This is mostly achieved due to reduced amount of human work put towards handling data; most of the small errands are run by the system. This dramatically reduces the amount of effort put into

managing data, especially if the company is huge and consists of a high consumer base we could be looking at thousands of customers, imagine if you wanted to send mass message to everyone containing an item X, such actions should not be done manually and should be handled by a system. This means instead of wasting resources on a worker to do those jobs manually you can invest that resource into something else. This means 2 things; 1, the efficiency of the company is improved since you have a better way of managing resources 2, you have reduced cost since you need less resources now.

Most information systems consist of two member groups, producers and consumers. Producers set up the system, set up necessary information the system may require and/or the system will operate on for example: item types, item prices, item details and so forth. Producers and consumers have different access to different features, for example a producer can add an item and set its details, however a consumer can only see the item and its details. A consumer usually benefits from services provided by the system. These services depend on the company; these services can vary from purchasing an item or a service from the company or simply see information provided by them.

2.2 Development of an Information System

The biggest requirement of an information system is that it should be available to the public or at the very least available to the relevant user base. This means the system should be placed on a network that is accessible by the user base, the recommended network for such system is the Internet, it is crucial that the information system is accessible by all the users, be it the consumers or the producers. The server will contain almost all the information about the system and its components and the data inside it, which means if this server goes down, the performance and the efficiency of the company takes a big hit. Due to all this, typically, an ideal Information system is stored to a server, which is powerful enough and accessible. Such a server is recommended to be local and is recommended to host to the Internet via that server.

It is extremely important that such system is extendable with modules, an information system is bound to have changes throughout its development cycle and even after it completes its development cycle.

Such a development cycle is also known as, "The systems development life cycle (SDLC) is a conceptual model used in project management that describes the stages involved in an information system development project, from an initial feasibility study through maintenance of the completed application."

Below you can see the life cycle of the said system:

Planning: “Establishes a high-level view of the intended project and determines its goals.” [4]

Analysis: “Refines project goals into defined functions and operation of the intended application. Analyzes end-user information needs.” [4]

Design: “Describes desired features and operations in detail, including screen layouts, business rules, process diagrams, pseudo code and other documentation.” [4]

Implementation: This is where you turn your designs to reality.

Maintenance: “What happens during the rest of the software's life: changes, correction, additions, moves to a different computing platform and more. This, the least glamorous and perhaps most important step of all, goes on seemingly forever.” [4]

Then you go back to planning the system again.

You can see the system design below:

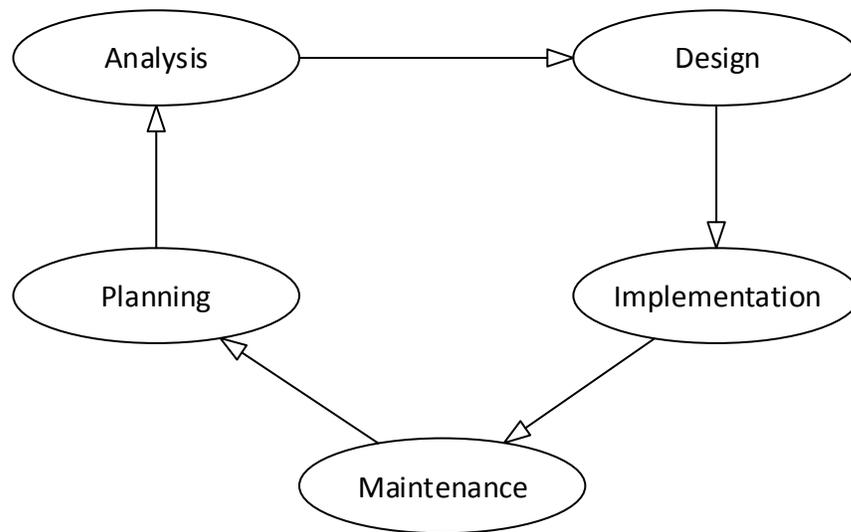


Figure 1: System development life cycle

As you can see from the scheme above, the system is in an everlasting production, on such systems you should make sure that modules can be easily added to the system, since the system is directly proportional to a company in real life, it can be easily effected by external factors such as; a new form of transportation was needed to be created, or any other part of the system is needed to be altered or modified.

2.3 Accessibility of an Information System

Accessibility by word definition is: “the quality of being at hand when needed” [5]. Accessibility is extremely important part of an information system. The degree of information system should be high and low both at the same time. Accessibility can be divided into two parts 1, who can access the

system? What is the accessibility of the system to an unknown user or IP? 2, where can you access the system from? Which devices can run it? Overall how easy it is to access the system?

For point 1, an information system should be limited to the users that are accepted by the system. The inner information should not be visible to everyone, such information could be important and secret thus should be kept as one. This means the information system should not be accessible by any outer user that tries to establish a connection with the system. However any user should be able to access the server and request access to the system, in other words the system should be accessible at a high degree if the user simply wants to request access to the system, and low accessibility if the user wants to access the data inside the system.

As for point 2, an information system should be extremely accessible in terms of what devices can access it, or how easy it is for a privileged user to get information. The information system should be cross platform. As for where can you access the system from the accessibility could be way less for extra protection, which is not recommended since customers could be world-wide or travel a lot.

Also according to Romano, N.C. accessibility also defines how easy a system is to access regardless of the health of the user, and how easy it is for him or her to use the system with any form disability that could happen [6].

To make the system as accessible as possible it is important that the system is compatible with a big variety of devices, which brings us to our next point.

2.4 Technologies Used

To make the system available to as many users and/or operating systems, devices and so on, we have to take into consideration that the technology used is supported by those devices and/or operating systems, we should take into consideration that the technology is powerful enough to fulfill our needs and expectations.

If we take into consideration that the system should have constant interaction with the server, we should decide on database engine, top engines used are as follows: Oracle, MySQL, and Microsoft SQL Server [7]. We can later decide if the application should be an application or a website.

2.4.1 MySQL vs. Oracle

Oracle is the highest scoring engine out of all the engines, it has good amount of features MySQL doesn't have and it is also the go to for huge amounts of transactions and less memory usage. However our choice will be MySQL over Oracle. There are multiple advantages MySQL has over Oracle, the biggest being MySQL is open source meaning not only the code is available

to the public but it is also free, where as oracle is not. MySQL is more portable it can be carried around way easier and it is also simpler to use.

2.4.2 Application vs. Website

This one is pretty easy to decide, an application is usually over-complicated to design such massive applications like this, and it is also way harder to distribute then a website. A website is a lot easier to interact with the database than an application, and is easier to export the website to a network. We should remember that an information system should be accessible, when we take that into consideration for guest accounts and/or travelling users a website is way more accessible than an application is.

A website can benefit from HTML5, W3schools definition for HTML5 is as follows: "Html 5 is specially designed to deliver rich content without the need for additional plugins. The current version delivers everything from animation to graphics, music to movies, and can also be used to build complicated web applications.

HTML5 is also cross-platform. It is designed to work whether you are using a PC, or a Tablet, a Smartphone, or a Smart TV." [8]

Which is an extremely important feature for an information system, using, Html5 you can make website compatible, fully functional and responsive for any device capable of running modern browsers.

CHAPTER 3

Software Analysis and Design

As stated in the earlier chapters the system will be accessed by a large number of users, and these users will have constant interaction with the server this brings some certain necessities such as performance, stability, and security. The system has certain requirements to be effective these requirements are divided into 2 categories

3.1 Functional Requirements

Functional requirements are must functionalities the system will provide, inputs to the functions are given and their respective output is shown [9].

3.1.1 Usability Requirements

The system is a website, access to the application is as simple as entering the IP address of the server, as soon as you enter the website you'll be shown a login form, this form is extremely basic and the style changes dynamically according to the device and width of the device so it isn't graphically ruined on different scales in other words, the website is responsive. As soon as the user is logged to not over bore the user with information the most basic view is shown, the profile. Navigation is divided into segments/parts and thus is

easier to navigate throughout the page. Every file has direct navigation to the parent of any other file in the system making navigation extremely simple.

The system has a single login form, for multiple user type's being student, teacher, administrator, working staff and parent, all these users log in from the same form however have access to different features. This makes the navigation through the page much simpler for the user.

3.1.2 Performance Requirements

Due to heavy load of server interactions the system was decided to be divided into segments every page has less amounts of code and as little server interactions as possible, default packages and libraries are used, so there are no outer dependencies for the server, which increases the performance significantly. The website takes up an extremely low amount of system memory for two reasons, one for being website which means it usually consumes low amount memory, as for second the application doesn't load a huge amount of images which is extremely good performance wise.

3.1.3 Supportability Requirements

The application is supported on almost every device, if accessed through the correct means the system will adapt to the device it is in. The application runs well on any device that supports any of the latest browsers.

3.1.4 Security Requirements

The user is able to login to the server in case he has a valid username and password registered on the server which is also confirmed by the system administrator, only then the user will be granted access to the system, depending on the privileges of the user, he or she will be granted access to certain features of the website, access will be blocked to certain feature incase the user doesn't have the access to it. In case the user doesn't have a valid username and a password an error message will be printed, and the user will be asked if he or she wants to regain access to his account, moreover the user can change the password, as he or she seems necessary. The connection should be established through a secure connection such as https:// to prevent session hijacking, form inputs should be cleaned from MySQL Injections.

3.2 Non-Functional Requirements

As it stands are functions that do not do anything on the system but rather help define the system to do the audience.

3.2.1 Hardware Requirements

A working server is required to run this project; the server should have the following features set up; PHP version 5 or higher, and a working MySQL server. This is necessary to store values to be it either Live Session environment; grades; user information and so on.

3.2.2 Software Requirements

Extremely basic stuff is required to develop the project to, a working local server is recommended for testing purposes as for the development environment your favorite text editor should do, and be it vim, notepad++ or phpStorm.

3.2.3 Security Requirements

The database is to have restricted access from the outer/unknown Internet protocol addresses, an important security measure is that the database is to have encrypted values; the encryption algorithm is to be defined by the provider.

3.3 User Profiles

The system consists of 3 primary actors, the parent, the student and the teacher. The feature list of the actors is explained as follows.

3.3.1 Student

The student is the primary actor of the system; almost every feature of the system revolves around the student. With the system the student is able to have more information about his/her current state in his education. The system offers almost any feature a normal information system would offer

like, access to his/her course grades, access student discipline information and see his/her profile, however the main feature of the Student information and activity system is the ability for a student to join a live session. A live session is an environment where the student and the teacher can interact. If joined a live session a student is able to take live quizzes, is able to access the current course material, request as being attended to the lesson, save the lecture assignment and also insert live comments about the lecture and or the teacher. After a live session is over the temporary information is deleted and is inserted as permanent information to relevant tables.

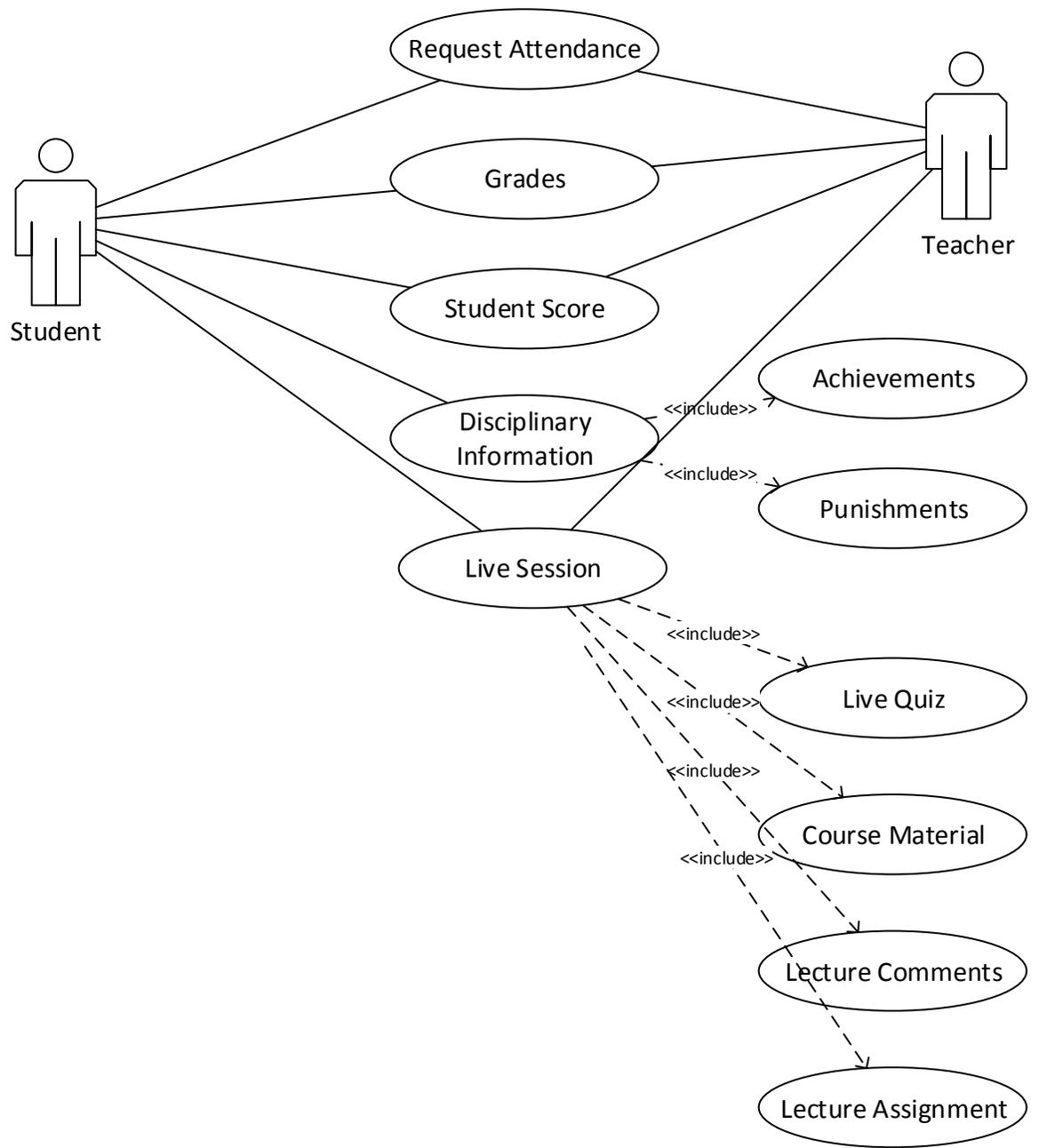


Figure 2: Use Case diagram of a student

3.3.2 Parent

The parent is doesn't benefit from the interactivity of the system, but more from the information side of the system. A parent can have multiple students registered to the school so he/she should be able to access all of his/her child's information. The parent also should be able to send any information the teacher should know about his/her child. If wanted the teacher can also pay school fees from the website.

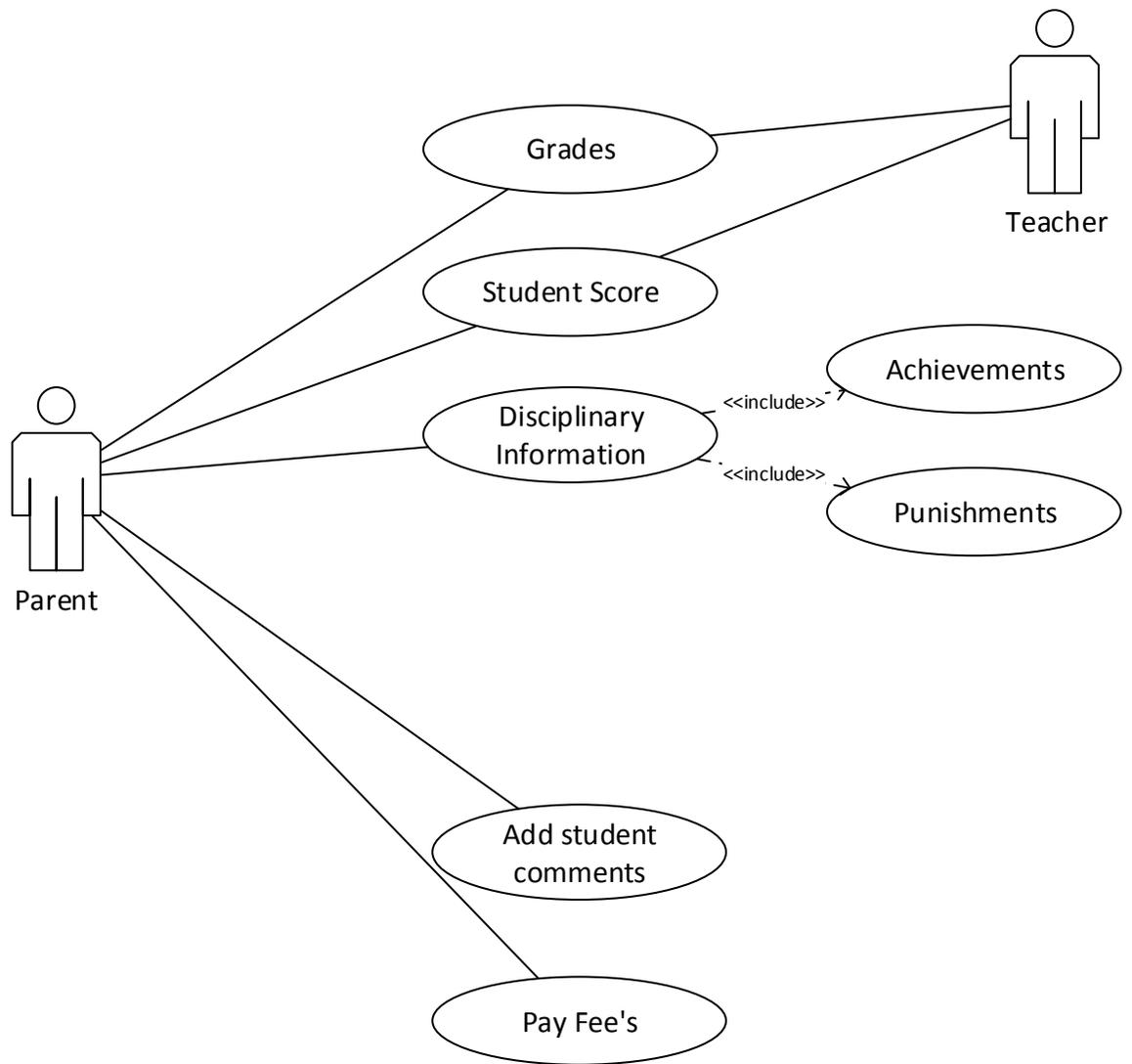


Figure 3: Use case diagram of a parent

3.3.3 Teacher

One of the main actors to the system is the teacher. He/she usually sets up the overall environment of the system and takes note of the student state. The teacher should provide the course grades to the student as regularly as possible; he/she should also send feedback to the student's parents. The teacher should also insert any misbehavior or any accomplishment points the student has earned causing the student score to change. The main difference of student information and activity system from other information system is the interactivity of the classes, thus the main feature also being the main duty of the teacher is to provide live session to the students. The teacher should create html formatted quizzes; he/she should also confirm attendances of the students. The teacher should also provide material for the specific lecture.

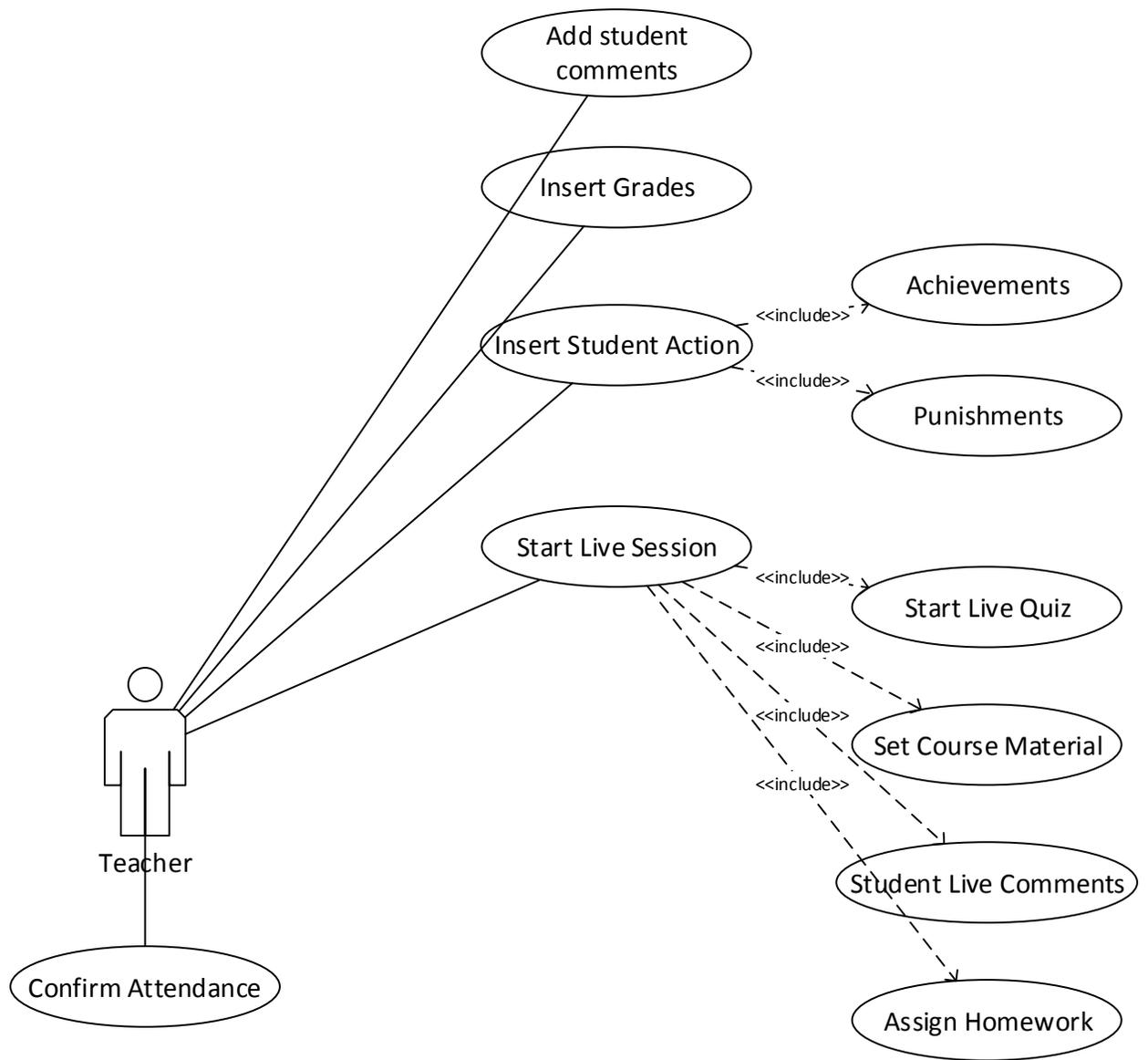


Figure 4: Use Case diagram of the teacher

3.4 System Design

This section explains more into detail on how the system works.

Being the primary actor of the system the student's activity sequence is as follows; the student inputs login form if the login is successful the student then proceeds to the main application, otherwise he/she is taken back to the login form. Once logged in the student is displayed a window where he/she sees the main application window. He can select the tabs live session, check his her grades and or check his disciplinary information. On the live session the student sees the available live session that he can join, there could be multiple live sessions for a single class considering the possibility of elective courses. Once joined on a live session the student can later see options enabled that the teacher has set, possible options that are enable are; do a live quiz, request as attended, download course material and insert live lecture comments. If the student click the request as attended button, a notification is sent to the teacher saying the student has participated to the lecture. If the student check the insert live lecture comments button, the student is displayed a form where he/she can type he can later submit the form and the form is inserted to the table for parents or teachers viewing. If do a live quiz option is selected the student is shown a quiz the teacher has prepared in html format. The student can also see his/her grades, which include attendance, exam grades, and assignment grades. This information is

displayed in a table view or each and every subject. The student can also see his/her disciplinary actions being view bad behavior or view achievements. This information is displayed together with reasoning of the specific action that caused it in a student comment form. These actions and scores define the student score.

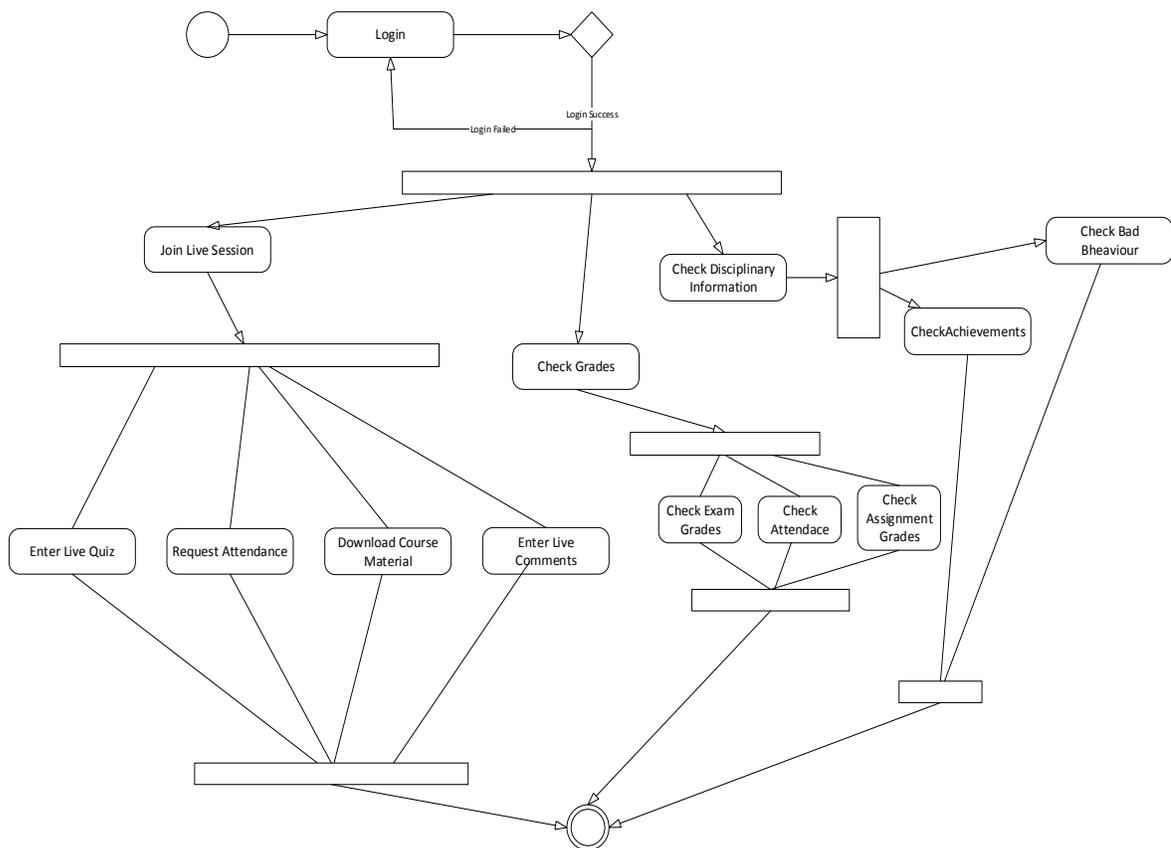
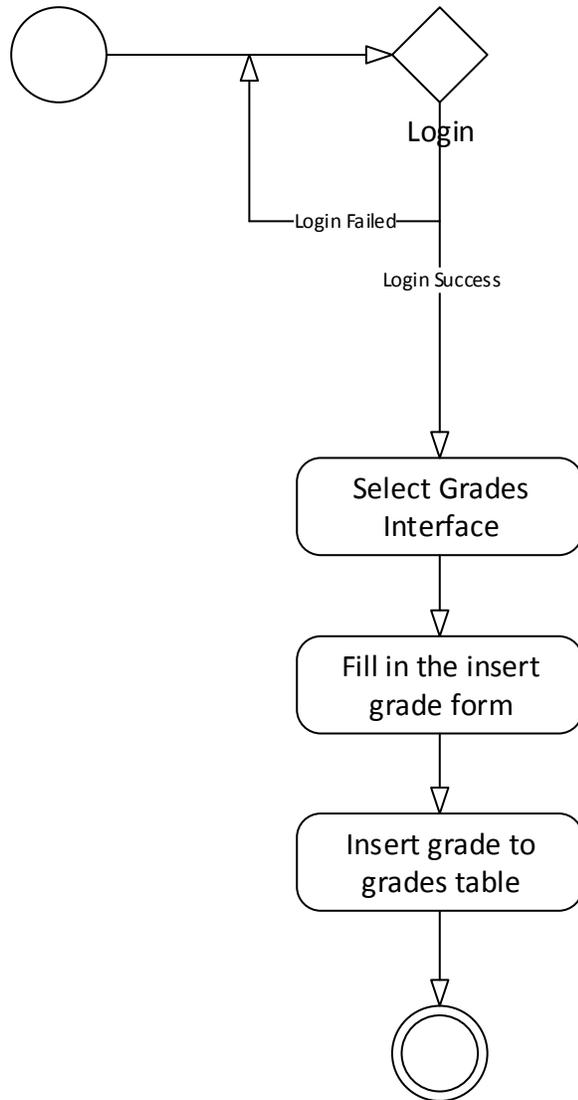


Figure 5: Activity diagram of a student

The system has an extremely simple to use, let's take the example of a teacher inserting a grade to for a student, first the teacher log's in, he/she is redirected to the login page if the login was unsuccessful. The teacher later

selects the grades table and fill in the insert grade form. This consists of grade type, grade name and grade values of each individual student.

Figure 6: Activity diagram of grade insertion



Below is the sequence diagram of insertion grade what this signifies in which order the function goes through the actors. The teacher first log's in and then

accesses the student information from the database and inserts a new grade there. The student information then displays the grade to the student

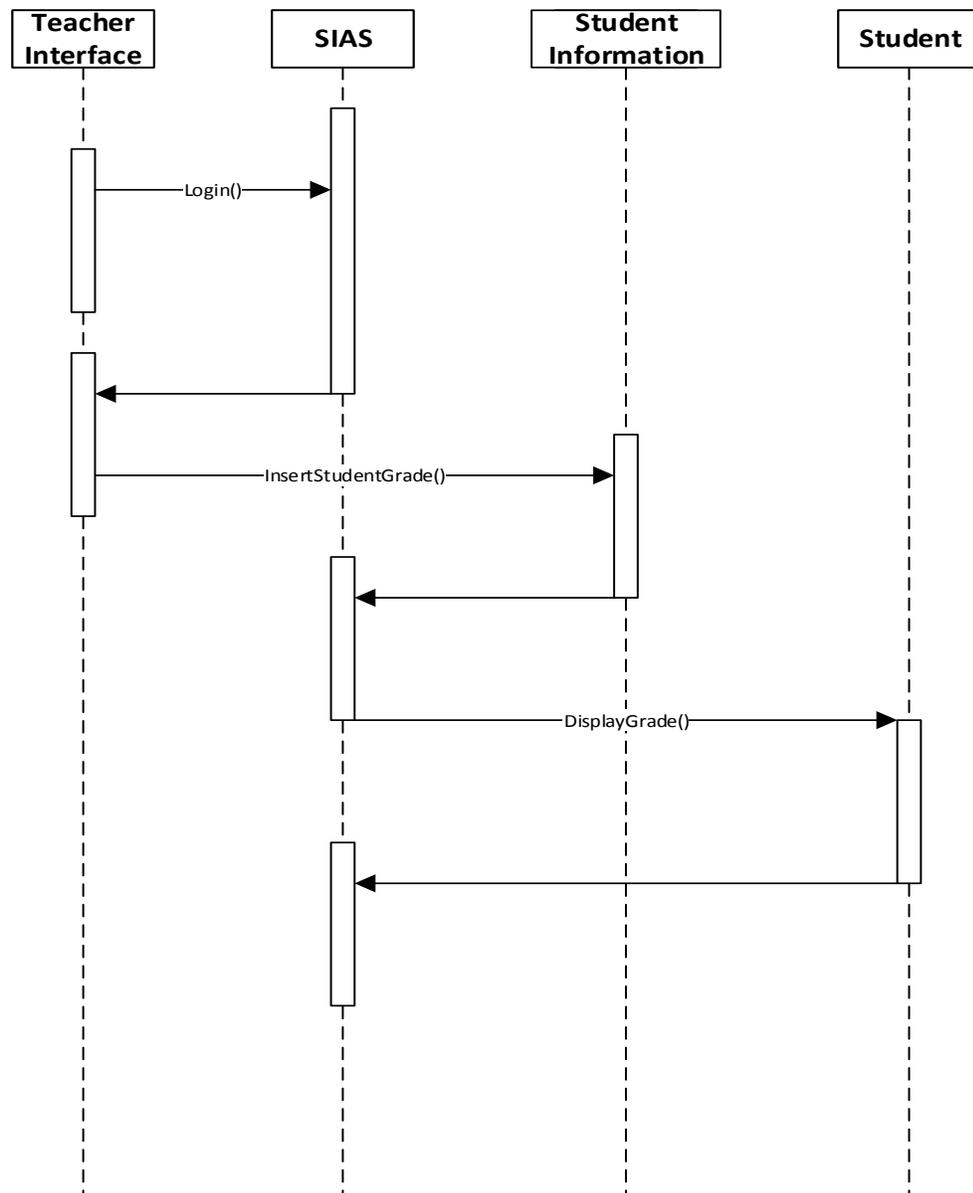


Figure 7: Sequence diagram of grade insertion

To define the system more clearly see the class diagram below. Since the application is a webpage we have divided each page to segments/files, first

you see a main page where you can access all the other classes, this main page simply hold functions that redirect to other files. This class as default shows the profile first, profile holds variables such as name, surname, photo id, birth date and location. The grades class displays grade information and functions that display these grades. The general features class is a class where you can do the general features if you're a student you can see your student grade or pay your fees if you're logged in, as a parent features as such are included there. The admin class has functions that let you to create users and give rights to certain users. The live session class which is the most important class in the system includes the main live session function which as explained earlier are create live quiz, get attendance, set course material and joining/creating sessions. Base connection class 'connect' is a module that connects to the database and returns the relevant database.

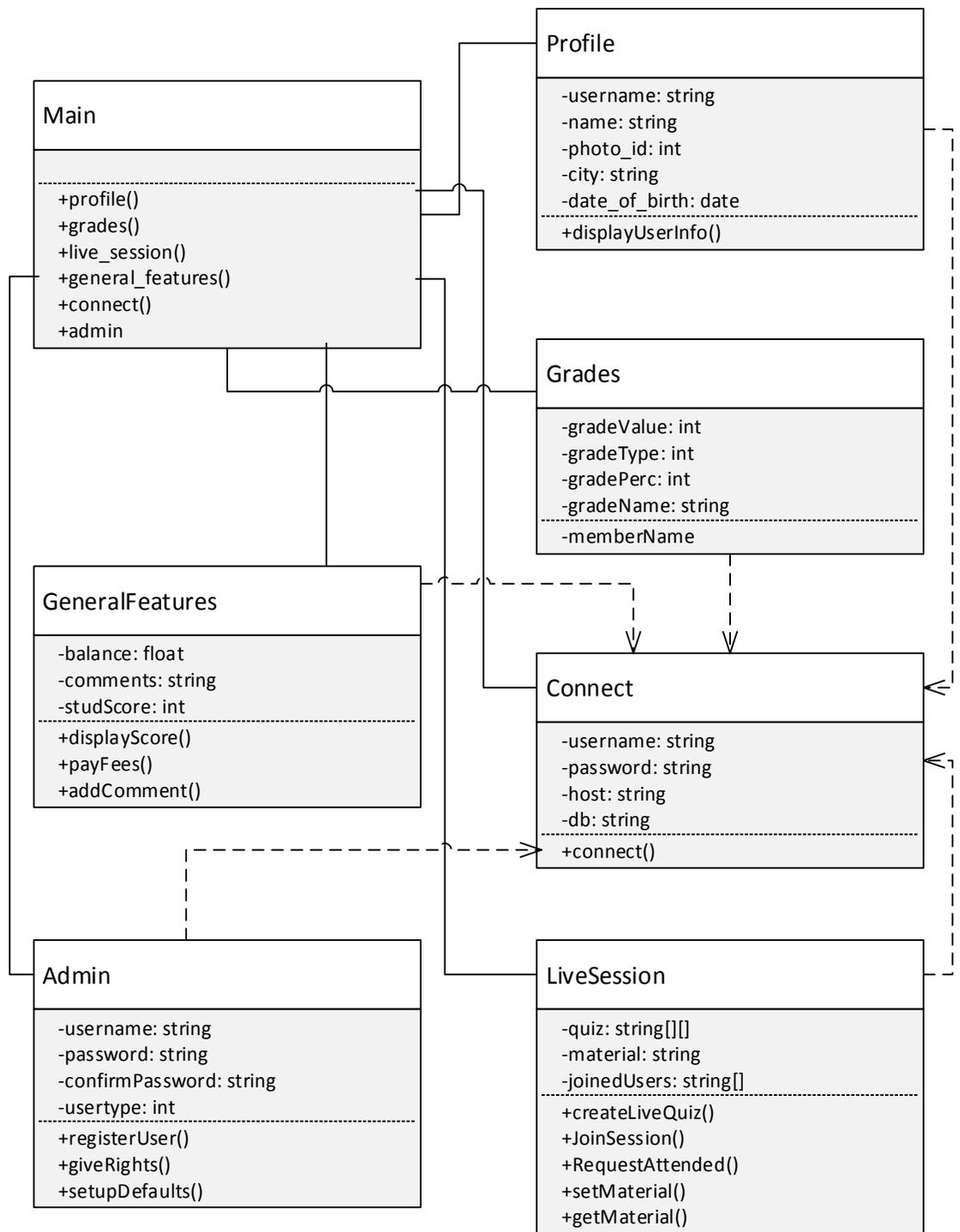


Figure 8: Class diagram of the system

If you're building a high level system such as student activity and information system it is important that you make a component diagram. The diagram below illustrated the components of the system

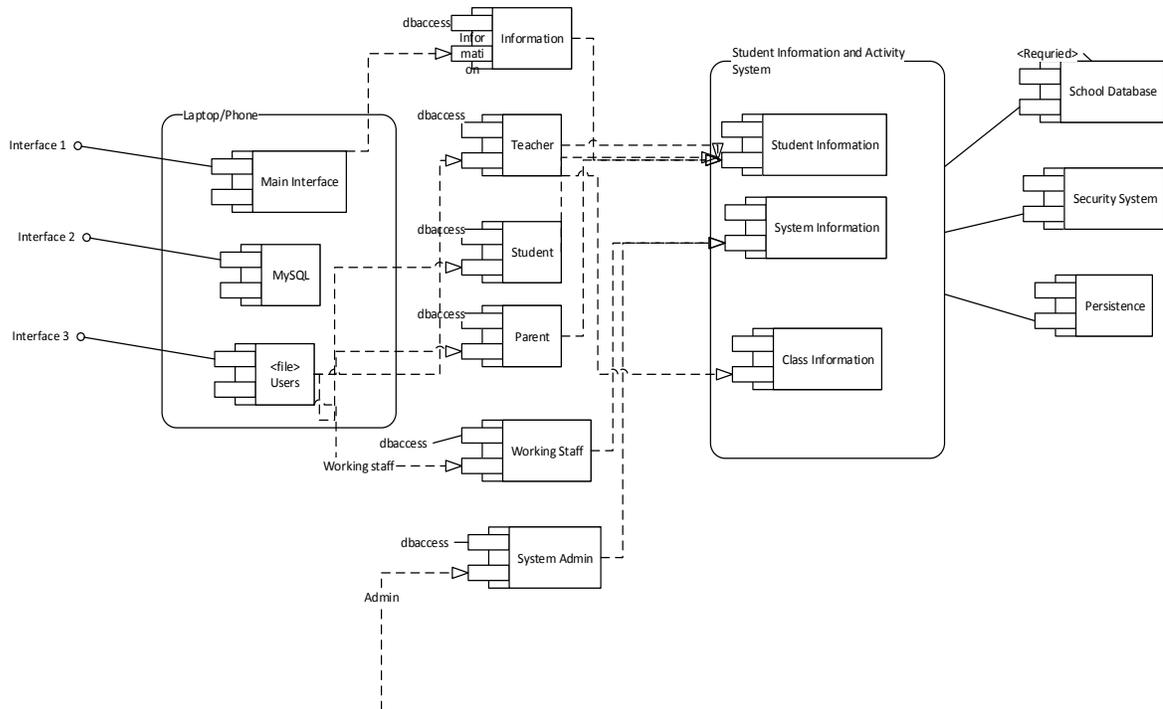


Figure 9: System component diagram

As seen from the diagram above it is clear that the system works in the following. The user connects to the web server via a computer or a phone, PHP accesses the server and parser the server side information. This is displayed more clearly with the deployment diagram below.

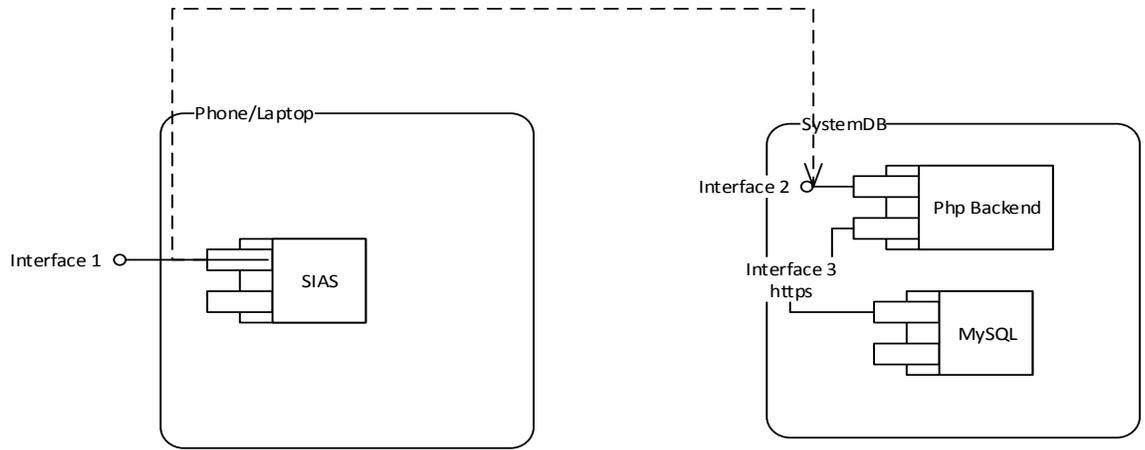


Figure 10: Deployment diagram of the system

CHAPTER 4

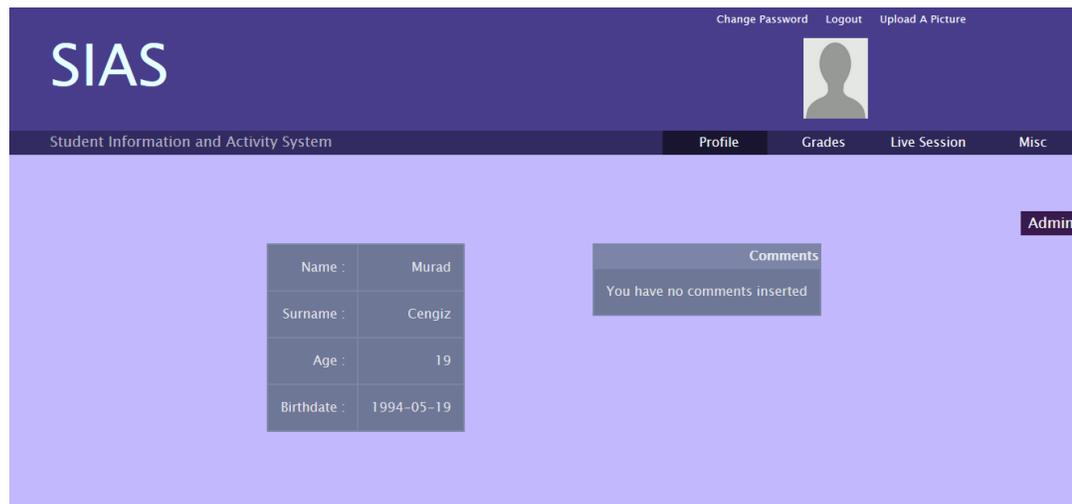
Implementation

One of the biggest requirements for the system was that should be responsive or in other words accessible, we want the project to be able to be accessed anywhere and at any time. Considering this we made the decisions we made.

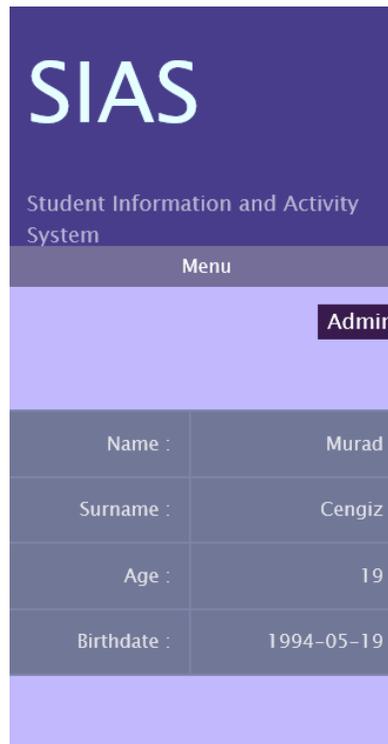
4.1 Technologies Used

4.1.1 CSS3

Since the website should be responsive and accessible we decided that we should implement a responsive design using basic CSS3 styles.



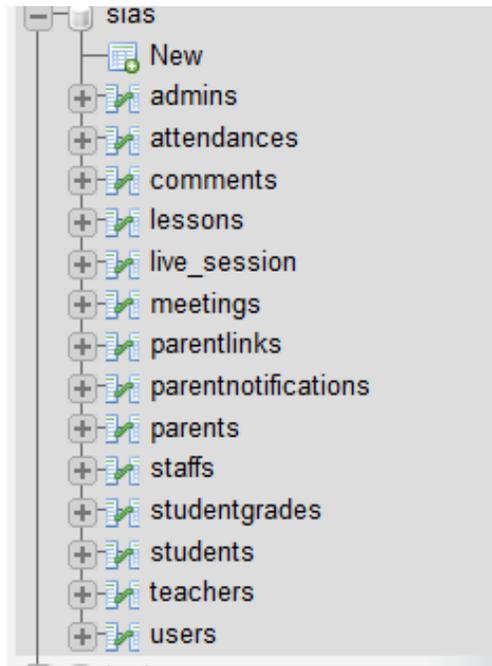
CSS3 allowed making the view different and adaptable depending on its size of the viewport and the device the user is using. There should be a different style for every size of the viewport and a different style depending if the device is in portrait or in landscape.



4.1.2 Database structuring

After the design of the system was done, the implementation for backend was necessary however as shown in the image below the database was created as follow so the coding part with the back end was easier. As seen

below the database was divided into multiple tables so it's easier to manage coding.



The database is ever changing in this system. New tables should be created and deleted every moment. For example when a live session is created a new table should be created and deleted when the live session is over.

After the live session is over the information gathered throughout that session becomes permanent in other tables such as grade information and student comments section. These permanent tables store the quiz, attendance, scores of the student, and obviously comments regarding the student and or the teacher respectively.

4.1.3 Server side scripting

Also known as backend scripting, the system uses the most common backend programming language PHP, which will take care of every server side query that will be made, it is responsible for fetching user data and display them in their respective locations.

PHP is also to define if a logged user is an admin, a student, a teacher, a working staff or a parent. Depending on which it is it will display certain forms and buttons and give access to certain files and deny access to some others.

```
$user = $_SESSION['user'];  
$query = mysql_query("Select * from users where username = '$user'") or die("Query error!");  
$row=mysql_fetch_array($query);  
$priv=$row['priv'];
```

In the screenshot above a variable '\$priv' is defined the variable is set to 0 if the user is an administrator, it is set to 1 if it's a teacher, 2 if it's a working staff, or 3,4 parent and student respectively. Depending on this information the user will have restricted access to certain features. Note that this information is fetched from the server.

4.1.4 JQuery

“Query is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript.” [10].

JQuery was used to enhance the design and make the web site more responsive. It enable to us to execute server side queries post and/or get methods without refreshing the page. It is important for a mobile to stay on the same page as much as possible since it doesn't have as much memory as the computer. It also makes the web site less of a nuisance. JQuery is capable of executing dynamic styles.

```
$('#prof-action-1').click(function() {  
    $('#blackout').show();  
    $('#change-password').show();  
});  
$('#prof-action-3').click(function() {  
    $('#blackout').show();  
    $('#change-profile-picture').show();  
});  
$('#pr-cancel').click(function() {  
    $('#blackout').hide();  
    $('#change-profile-picture').hide();  
});  
$('#ch-cancel').click(function() {  
    $('#change-password').hide();  
    $('#blackout').hide();  
});  
$('#ch-cancel2').click(function() {  
    $('#change-password2').hide();  
    $('#blackout2').hide();  
});
```

As told before JQuery makes the web page more interactive which is important with live session and live quizzes.

```
jQuery(document).ready(function() {
    $('#blackout').hide();
    $('#change-password').hide();
    $('#change-profile-picture').hide();
    $('#select-teacher').hide();
    $('#select-parent').hide();
    $('#select-staff').hide();
    $('#notifications-display').hide();
    $('#tf-notify').click(function() {
        $('#blackout').show();
        $('#notifications-display').show();
    });
    $('#close').click(function() {
        $('#blackout').hide();
        $('#notifications-display').hide();
    });
    $('#select').change(function() {
        var selectvalue = this.value;
        if(selectvalue == "Parent"){
            $('#select-student').hide();
            $('#select-teacher').hide();
            $('#select-parent').show();
            $('#select-staff').hide();
        }
    });
});
```

The image above describes of how JQuery helped us enhance the interactivity of the web page. As a live session is created the website isn't refreshed but it simply replaces the html in the document.

CHAPTER 5

Conclusions and future work

Information systems are the backbone of big organizations, schools where one of the organizations that would benefit highly from such system. Schools is the place where individuals get their base characteristics from, you should keep close eye on the students, and make sure they're not getting bad but they're getting good examples from the most influential environment of a person's life. A school should get constant feedback from the students and make student be interested to the lesson and make them give their feedback. Live session solves a little of this problem however the system should be more interactive and more informational, one feature that Is planned to be added is an informative gaming environment since children already play games at least we can make those games informative and keep track on what games they're playing. The second feature we're planning on adding is a more information about the student like health tests, skill tests surveys for more feedback from the students.

APPENDIX A

DIAGRAMS

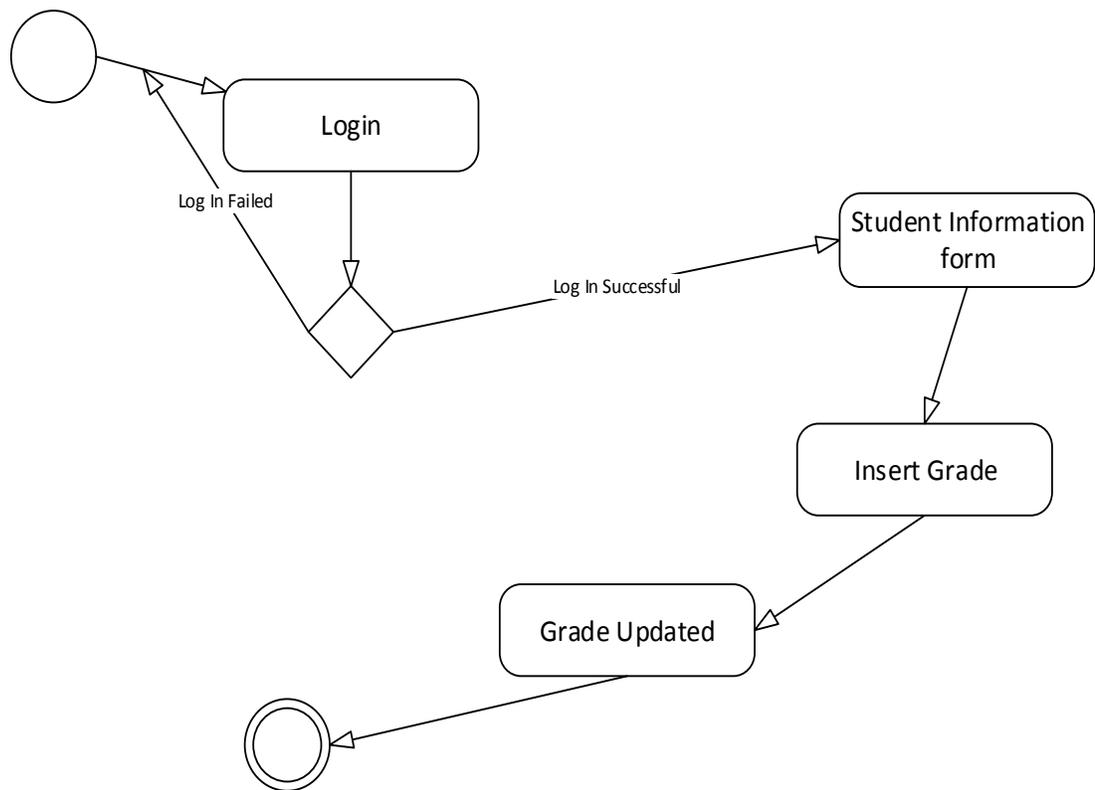


Figure : 11 State Diagram Insert Grade

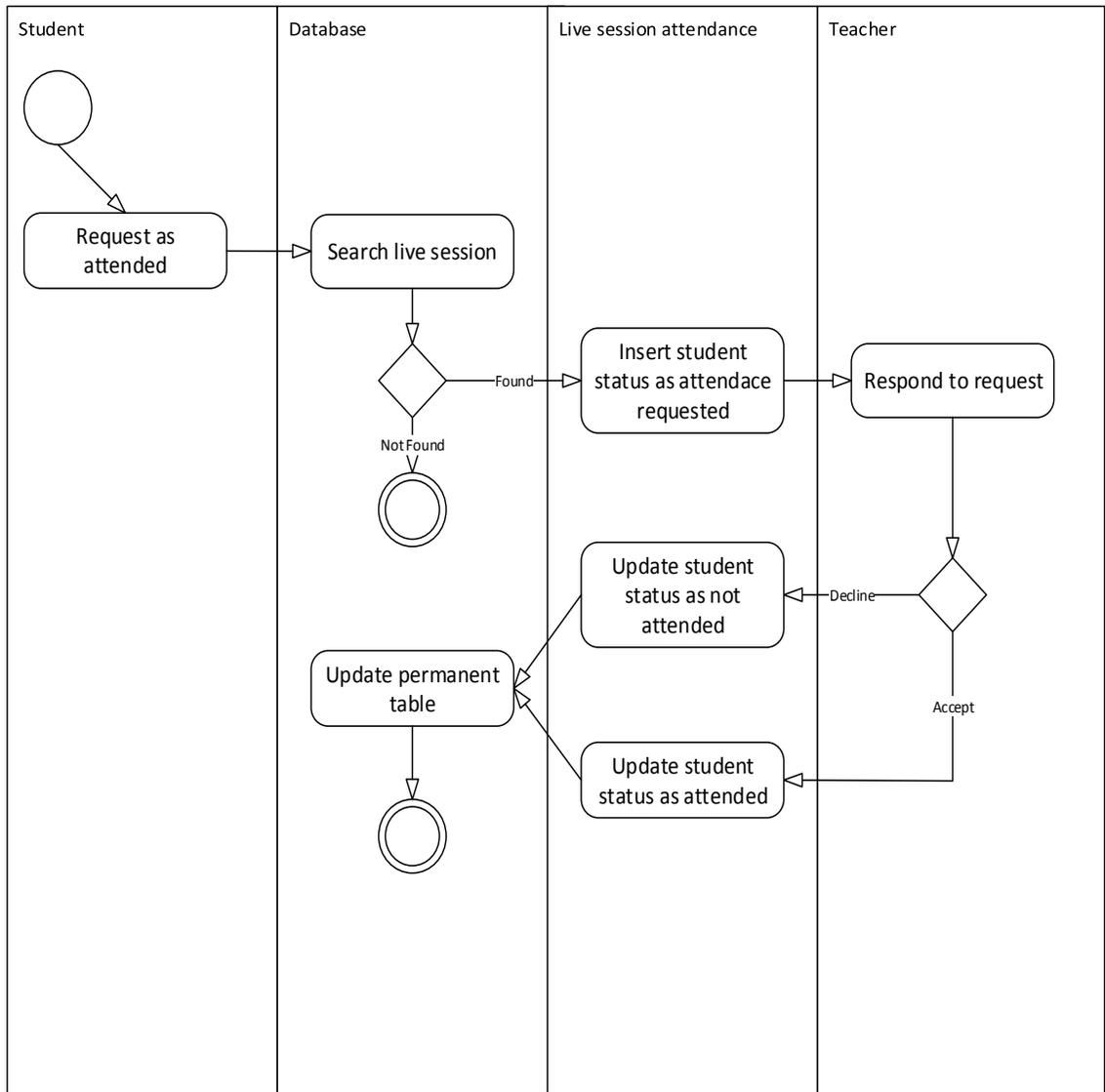


Figure 12: Swim lane Request attendance

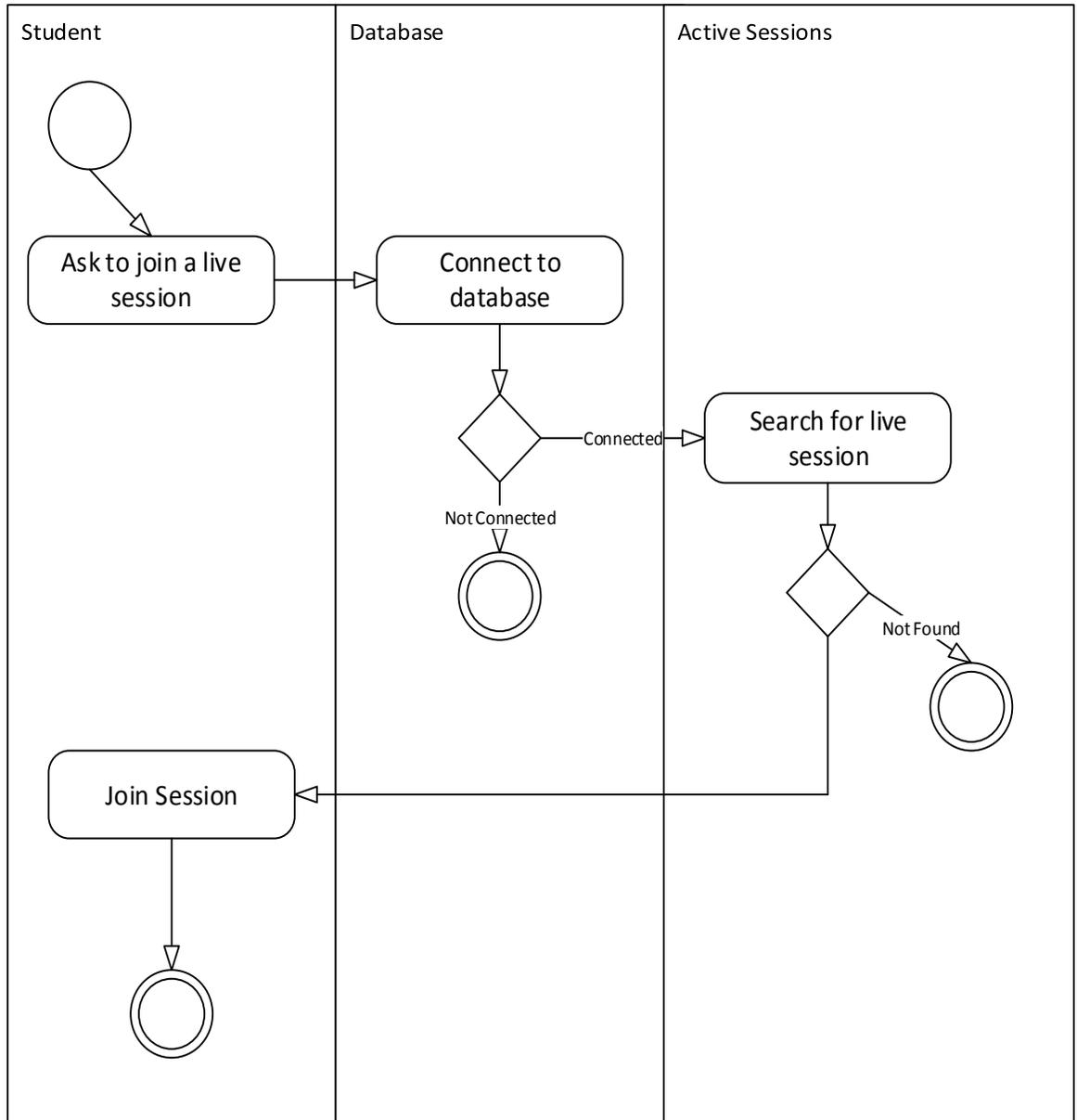


Figure 13: Swim Lane Join live session

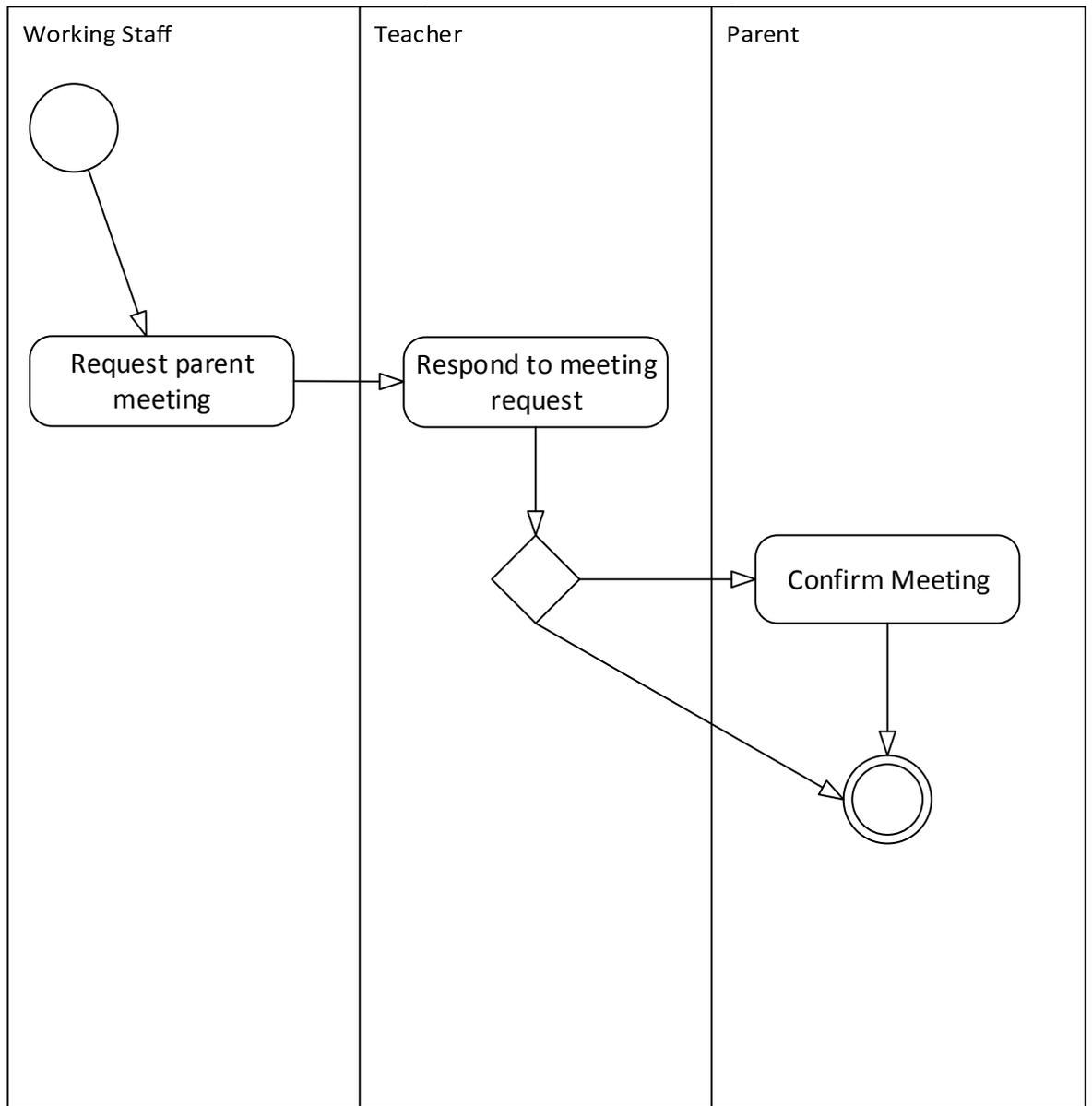


Figure 14: Swim Lane Arrange parent meeting

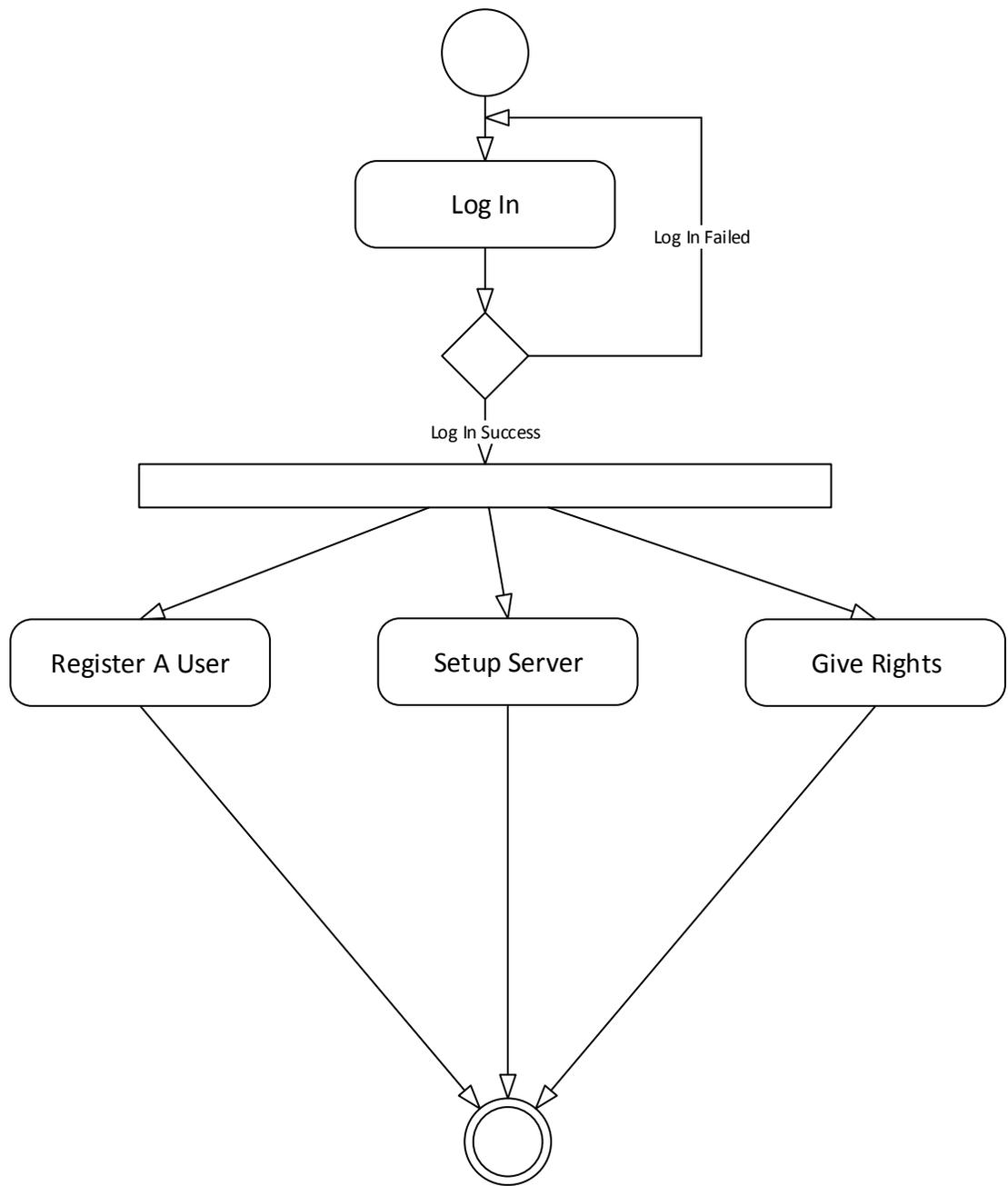


Figure 15: Activity Diagram System administrator

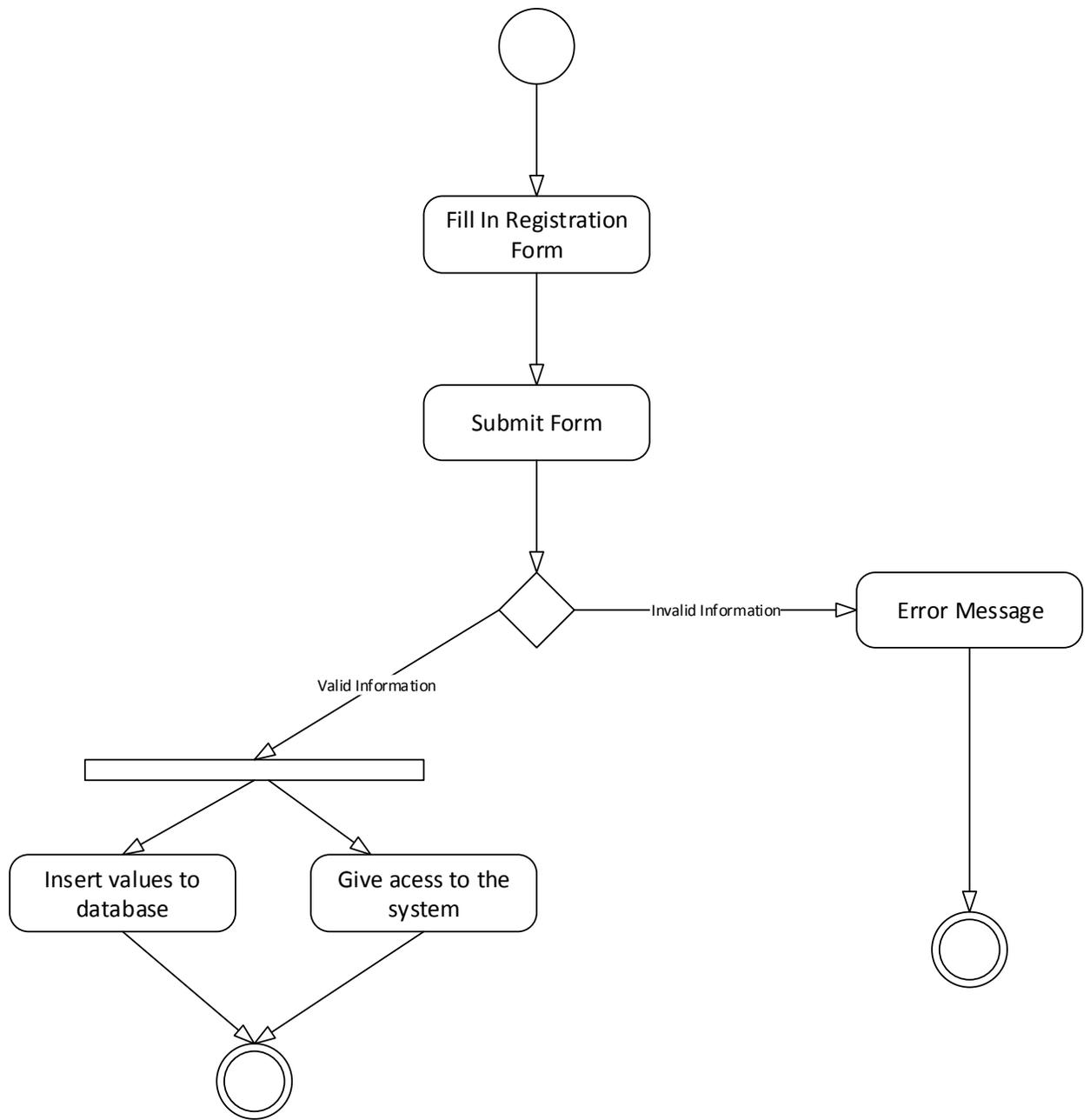


Figure 16: Activity Diagram Registration

REFERENCES

- [1]: WordNet Search. Retrieved April 11, 2014, from <http://wordnetweb.princeton.edu/perl/webwn?s=information%20system>
- [2]: Zwass Vladimir, information system. Retrieved from <http://www.britannica.com/EBchecked/topic/287895/information-system>
- [3]: Rouse Margaret, 2009, May 19. Retrieved from <http://searchsoftwarequality.techtarget.com/definition/systems-development-life-cycle>
- [4]: Kay Russell, 2002, May 14. Retrieved from http://www.computerworld.com/s/article/71151/System_Development_Life_Cycle
- [5]: WordNet Search. Retrieved May 14, 2014, from <http://wordnetweb.princeton.edu/perl/webwn?s=accessibility>
- [6]: Romano, N.C.; Loiacono-Mello, E.T. ; Mccoy, S. Information Systems Accessibility. 2003. Retrieved from <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1385604&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F9518%2F30166%2F01385604>
- [7]: DB-Engines Ranking. June 2014. Retrieved from <http://db-engines.com/en/ranking>
- [8]: HTML5 Introduction. Retrieved June 11, 2014, from http://www.w3schools.com/html/html5_intro.asp
- [9]: Briland Hitaj, Applying Mobile Technology for a Pervasive University Information System pg. 17
- [10]: Retrieved June 15, 2014, from <http://jquery.com/>