SMART NOTICE

A thesis/project submitted in partial fulfillment of the requirements of Varendra University for the degree of BSc Engineering in CSE.

January 2018

Submitted by:

Name: Maksuda Mimi

Id: 141311030

Department of Computer Science and Engineering

Varendra University

Rajshahi, Bangladesh

Supervised by:

Nakib Aman Turzo

Lecturer

Department of Computer Science and Engineering

Varendra University

Rajshahi, Bangladesh

Abstract

This document is a proposal for the “Smart Notice” as the minor project for the partial fulfillment of the course of Bachelor in computer Engineering.The system will primarily be web based as to facilitate easy access to all clients’ regulars of their location or the type of machine.Apart from this, it will also have supplementary applications program for Today’s leading operation performs like android and windows .These application be very user-friendly and will have features like auto-update and push messaging for user convenience. This project will teach us the practical implementation of procedural programming .Moreover it will also provide us first-hand experience about several important fields of a computer engineer, like web development and application development. It will also help us learn web based scripting like PHP, Java Script and CSS and RDBMS like MYSQL .We strongly believe going through this project will make us more competent for similar future tasks and enhance our technical and other abilities.

Acknowledgements

First and foremost, we would like to thank Nakib Aman Turzo for instructing us about the methods of conducting feasibility and requirement studies as well as system designing. Furthermore, we would like to express our deepest gratitude MD.Khademul Islam Molla Professor, Department of Computer Science and Engineering, for this never ending support, moral or otherwise.Finally, we would like to collectively thank any and all people that have helped us in any manner with respect to this proposal.

Contents

Figure /Table of Contents

Chapter 3: Validation

Chapter 1

Introduction

1.1 General Introduction

A Smart Notice is a place where people can leave public messages, for Example, to advertise things, announce events, or provide information.

Online Notice board is a web application which is engaged in providing up-to-date articles & notices and other information's for all the users or student associated with the particular campus or department. The paper aims at, how the online notice board can improve the efficiency of the student when it comes to gaining the information from the college. Online notice board is one of the applications to improve the usage of notice board of the college by making it available online. This web application helps the students to retrieve all the notices and articles directly through their cell phones, laptops and computers.

1.2 Applications of smart notice

1. **To eliminate wastage of time and energy:**

   Smart Notice app will be able to save lot of paper and time. It directs both teacher and pupil's energy and attention to one thing at a time by placing proper persons at their proper places at the proper time. Everything will be instantaneous.

2. **To avoid duplication and overlapping:**

   This application will help to remove the duplicity of notices. Only one person, who is admin, can post the notice. No one else would be able to do so student and staff will be given correct information all the time.

3. **To ensure due attention of student to each and every notice:**

   Smart notice App ensures that everyone has kind attention to every notice and updates going on in college. There will be a buzz at each and every notice to drive the attention of student to check it once. In this way, students will be well informed about their college activities.

4. **To bring system into college life:**

It would be dire need of all colleges as it's easy and shortcut method to inform all the students. In the absence of proper notification system will make it very difficult to inform students at right time.

5. **Searching a particular Notice:**

This application allows you search the notice very easily through title of notice. If anyone forgots about the notice details, he can search it out very easily.

6. **Free Service:**

It gives free service to notify all the students. There will be no cost of sending notification to all. Just have the good system implemented in college and that too free of cost.

7. **Prevent Crowd in College:**

As you can see, there is always a crowd at notice board. As notice board is one, and people to see notice are more. With this application there will be no more Crowd. Everyone will be well informed even at their homes. So they are free to do there other work.

8. **Automatically Updated Dashboard:**

The dashboard of notice is automatically updated when a new message arrives. The user can himself refresh the dashboard to see any new notice.

9. **Anytime Anywhere Services:**

With this application, notices will be delivered anytime and at any place. There is no restriction of time to send a notice.

10. Keeping Notices at one place:

This application allows you to have notices in one place only. If there is an attachment with that, all will be placed in a separate folder dedicated to that application. So there will be no here and there of notices.

11. Battery Saving Application:

This application saves your battery too. It’s because, the service implemented in application is not running all the time. Whenever GCM ping the mobile, only then it makes a broadcast to phone that initiates the service. In this way, it’s saving your battery a lot.

12. Anytime Anywhere Service:

With this application, notices will be delivered anytime and at any place. There is no restriction of time to send a notice.

13. Keeping Notices at one place:

This application allows you to have notices in one place only. If there is an attachment with that, all will be placed in a separate folder dedicated to that application. So there will be no here and there of notices.

14. Free Service:

It gives free service to notify all the students. There will be no cost of sending notification to all. Just have the good system implemented in college and that too free of cost.

1.3 Identification of Need:

2. As we discussed earlier that manual maintenance of a notices is a tedious Job. So to enhance the ease of working, we go for this package.
3. Giving the facility to convey messages to all students anytime and anywhere
4. Making students updated about all the events and activities going on in the college.
5. The student will not require standing in the crowd to see the notice. There will be no issue of fighting in order to see the notice first. Everyone is first to see that notice inside their own mobile phone anywhere and anytime.
6. The least but most important it saves time.
7. Utilizing less man power. As there are many persons involved in circulating the message. With this application, only one person is required to post the notice. Rest of the man power is saved in the entire process.

1.4 Objectives

The proposed system's objectives are to overcome all the limitations and drawbacks of the existing system.

The online smart notice application is user-friendly android application. The main objective of the application is its simplicity of design and ease of implementation that shows and helps to collect most of the information about events going on in college premises. The interface will be very user-friendly.

The main objectives of the proposed system can be enumerated as follows:

1. Faster dissemination of notices regarding education, technical events, cultural events.
2. Any lost/found going out in college.
3. Easy way to broadcast your message.
4. Helps you to be updated with what's going on in College.
5. Good way to advertise about Tuitions/ Coaching and Courses.
6. User can also follow a group notice board.

1.5 Disadvantages

1. It is a wasting a time and paper also.
2. To see notice student or user can require register themselves; otherwise they will be not able to see notice.

1.6 Advantages

1. It is not required to going at College Notice Board see Information regarding to any event or advertise or a messages.
2. The College easily posts all types of message on the notice board.
3. Students can easily go through the notices through their logins.
4. The management authorities can automatically set to activate/deactivate a given notice for a given time period thus helping in atomizing the posting and removal of notices after the given time period.
5. The management authorities can also manually activate/deactivate the notices incase the college/department wants to remove the notice bearing wrong message.

1.7 Primary Aim

The primary aim of the Online Notice board Software project is to create a fully functional digital notice board system which will efficiently handle all assigned tasks. It may be able to, in due courses, heavily minimize, if not eradicate, the conventional, Physical notice boards. With the help of the supplementary applications, users will be able to receive real-time notifications of any and all notices posted by another user with privileges as such.

1.8 Unique Features of System

The unique features of this application are as follow:

1. **GCM Notification:**

   Google Cloud Messaging has been used to broadcast notices to all the students who are registered with this application. It will help you receive the notice even if you application is closed. So it implements anywhere anytime notifications. Google Cloud Messaging (GCM) is a free service for sending messages to Android devices. GCM messaging can greatly enhance the user experience. Your application can stay up to date without wasting battery power on waking up the radio and polling the server when there are no updates.

2. **Battery Saving Application:**

   This application saves your battery too. It's because, the service implemented in application is not running all the time. Whenever GCM ping the mobile, only then it makes a broadcast to phone that initiates the service. In this way, it's saving your battery a lot.

3. **Automatically Updated Dashboard:**

   The dashboard of notice is automatically updated when a new message arrives. The user can himself refresh the dashboard to see any new notice.

4. **Anytime Anywhere Service:**

   With this application, notices will be delivered anytime and at any place. There is no restriction of time to send a notice.

5. Keeping Notices at one place:

This application allows you to have notices in one place only. If there is an attachment with that, all will be placed in a separate folder dedicated to that application. So there will be no here and there of notices.

6. Free Service:

It gives free service to notify all the students. There will be no cost of sending notification to all. Just have the good system implemented in college and that too free of cost.

7. Case Pictures in One Folder:

The attachment of notices goes at one place only. If you want to see all attachments, you can go in gallery and can see it. It’s a good feature of this application that cares for our files organization in phone too.

Chapter 2

Software Requirement Specification Document

2.1 Data Requirements

Data requirement is meant to be the data that will be used in our application. Data required in this project is all notices that need to be conveyed to the user. This application also requires the username and passwords of persons in order to register them and sending notification about updates. So two main requirements are:

1. Notice Details
2. User Details

2.2 Functional Requirements

In order to make this application functional, we require the following:

1. **Download mobile application:**

   A user should be able to download the mobile an application through either an application store or similar service on the mobile phone. The application should be free to download.

2. **User registration:**

   Given that a user has downloaded the mobile application, then the user should be able to register through the mobile application. The user must provide User-name, password and e-mail address. The user can choose to provide a regularly used phone number.

3. **User Login:**

   Given that a user has registered, then the user should be able to log in to the mobile application. The log-in information will be stored on the phone and in the Future the user should be logged in automatically.

4. **Reset Password:**

Given that a user has registered, then the user should be able to retrieve his/her password by e-mail.

5. **Dashboard:**

Given that a user is logged in to the mobile application, then the first page that is shown should be the dashboard page. The user should be able to see all the college notices.

6. **Search Notice:**

The user should be able to search for a notice by its title. For example, if a user types fee, all the notices having fee in their content get displayed.

7. **Selecting a Notice:**

A user should be able to select any notice from list view. The click on particular notice will take him to notice details of that particular notice.

8. **Navigating back to Notices List:**

The user should be able to navigate back to notices list from the notice details section. This is required to give a good user experience.

9. **Deleting Notices:**

The user should have the option to delete the unnecessary notices from his phone; by ticking them one by one and then deleting those in one go. This way, user can save this phone memory from unrequited notices.

10. Posting Notices:

The admin of this application should be able to post the notices. He should be able to add a picture within notices. That picture can be taken either from gallery or by using the camera of the mobile Phone.

11. Notification Alert:

All the registered users should be able to have a ping or notification on their mobile phone whenever a new notice is posted.

2.3 Performance Requirements

The requirements in this section provide a detailed specification of the user interaction with the software and measurements placed on the system performance.

1. **Prominent search feature:**

The search feature should be prominent and easy to find for the user.

2. **Usage of the Notice Information:**

The notice link should be prominent and it should be evident that it is a usable link. Selecting the notice link should only take one click.

3. Response Time:

The response time should not be more than 5 seconds if users have a proper internet connection.

4. Fault Tolerance:

The fault tolerance of the system should be very good. If the system loses the connection to the Internet or the system gets some strange input, the user should be informed.

2.4 System Dependability

Following are the requirements that an application require from the device/mobile on which it is installed.

1. **Internet Permission:**

Application developed, require full internet permissions of mobile so that it can fetch notices from at the same time, it should be able to receive buzz or notification tone the server.Whenever new notice is posted by admin.

2. External SD Card Writable Permissions:

This application would be requiring read write access to SD card. It is required in order to download the notices attachment and save in SD card of mobile phone.

2. **System Tools:**

This application requires various system tools to be used. For example, it requires Camera of mobile in order to click the image and post in into notice. It also requires a system tool that prevents it from sleeping.

3. **Hardware Control:**

It uses vibrator of mobile phone whenever any notification arrives.

4. **Account Info:**

It also fetches your Google account information in order to get the user registered with Google Cloud Messaging.

2.5 Security Requirements

1. **Communication Security:**

There should be security of the communication between the system and server. The messages should be encrypted for log-in communications, so others cannot get user-name and password from those messages. Every exchanged of Information between client and server should be encrypted so that no one can track it.

2. **Admin Login Account Security:**

If an admin tries to log in to the web portal with a non-existing account then the admin should not be logged in. The admin should be notified about log-in failure.

3. **Admin Account Security:**

There should be security of admin accounts. An admin and IP address should not be able to log-in to the web portal for a certain time period after three times of failed log-in attempts.

4. **User Create Account Security:**

The security of creating account for users of the system should be maintained. If a user wants to create an account and the desired user name is occupied, the user should be asked to choose a different user name.

2.6 Requirement Analysis

Major Functional Requirements:

1. The system should meet the following functional requirements:
2. The system should be able to manage and store notices and files.
3. The system should provide appropriate UI for notice and file upload and Download.
4. The system should be able to recognize and authenticate several levels of Users.
5. The system should be able to manage and maintain a proper database.
6. The system should be easily operable and user friendly.
7. The apps should be able to work seamlessly with the web server.
8. The apps should be able to provide push messaging in real time, whenever Available.

2.7 Push Messaging

1. Users and Priority:

The system shall notify the primary users with recent notices and events with automated message in their mobile phones or computers.

2. Stimulus/Response Sequences:

A change in the value in the database, caused by a user uploading a notice from his client side, will trigger an event which in turns triggers the push messaging event across all users who are set to receive push message for that particular notice and/or uploading user in question.

3. Functional Requirements:

- 1: Access to the Internet.
- 2: Push Messaging Supporting Device.
- 3: Proper user authentication.

2.8 Database and UI Auto

1. Update Description and Priority:

The system shall auto update its database. Recently posted notices and Events as well as prioritized files shall be focused while outdated notices and files shall be cleaned up to make memory space available for new posts.

2. Stimulus/Response Sequences:

Any notice uploaded by any of the users triggers an event to append it To the database and update the UI accordingly across all the clients with Users of the same class.

2.9 External Interface Requirements

1. User Interfaces:

The system will start with a login page to identify the user level and the features will be available accordingly. The application will have easily accessible buttons for all major activities of the software. Key features and recent events will be highlighted to attract user attention.

2. Software Interfaces:

It will have the most interaction with an external database server. It may also be integrated with other systems like Employee Management System and/or Student Management System.

3. Communications Interfaces:

An active internet connection will be required for the functioning of the software. In case of the mobile apps, an internet connection will then allow the software to connect to the online database. The system will use the Hyper Text Transfer Protocol (HTTP) to transmit data.

2.10 Other Nonfunctional Requirements

1. Performance Requirements:

The system in itself does not require anything specific for basic operation, but the complete software with all its components running may have some performance requirements. Except viewing pre-loaded notices and files (in mobile apps), an internet connection is needed for the features of the system to become available.

2. Safety Requirements:

Major attention should be given to the safety and security of the data and information
That is stored in the software. The database must be trustworthy and non-leakage to ensure no data loss occurs.

3. Security Requirements:

User authentication must be absolute and non-by-passable. No user should be able to access the software without providing proper authentication. In case of guest users, only Public, notices and events should be visible. Also, the IP address of the client machine can be recorded for future follow up of any security issues that may arise.

4. Software Quality Attributes:

Several additional qualities and characteristics of the system will be important to the Client and/or the developers like correctness, maintainability, portability, and testability And usability. For correctness, proper care and attention should be given during the design and coding from both developers and customer (should correct some false And unwanted features) side. Usability is achieved by developing the product as User friendly as possible. Similarly, maintainability and testability plays vital role In the long life of the software.

Chapter 3

Validations

3.1 validations of smart notice:

Any application is useless without validation. There should be a way to validate the user input first before sending the user request to the server. Following are the validations implemented in proposed system:

1. **User Password Validation:**

   The application should check the user and password fields before sending any request to the server. It should check whether the fields are filled or not. If fields are not filled up, user should be instructed to fill up the fields before moving Further. In this way, there will be less traffic on the server.

2. **Validations During Registration:**

   There are a lot of validations that needs to be implemented in the application. They are as follow:

1. **First and Last Name of User:**

   The first and last name of user should be not null. Also first letter of first and last name should be in uppercase.

2. **Username:**

   The username can contain only alphabets, digits, underscore and hyphen. It should be at least 3 characters long and maximum of 15 characters.

3. Password:

The password must contains one digit from 0-9, one lowercase character, one uppercase character, one special symbols in the list"#$%" and length of password must be at least 6 characters and maximum of 20.

4. Email:

The application must validate and email address entered by the user before sending request to the server.

5. Mobile Number:

The mobile number should be of only twelve digits. No more, no less than that.

6. Validating During Posting Notices:

The application should validate the notice posting fields before posting any notice. It should check whether title and description fields are filled or not. If not, it should tell the user to fill up the required fields while posting the notice.

7. Reset Password Validation:

The application should check that user has entered the username or email in the given filed before pressing the reset password button.

3.2 Detail Design

The detailed design of this application is as follow:

1. **Registering a User:**

   The first step in this application is to get the users registered to both GCM Server and to Remote Web Server. For this, user will provide all the necessary details and press the register button. The request will first go to Google Cloud Messaging Server. GCM Server will provide the registration id for that device. After that, all the information along with registration id is stored on Web Server and the user gets registered.

2. **User Login:**

   After registering, the user is allowed to log in. Username and password after validating at client side, is sent to server side to authentication. After authentication response is sent by the server to client, and then user gets logged in.

3. **Viewing the Notice:**

   At the first time, when you are using this application for the first time, it will fetch all the notices from server. In all the other case, all previous notices are fetched from application's own database stored inside client mobile. It then checks for new notices from the server. If there are new notices fetch all those notices.

4. **Searching a Notice:**

   The user is able to search the notice in list view depending on the title of the notice. It helps user to get the desired notice instantly.

5. Deleting a Notice:

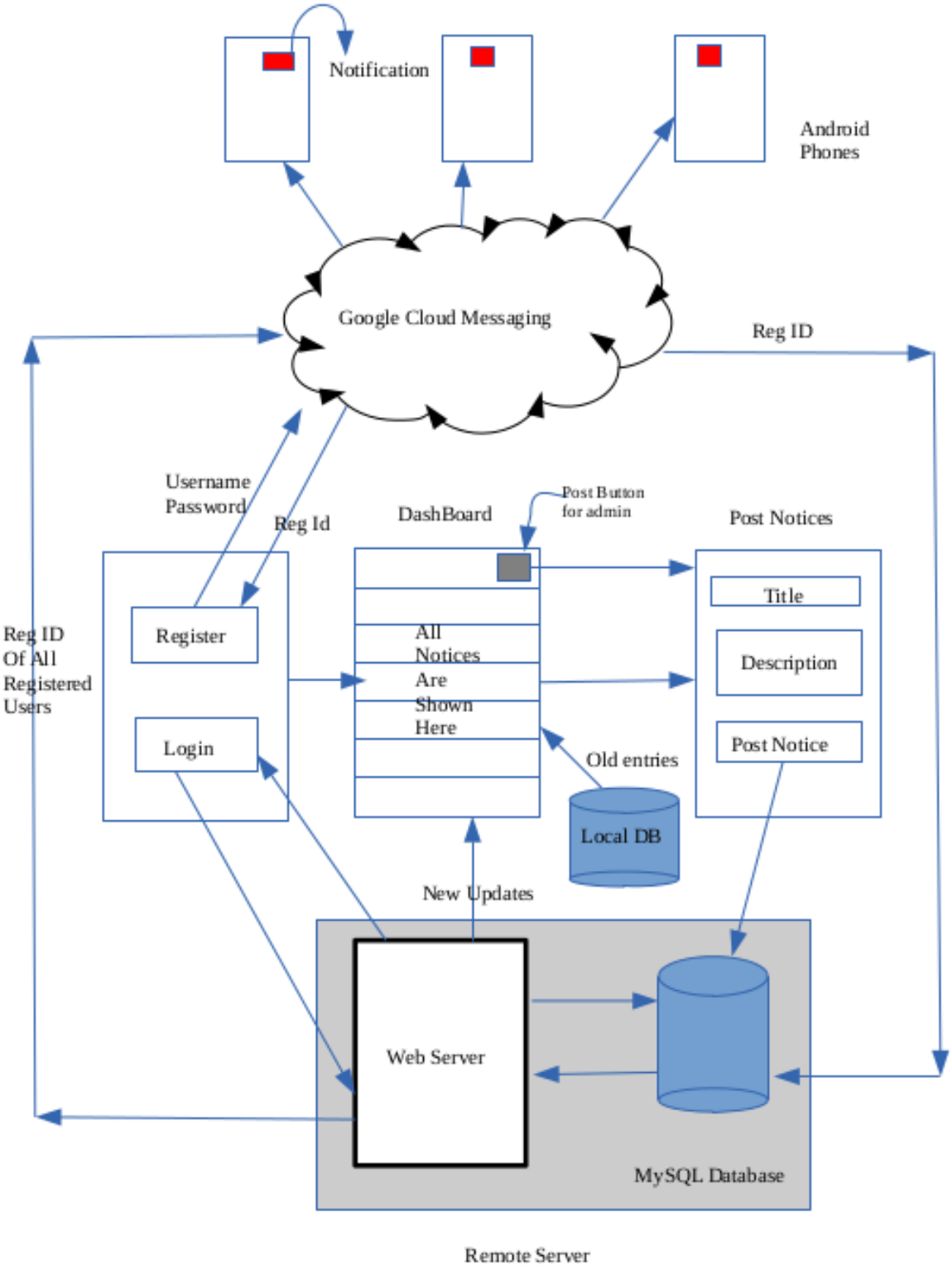
If the user does not want some notices, he/she can delete it from their phone. There will be no effect on server entry.

6. **Posting a Notice**:

If a user is an admin, he is able to post the notice. In order to post the notices, he has three options. One option is that, he can post a simple text notice. Another option allows him to send some attachment image with the notice. In this, he has two options. Either he can pick the image from the gallery or he can click a picture on the spot by using camera. After that, press the post button to post the notice.

7. Notification Buzz:

As soon as the admin post a notice, the script will run with which request is made by GCM Server to Web Server for all the registered IDs. After getting all the registered IDs, notification is sent to all the users registered with this application. Notification has a tune and vibration that runs whenever there is a notification received by the user from GCM Server.


Notification
Android
Phones
Google Cloud Messaging
Reg ID
Username
Password
Reg Id
DashBoard
Post Button
for admin
Post Notices
Reg ID
Of All
Registered
Users
Register
Login
All
Notices
Are
Shown
Here
Title
Description
Post Notice
Old entries
Local DB
New Updates
Web Server
MySQL Database
Remote Server

3.3 System Design

The system design can be clearly explained from the following diagrams:

1. Use Case Diagram:

A Use Case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. This type of diagram is typically used in conjunction with the textual use case and will often be accompanied by other types of diagrams as well. There are two types of user in this application, user and admin. Following depicts their use case diagram:

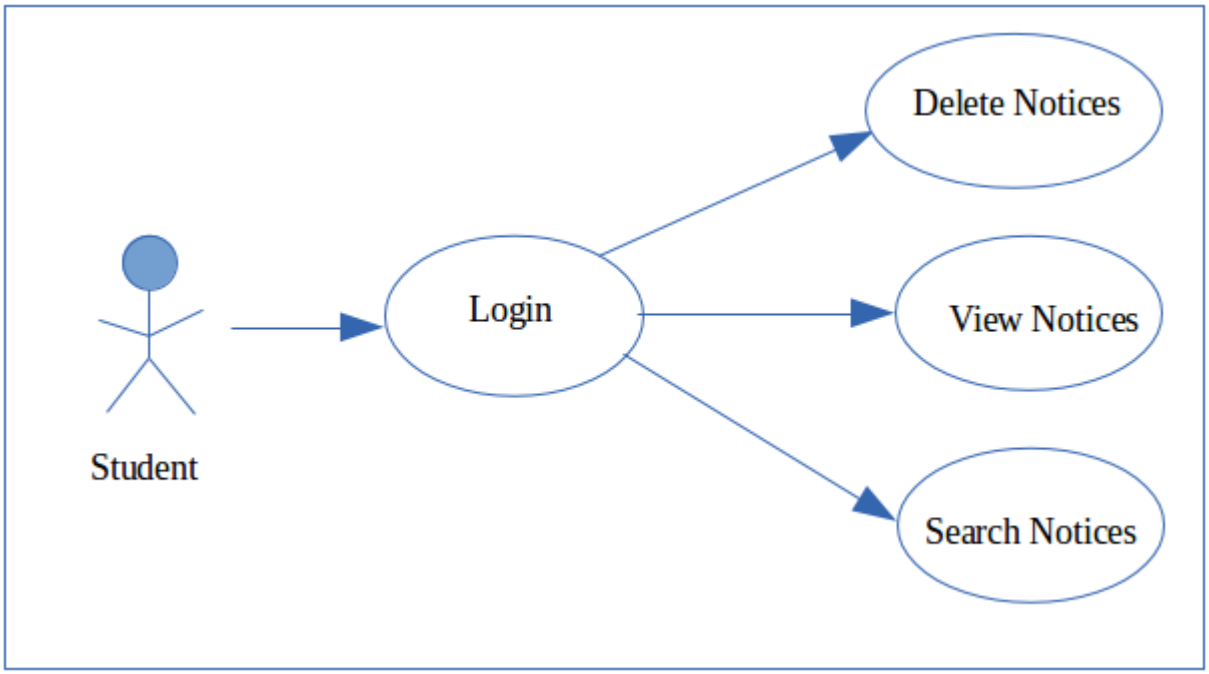


Figure 3.2: Use Case Diagram for User

This diagram is showing what a normal user can do with this application. The user can login, after that he can view the notices, delete the notices and can search for particular notices.

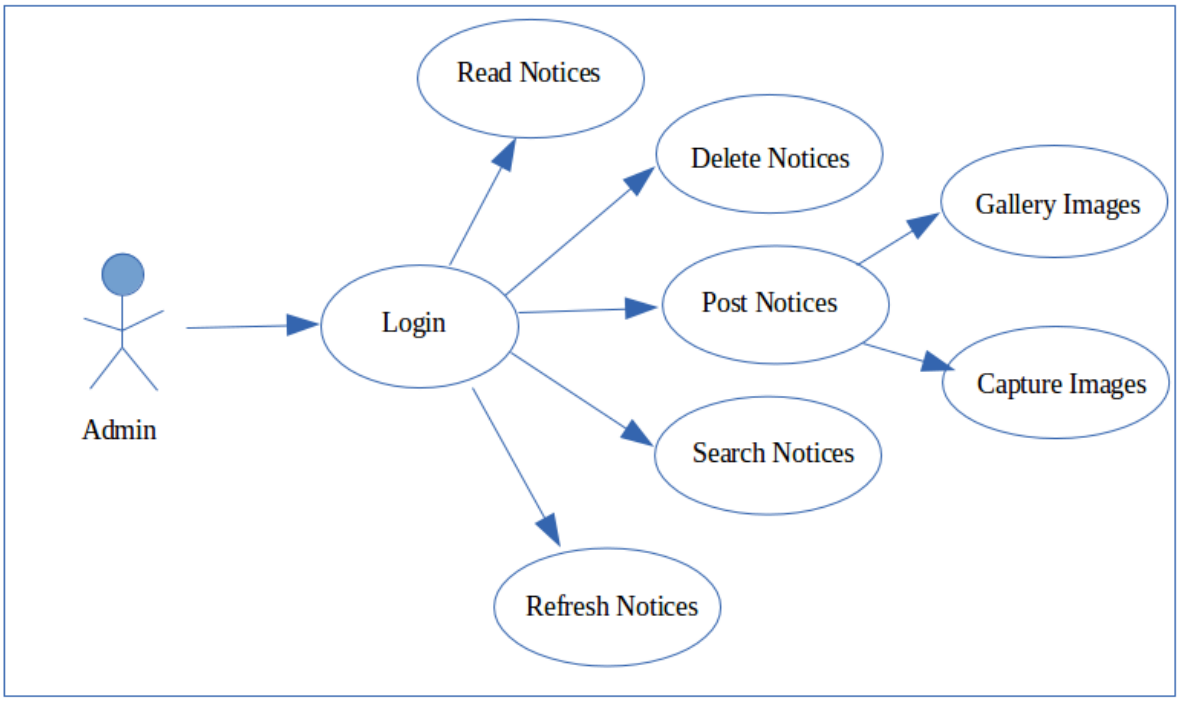


Figure 3.3: Use Case Diagram for Admin

This diagram shows the privileges of admin. An admin can post the notice in addition to viewing and deleting it. He can also post images as an attachment to the notices. Images can be chosen from the gallery or he can click the picture instantly using this application.

3.4 User Interface Design

User Interface Design means the design of application with which the user interacts. So it should be kept in mind that UI should be very simple and easy to use. It should be simple enough in look and feel also.

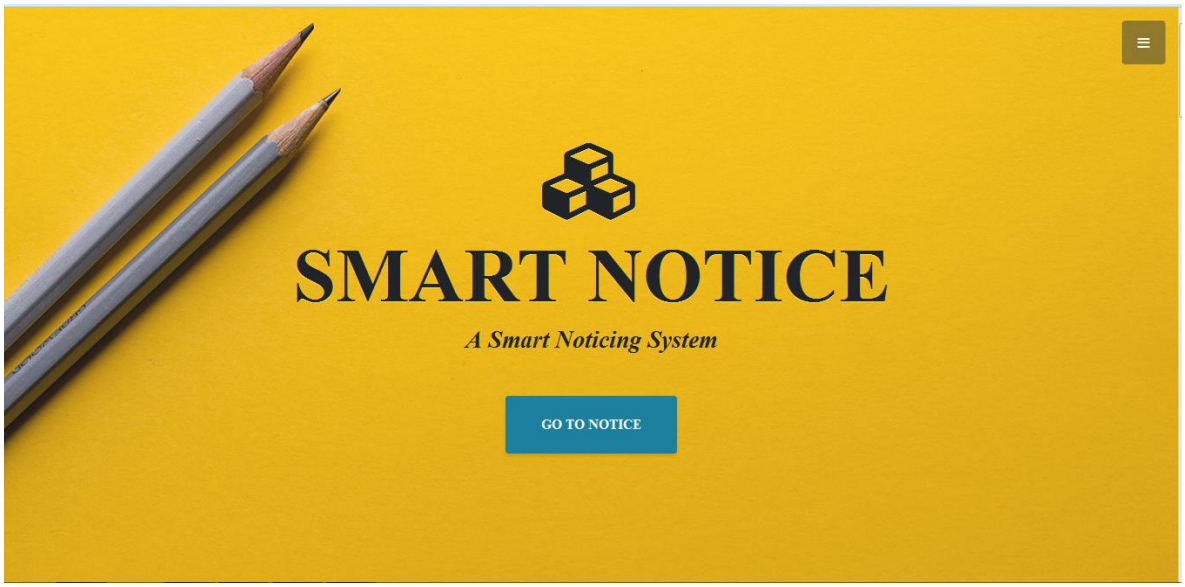


Figure 3.4: Front Page

This page is the first page which is presented to the user. It has two buttons, to login or to register.

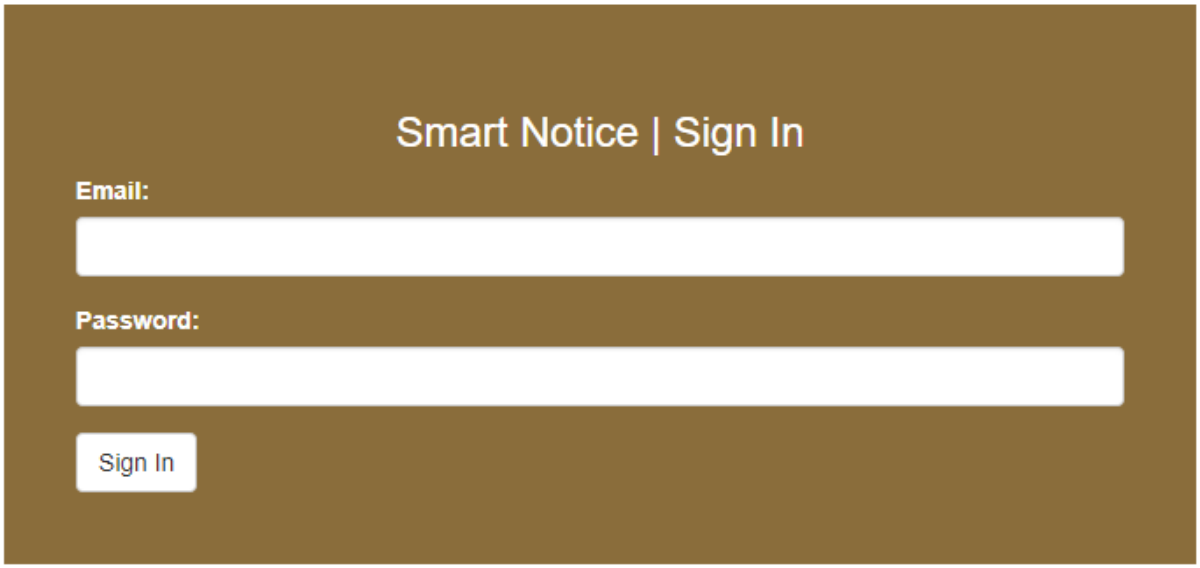


Figure 3.6: Sign in Page

This is the Sign in page where user enters his username and password in order to access the notices.

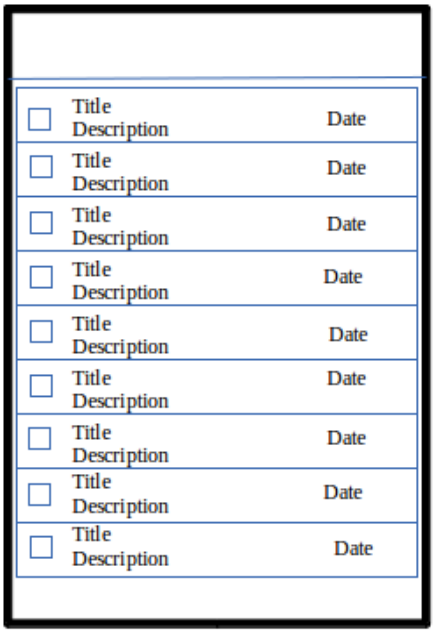


Figure 3.7: Dashboard of Notices

This is the main dashboard, which receives all the notices from the user. The user can click on any notice, to see its details. In addition to it, user can search and delete the notices.

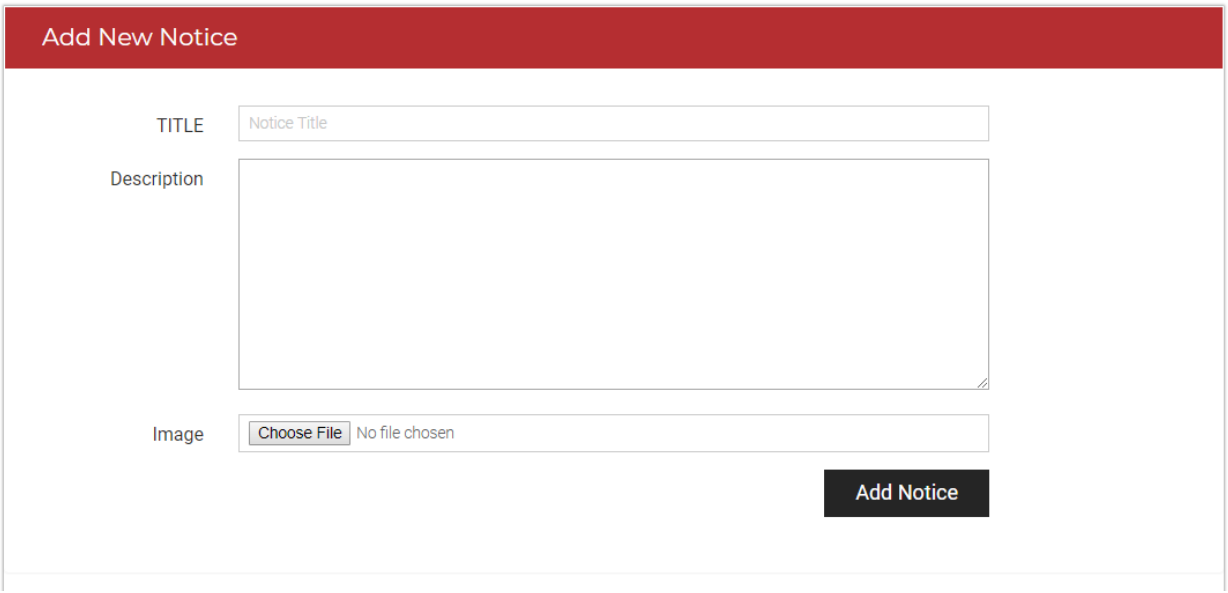


Figure 3.8: Add Notice Page

This is the add notice page, which we can add new notice and see all the new notice.

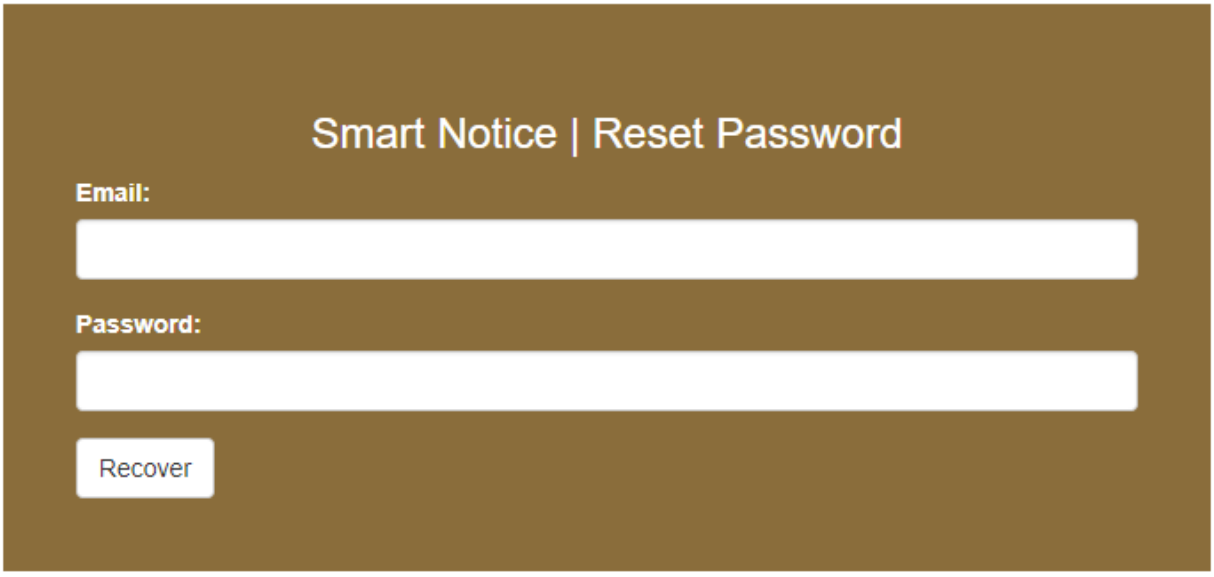


Figure 3.9 Reset Password Page

This is Reset Password page where the user can reset the password in order if he forgets his password.

3.5 ER Diagrams

An ER diagram is a diagram that helps to design databases in an efficient way.Attributes in ER diagrams are usually modeled as an oval with the name of the attribute, linked to the entity or relationship that contains the attribute.Within the relational model the final step can generally be broken down into two further steps, that of determining the grouping of information within the system, generally determining what are the basic objects about which information is being stored, and then determining the relationships between these groups of information, or objects.

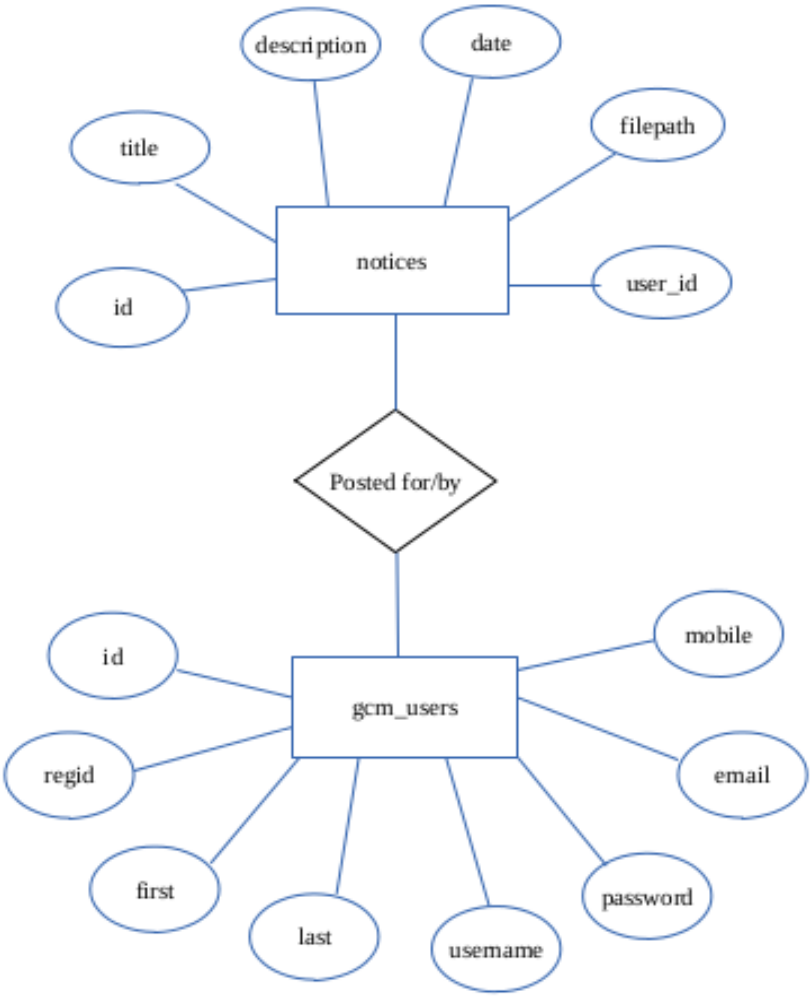


Figure 3.10: ER Diagram

3.6 Database Design

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data. In the relational model these are the tables and views. In an object database the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the database management system (DBMS).The process of doing database design generally consists of a number of steps which will be carried out by the database designer. Usually, the designer must:

1. Determine the relationships between the different data elements.
2. Superimpose a logical structure upon the data on the basis of these relationships.

Design process

1. **Determine the purpose of the database** –
   This helps prepare for the remaining steps.

2. **Find and organize the information required** –
   Gather all of the types of information to record in the database, such as product name and order number.

3. **Divide the information into tables –**

Divide information items into major entities or subjects, such as Products or Orders. Each subject then becomes a table.

4. **Turn information items into columns** –

Decide what information needs to be stored in each table. Each item becomes a field, and is displayed as a column in the table. For example, an Employees table might include fields such as Last Name and Hire Date.

5. **Specify primary keys –**

Choose each table's primary key. The primary key is a column, or a set of columns, that is used to uniquely identify each row. An example might be Product ID or Order ID.

6. **Set up the table relationships –**

Look at each table and decide how the data in one table is related to the data in other tables. Add fields to tables or create new tables to clarify the relationships, as necessary.

7. **Refine the design –**

Analyze the design for errors. Create tables and add a few records of sample data. Check if results come from the tables as expected. Make adjustments to the design, as needed.

8. **Apply the normalization rules –**

Apply the data normalization rules to see if tables are structured correctly. Make adjustments to the tables.

	#	Name	Type	Collation	Attributes	Null	Default	Extra
☐	1	**notice_id**	int(11)			No	*None*	AUTO_INCREMENT
☐	2	**title**	varchar(255)	utf8_general_ci		No	*None*	
☐	3	**description**	text	utf8_general_ci		No	*None*	
☐	4	**date_time**	datetime			No	*None*	
☐	5	**image**	text	utf8_general_ci		No	*None*	

Figure: Notice table

	#	Name	Type	Collation	Attributes	Null	Default	Extra
☐	1	**user_id**	int(11)			No	*None*	AUTO_INCREMENT
☐	2	**user_name**	varchar(255)	utf8_general_ci		No	*None*	
☐	3	**user_email**	varchar(255)	utf8_general_ci		No	*None*	
☐	4	**user_pass**	varchar(255)	utf8_general_ci		No	*None*	

Figure: User table

3.7 Cost Estimation

The following factors will draw financial expenses for the proposed system to be a fully-functional software:

1. Domain Name and Web Space cost.
2. Cost for app location registration on respective markets.
3. Platform and language specific trainings and courses.

Chapter 4
Conclusion and Future Scope

4.1 Conclusion

I learned a lot by doing this project.

1. Operating system: Linux
2. Languages used: Java script, PHP
3. Servers Used: Apache
4. Database: MySQL

So during this project I learned all the above things. Before this project, I had no idea about Java script and PHP. But now I learned PHP for server side scripting. If I talk about the project, smart notice Application has reduced lot of manual work. It has made notifying each and every user very easy and that too with no time and place here.

4.2 Future scope

The future scope of the project is that it can be used as any news giving application or it can be used to advertise your products, telling the customers about new schemes and products coming to your shop. This application of e-Notice can be further extended to include the following features:

- **Categorization of Notice:**

Notices can be categorized in different categories, so that its possible for user to easily manage the notices. Categorization can also be done by making groups. Defining the notice to be circulated in a particular group can make it more secure.

- **Documents and PDF files:**

The attachments can be further improved to include PDF files or Doc files. Then there will not be much need to send images with the notices. A single file would serve all the purposes.

- **Feedback**:

Feedback on the notices can also be taken. It can increase communication among connected members and any issue can be easily sorted out on the spot.

References

[1] Anemuller, J. and Gramss, T.: On-line blind separation of moving sound sources. *Proc. of Int. Conf. on Independent Component Analysis and Blind Source Separation (ICA'99)*, pp: 331-334, 1999.

[2] Asano, F., Goto, M., Itou, K. and Asoh, H.: Real-time sound source localization and separation system and its application to automatic speech recognition. *Proc. of Eurospeech01*, pp: 1013-1016, 2001.

[3] Allen, J. B.: How do humans process and recognize speech? *IEEE Transaction on Speech and Audio*, 2(4), pp: 567-577, 1994.

[4] Brown, G. J. and Cooke, M.: Computational auditory scene analysis. *Computer Speech Language*, Vol. 8(4), pp: 297-336, 1994.

[5] Bofill, P.: Underdetermined blind separation of delayed sound sources in the frequency domain. *Neurocomputing*, Vol. 55, No. 3/4, pp: 627-641, 2003.

[6] Boll, S. F.: Suppression of acoustic noise in speech using spectral subtraction. IEEE Trans. on Acoustic, Speech and Signals Processing, Vol. 27, pp: 113-120, 1979.

[7] Bregman, A. S.: Auditory scene analysis. MIT Press, Cambridge, 1990.

[8] Bregman, A. S.: Auditory Scene Analysis: The perceptual organization of sound. MIT press, 2nd edition, 1999.

[9] Breebaart, J., Van de Par, S. and Kohlrausch, A.: Binaural processing model on contralateral inhibition. I. Model structure. *Journal of Acoustical Society of America.* 110, pp: 1074-1088, 2001

[10] Baeck, M. and Zolzer, U.: Real-time implementation of source separation algorithm. *Proc. of DAFx-03*, pp: 29-34, 8-11 Sep, 2003.