1. Project overview

We define Discourse as web-based service that allows individuals to construct a public or semi-public profile within a bounded system, articulate a list of other users with whom they share a connection, and view and traverse their list of connections and those made by others within the system. The nature and nomenclature of these connections may vary from site to site.

Since their introduction, social discussion sites (DS) such as StackExchange, Quora, Reddit, Brainly, etc have attracted millions of users, many of whom have integrated these sites into their daily practices. As of this writing, there are hundreds of DS, with various technological affordances, supporting a wide range of interests and practices. While their key technological features are fairly consistent, the cultures that emerge around DSs are varied. Most sites support the maintenance of pre- existing forums, but others help strangers connect based on shared interests, political views, or activities. Some sites cater to diverse audiences, while others attract people based on common language or shared racial, sexual, religious, or nationality- based identities. Sites also vary in the extent to which they incorporate new information and communication tools, such as mobile connectivity, blogging, and photo/ video-sharing.

A DS service is an online service, platform, or site that focuses on facilitating the building of discussion forums or social relations among people who, for example, share interests, activities, backgrounds, or real-life connections. A DS service consists of a representation of each user (often a profile), his/her social links, and a variety of additional services.

Its main goal is to solve problems that arise in student’s life. Those problems might be related to day to day activities or basically just an educational.

2. The Purpose of the Project

2.1 Background

We are living in 21st century and the way of learning new things certainly has changed. This would not have been possible without web based technology. So, through this platform users can ask questions about their interests and problems that are related to curriculum. Hopefully, this project will cover these problems. Solutions are provided by the people who are good in their fields. Basically, interested user can make problem really easy for other users by answering their questions. Other solutions are notes, videos which are mutually shared.

2.2 Main objectives

The primary objectives of this project are:

a) To standardize the traditional way of learning through online based educational discussion platform.

b) To create an enlisted group (or a circle) that actively participates in genuine discussions.

c) To help learners access “Vlogging” feature (an inclusion of blogs through videos).

d) To get access of resources (files) that can be viewed publicly.

3. Requirement specification:

3.1 Functional Specifications

- This system provides users to register their profile.
- This system provides the mechanism for connecting users by follow mechanism.
  - User can follow other user(s).
  - User can unfollow other user(s).
- This system provides users to send a scrap messages to their followers or to user who followed them.
- This system provides searching mechanism to find other people accounts easily.
- Users can post videos.
- Enlisted user can ask questions.
- Enlisted user can answer questions.
- Answers can be upvoted for confidence and genuine.
- Any unauthenticated user can search notes, get access to embedded videos and download.
- Any unauthenticated user can get access to files/study materials.

3.2 Non-Functional Specification

- Secure access of confidential data by user name and password. This application is secure for every kind of its users, because if any user logout from any session then nobody will be able to access his profile without knowing his confidential password.

- Better component design to get better performance at peak time.

- The database used here is robust, reliable & fast. So users will have to wait for the output very short time.

- This application can be accessed from any type of platform.

- There is no case of redundancy in the database so it will not take extra memory space.

3.3 Software Tools Specification

- **MySQL**

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used WAMP open source web application software stack (and other 'AMP' stacks). WAMP is an acronym for "Windows, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL.

MySQL can be built and installed manually from source code, but this can be tedious so it is more commonly installed from a binary package unless special customizations are required.

Though MySQL began as a low-end alternative to more powerful proprietary databases, it has gradually evolved to support higher-scale needs as well. It is still most commonly used in small to medium scale single-server deployments, either as a component in a WAMP-based web application or as a standalone database server. Much of MySQL's appeal originates in its relative simplicity and ease of use, which is enabled by an ecosystem of open source tools such as phpMyAdmin. In the medium range, MySQL can be scaled by deploying it on more powerful hardware, such as a multi-processor server with gigabytes of memory.

There are however limits to how far performance can scale on a single server ('scaling up'), so on larger scales, multi-server MySQL ('scaling out') deployments are required to provide improved performance and reliability. A typical high-end configuration can include a powerful master database which handles data write operations and is replicated to multiple slaves that handle all read operations. The master server synchronizes continually with its slaves so in the event of failure a slave can be promoted to become the new master, minimizing downtime. Further improvements in performance can be achieved by caching the results from database queries in memory using memcached, or breaking down a database into smaller chunks called shards which can be spread across a number of distributed server clusters.

- **Language: PHP**

PHP is a scripting language designed to fill the gap between SSI (Server Side Includes) and Perl, intended for the web environment. Its principal application is the implementation of web pages having dynamic content. PHP has gained quite a following in recent times, and it is one of the frontrunners in the Open Source software movement. Its popularity derives from its C-like syntax, and its simplicity. PHP is currently divided into two major versions: PHP 4 and PHP 5, although PHP 4 is deprecated and is no longer developed or supplied with critical bug fixes. PHP 7 is currently launched. PHP was designed by RasmusLerdorf to display his resume online and to collect data from his visitors.

PHP allows a static webpage to become dynamic. "PHP" is an acronym that stands for "PHP: HypertextPreprocessor". The word "Preprocessor" means that PHP makes changes before the HTML page is created. This enables developers to create powerful applications which can publish a blog, remotely control hardware, or run a powerful website such as Wikipedia or Wikibooks. Of course, to accomplish something such as this, you need a database application such as MySQL.

PHP code is interpreted by a web server with a PHP processor module, which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a interface capability and can be used in standalone graphical applications.

The PHP language was originally implemented as an interpreter, and this is still the most popular implementation. Several compilers have been developed which decouple the PHP language from the interpreter. Advantages of compilation include better execution speed, static analysis, and improved interoperability with code written in other languages.

PHP includes free and open source libraries with the core build. PHP is a fundamentally Internet-aware system with modules built in for accessing File Transfer Protocol (FTP) servers, many database servers, embedded SQL libraries such as embedded PostgreSQL, MySQL, Microsoft SQL Server and SQLite, LDAP servers, and others. Many functions familiar to C programmers such as those in the studio family are available in the standard PHP build.

- **Back End: PHP MyAdmin**

PHPMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. PhpMyAdmin supports a wide range of operations on MySQL, Maria DB and Drizzle. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc.) can be performed via the user interface, while you still have the ability to directly execute any SQL statement.

Features provided by the program include:

a. Web interface
b. MySQL database management

Import data from CSV and SQL

a. Export data to various formats: CSV, SQL, XML, PDF (via the TCPDF library), ISO/IEC 26300 - Open Document Text and Spreadsheet, Word, Excel, LaTeX and others
b. Administering multiple servers
c. Creating PDF graphics of the database layout
d. Creating complex queries using Query-by-Example (QBE)
e. Searching globally in a database or a subset of it
f. Transforming stored data into any format using a set of predefined functions, like displaying BLOB-data as image or download-link
g. Live charts to monitor MySQL server activity like connections, processes, CPU/Memory usage, etc.

- **Laravel Framework**

Laravel is a free, open-source PHP web framework, created by Taylor Otwell and intended for the development of web applications taking after the model-view-controller (MVC) architectural pattern. Some of the features of Laravel are a secluded packaging system with a dedicated dependency manager, different courses for accessing relational databases, utilities that guide in application deployment and maintenance, and its orientation toward syntactic sugar.

Some of the key point feature of laravel:

a. Restful routing.
b. Inherent Database Version control
c. A lightweight Blade Templating Engine

d. Composer – An amazing tool that lets us to manage our application’s third-party packages easily.
e. Comes bundled with Eloquent –
f. Built-in unit testing and simply readable impressive syntax
g. Larger Community catering to thousands of programming geeks and application developers
h. Intelligently designed to offer incredible flexibility to developers that help them create each and everything from small sites to giant enterprise applications.

4. System requirements

4.1 Hardware Requirements

Processor	:	Intel Pentium IV 2.0 GHz and above
RAM	:	512 MB and above
Hard disk	:	80GB and above
Monitor	:	CRT or LCD monitor
Keyboard	:	Normal or Multimedia
Mouse	:	Compatible mouse

4.2 Software Requirements

Front End	:	XAMPP/WAMP
Language	:	PHP
Back End	:	PHP MyAdmin
Operation System	:	Windows XP or above
Browser	:	Any latest browser

5. Design Specifications

5.1 Modular Design

This application comprises the following major modules:

- **User Module**

This module manages the user’s account information. It is used to retrieve users credentials from the database and finally to display in web view. It also consists the follow relationship between users to users.

- **File Module**

This Module is used to retrieve the file information. The uploaded file has pathname, faculty related to that file, subject of related file attributes. All these details are stored in database and File module retrieves these information for application use.

5.2 System Design

5.2.1 Use Case Diagram

A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Web site. Use case diagrams are employed in UML (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems.

- The actors, usually individuals involved with the system defined according to their roles.
- The use cases, which the specific roles are played by the actors within and around the system.
- The relationships between and among the actors and the use cases.

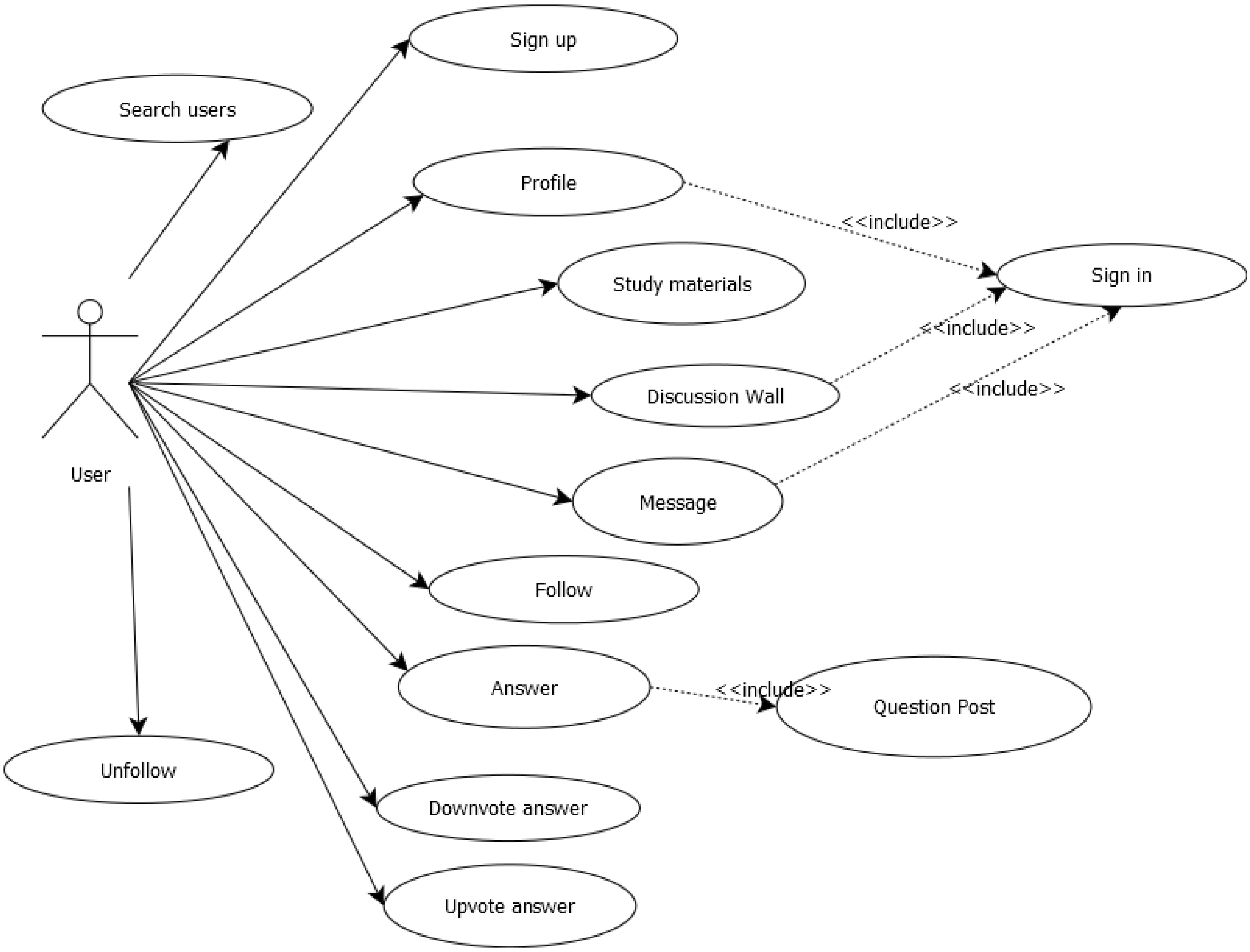
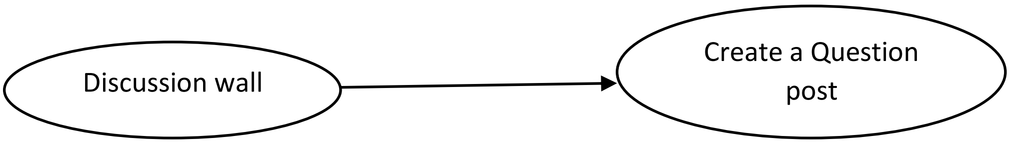


Fig 5.2.1: Use case diagram

Use case description:

- **Signup**
  Signup option Lets user to create new account; user can fill up form which consist email, username and password field.
- **Discussion Wall**
  Lets the user to see the posts of followers or the people they follow, and thereby allows them to create a new question post or answers in the existing question post. To post any question user have to be authenticated.



- **Answer**

Each questions can hold multiple answers. Answers can be upvoted and downvoted.

- **Search**
  Search option lets the user to see search with username (or even name).

- **Profile view**

Profile view lets the user to see their timeline and statuses posted. Only those users who are authenticated can view their profile.

- **Message**
  Any users who are mutually connected to each other can send messages.

- **Logout**
  Allows the user to log out of the session, user is redirected to home page.

- **Follow**
  Any user can follow other user but they have to be authenticated.

- **Unfollow**
  Unfollowing the user deletes the relationship between users from database.

- **Study Materials**
  Any user can access study materials. Simply, they don't have to be authenticated user. It contains all notes and documents for related faculty and subjects. It has another user case:

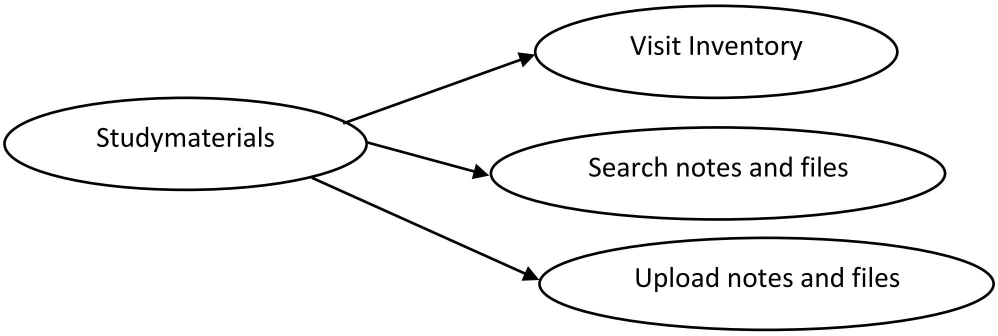


Fig: Study material use case description

a. Visit Inventory
   It contains all the files, notes and document.

b. Search notes and files
   User can search any files or notes based on the faculty name or subject name.

c. Upload
   Any user can upload notes and files. User have to specify the name of faculty and subject.

5.2.2 Context diagram

Context diagram represents how system reacts with environment. Physical view of our system has following design.

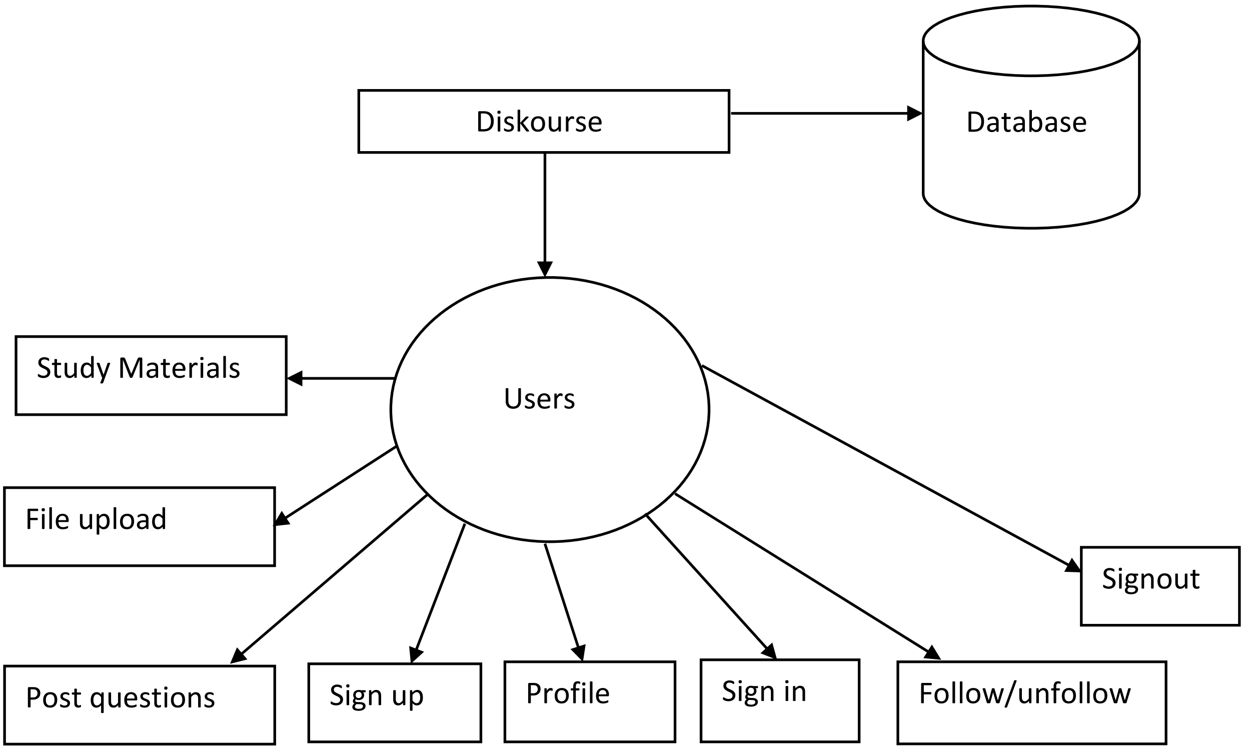


Figure 5.2.2: Context diagram

5.2.3 Entity Relationship (ER) diagram

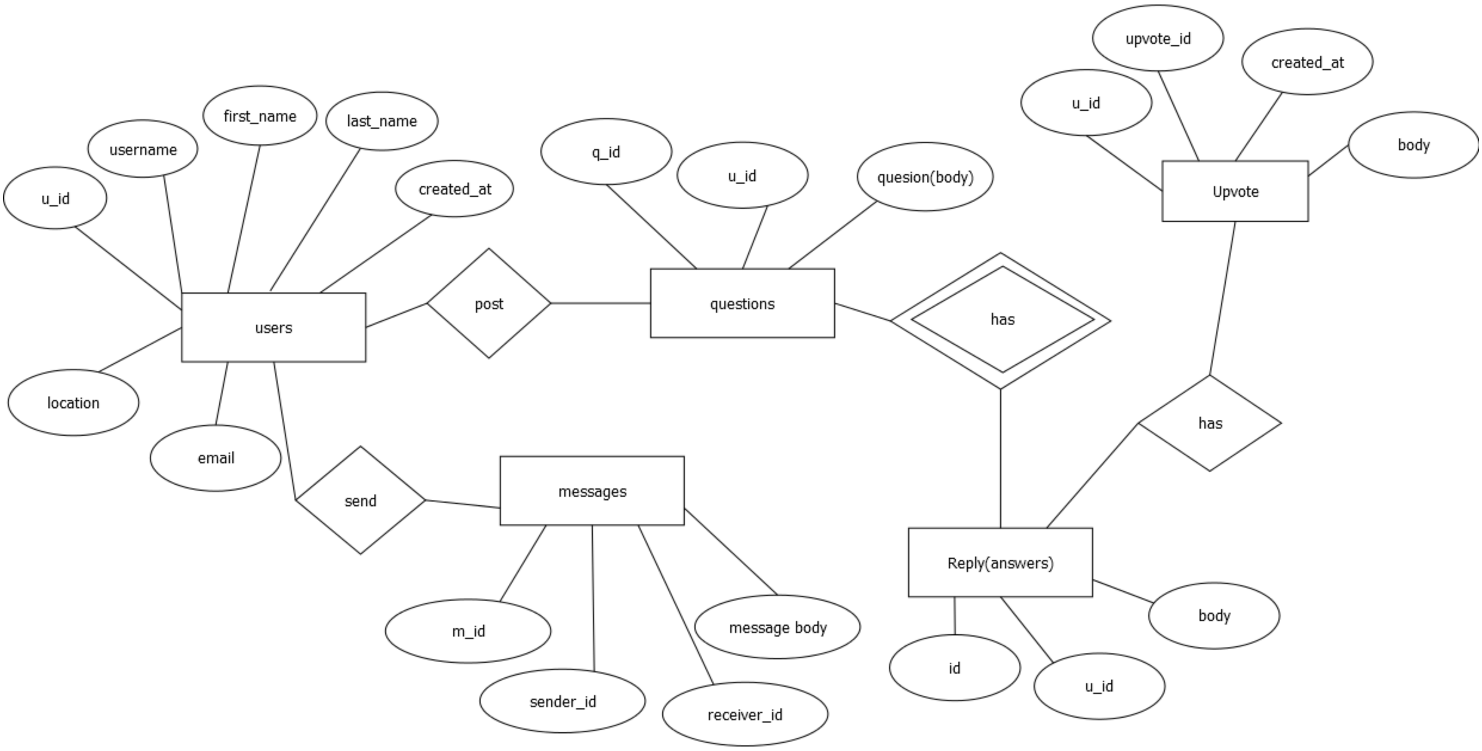


Fig 5.2.3: E-R diagram

6. Implementation Details

6.1 Screenshots

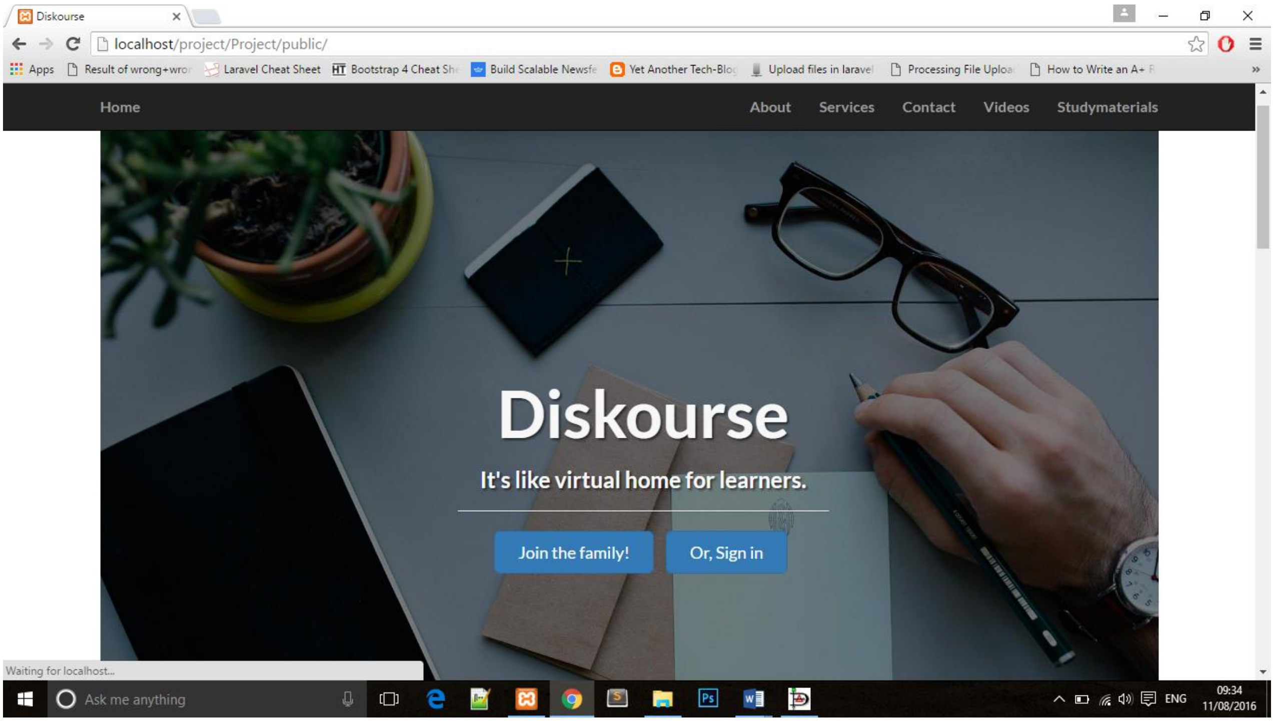
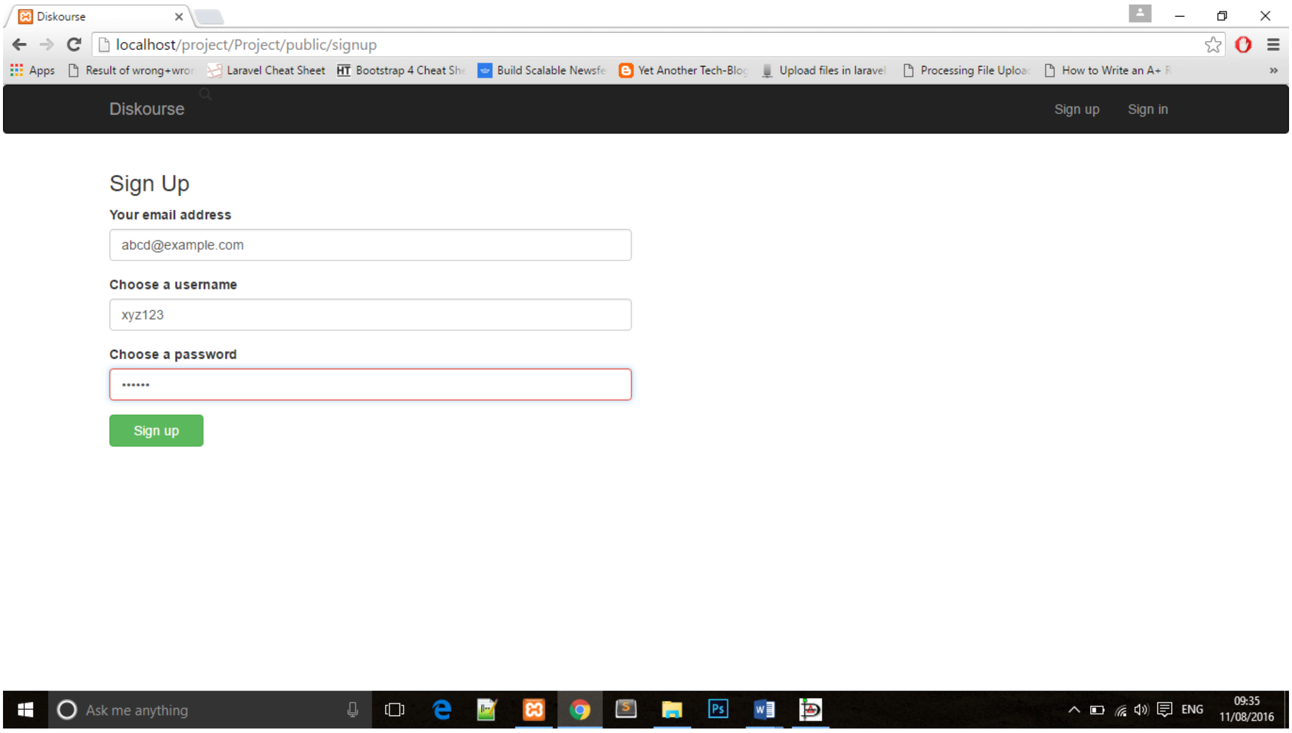


Fig 6.1.a: Home page



Sign Up

Your email address

abcd@example.com

Choose a username

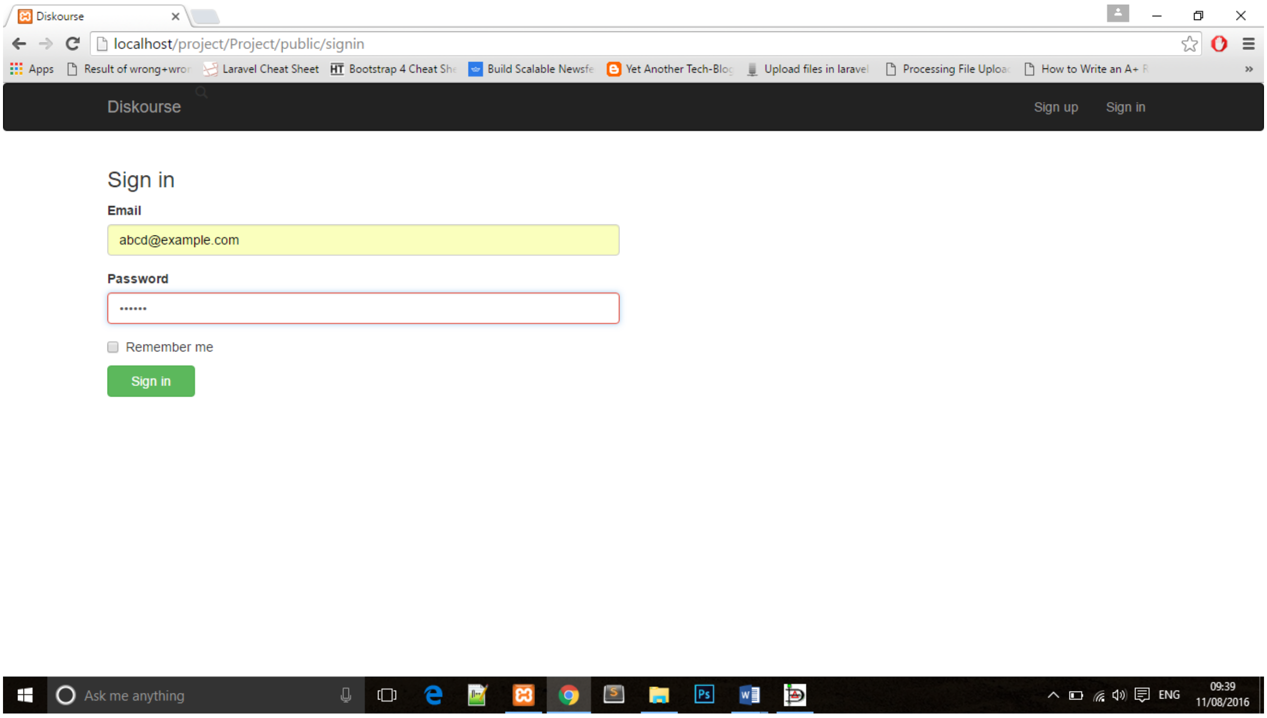
xyz123

Choose a password

••••••

Sign up

Fig 6.1.b: Sign up



Sign in

Email

abcd@example.com

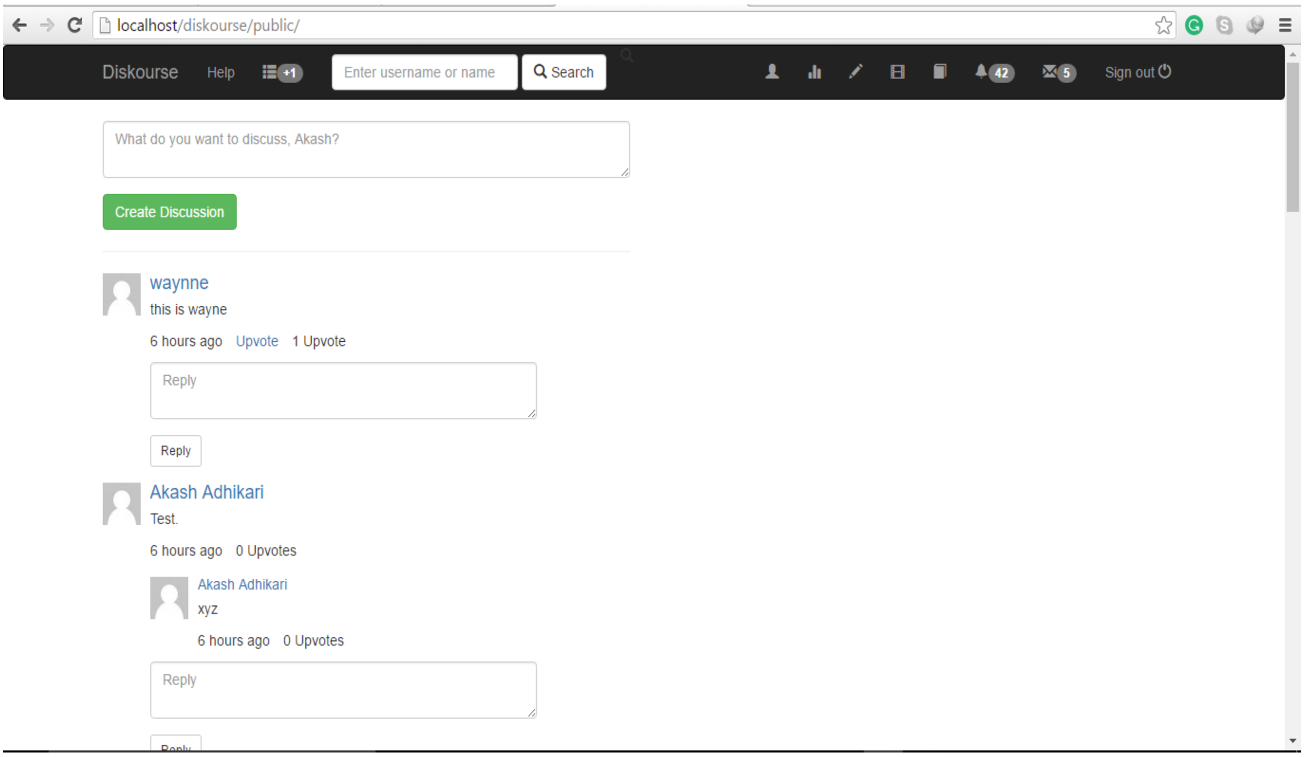
Password

••••••

□ Remember me

Sign in

Fig 6.1.c: Sign in



What do you want to discuss, Akash?

Create Discussion

waynne

this is wayne

6 hours ago Upvote 1 Upvote

Reply

Reply

Akash Adhikari

Test.

6 hours ago 0 Upvotes

Akash Adhikari

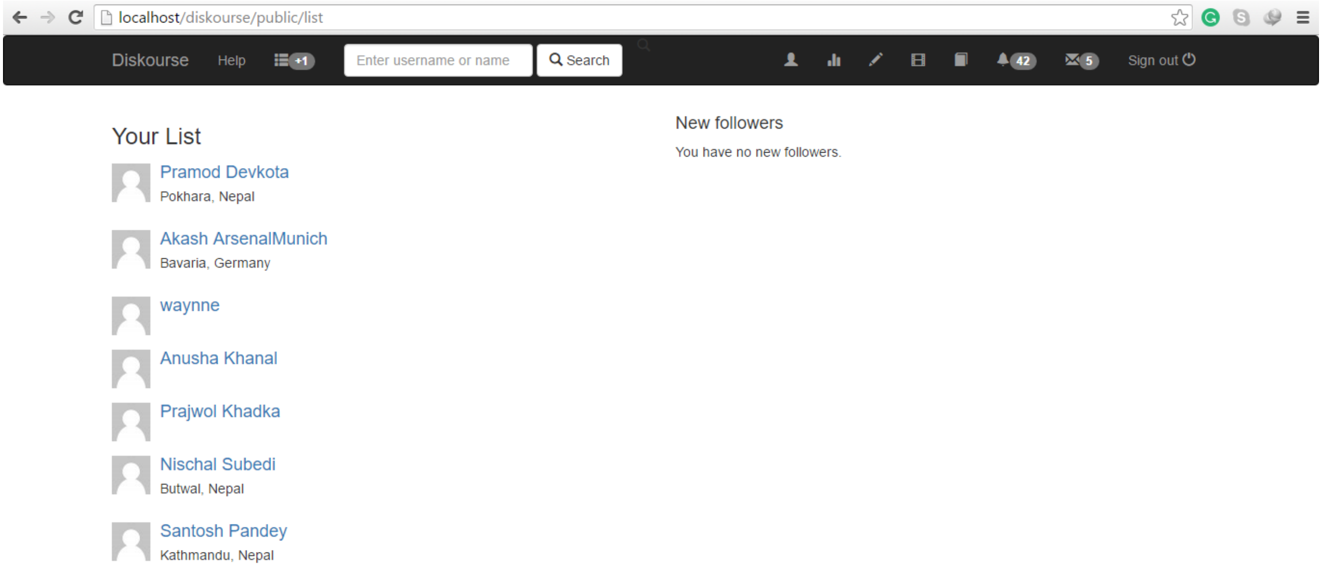
xyz

6 hours ago 0 Upvotes

Reply

Reply

Fig 6.1.c: Home page



localhost/diskourse/public/list

Your List

Pramod Devkota
Pokhara, Nepal

Akash ArsenalMunich
Bavaria, Germany

waynne

Anusha Khanal

Prajwol Khadka

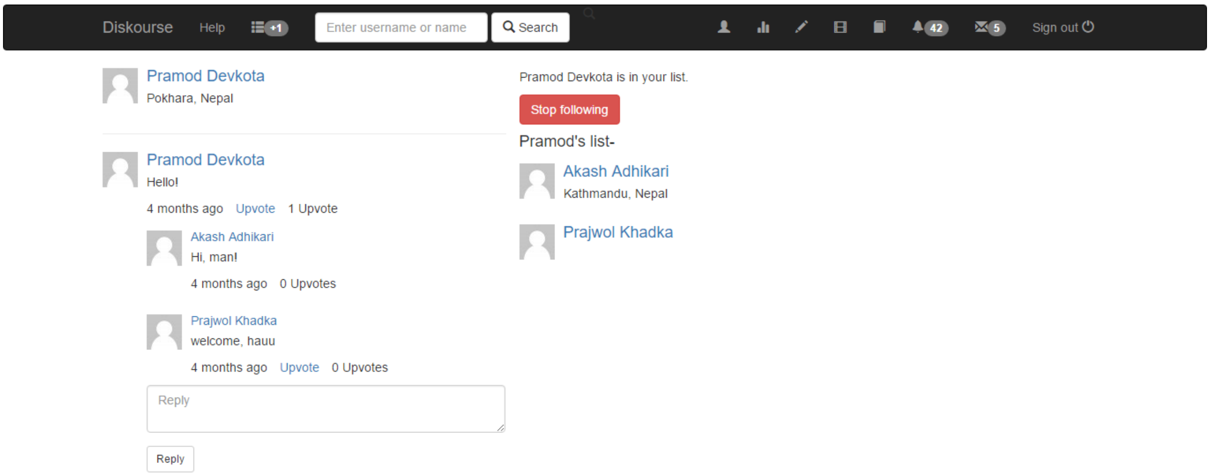
Nischal Subedi
Butwal, Nepal

Santosh Pandey
Kathmandu, Nepal

New followers

You have no new followers.

Fig 6.1.d: Users within list loop



Pramod Devkota
Pokhara, Nepal

Pramod Devkota
Hello!
4 months ago Upvote 1 Upvote

Akash Adhikari
Hi, man!
4 months ago 0 Upvotes

Prajwol Khadka
welcome, hauu
4 months ago Upvote 0 Upvotes

Reply

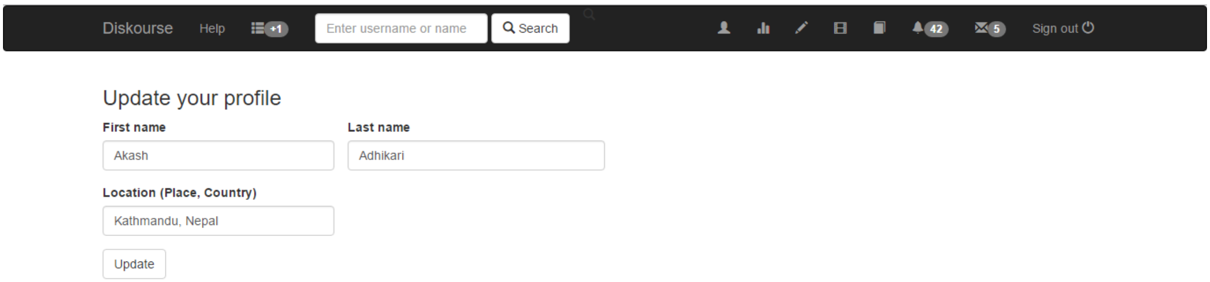
Pramod Devkota is in your list.

Pramod's list-

Akash Adhikari
Kathmandu, Nepal

Prajwol Khadka

Fig 6.1.e: Enlisters profile with “stop following” feature



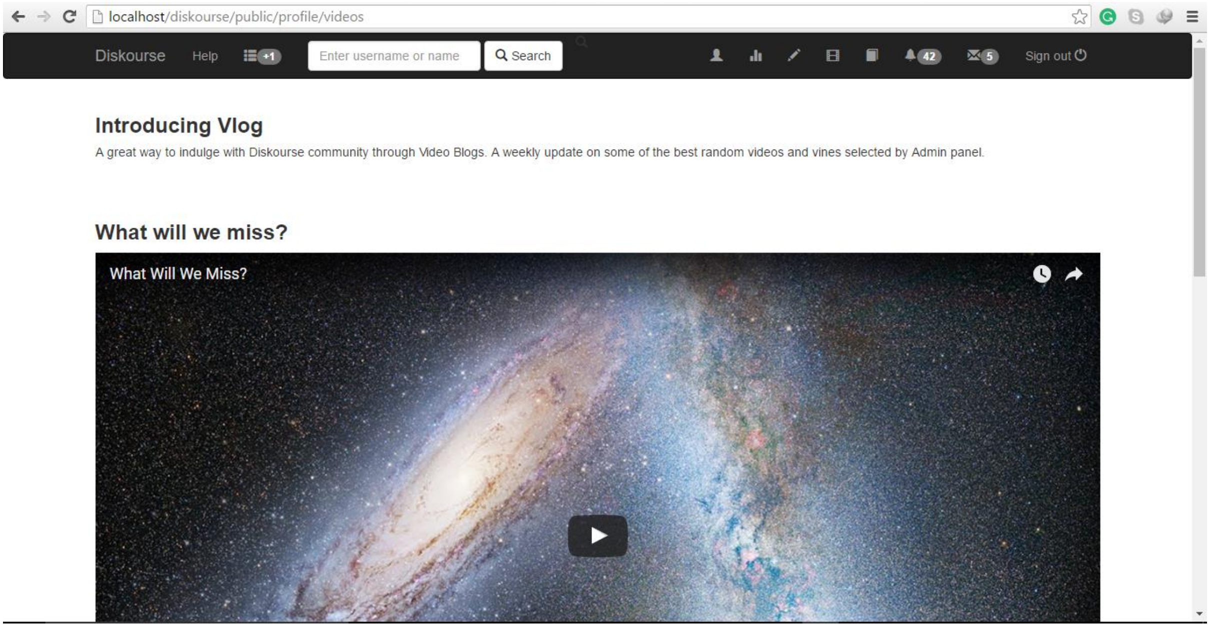
Update your profile

First name: Akash

Last name: Adhikari

Location (Place, Country): Kathmandu, Nepal

Fig 6.1.f: Profile update feature

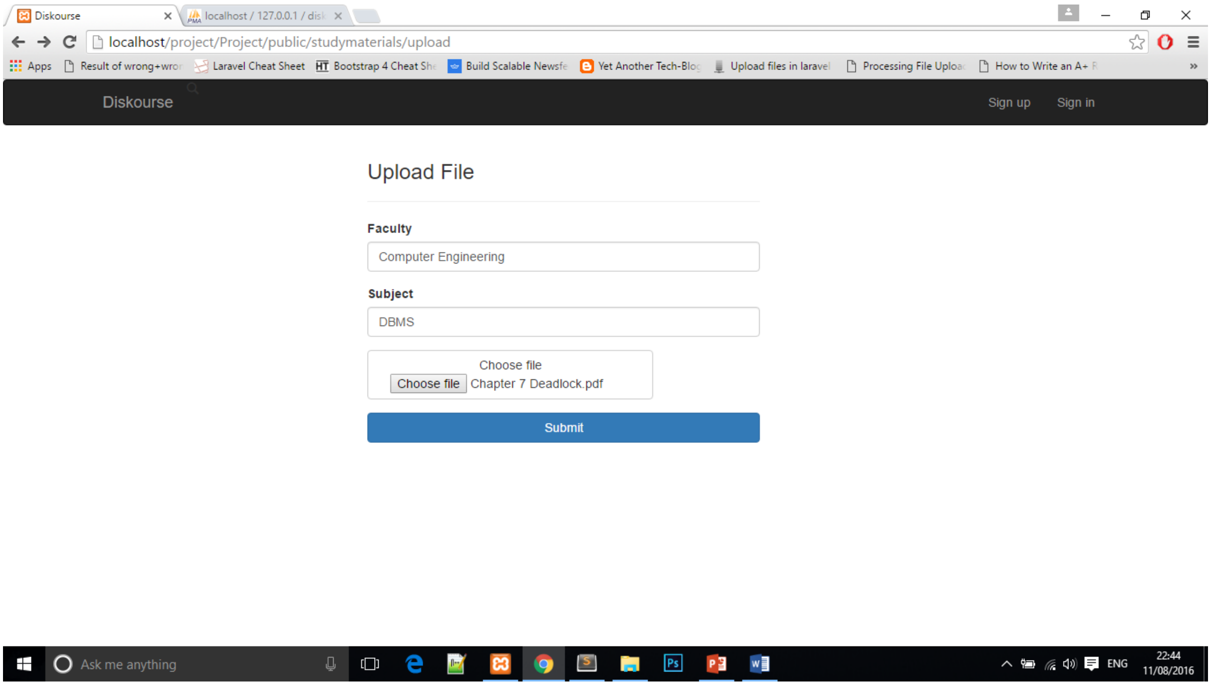


Introducing Vlog

A great way to indulge with Diskourse community through Video Blogs. A weekly update on some of the best random videos and vines selected by Admin panel.

What will we miss?

Fig 6.1.g: Vlog feature with responsive embed



localhost/project/Project/public/studymaterials/upload

Upload File

Faculty

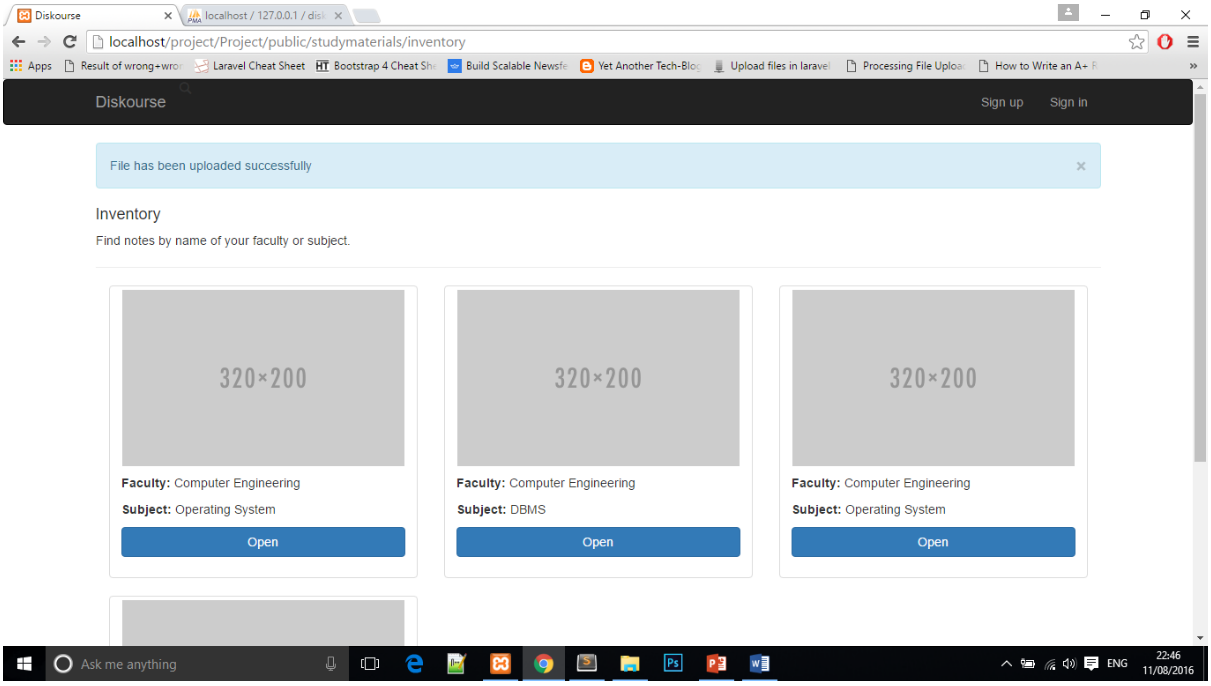
Computer Engineering

Subject

DBMS

Chapter 7 Deadlock.pdf

Fig 6.1.i: File upload



localhost/project/Project/public/studymaterials/inventory

File has been uploaded successfully ×

Inventory

Find notes by name of your faculty or subject.

Faculty: Computer Engineering
Subject: Operating System

Faculty: Computer Engineering
Subject: DBMS

Faculty: Computer Engineering
Subject: Operating System

Fig 6.1.j: Inventory section after file uploading file.

7. Testing and validation

- **System Testing**

System testing is a critical element of quality assurance and represents the ultimate review of analysis, design and coding. Test case design focuses on a set of techniques for the creation of test because that meet overall testing objective. When a system is developed it is hoped that it performs properly. The main purpose of testing an information system is to find the errors and correct them. The scope of system testing should include both manual and computerized operations. System testing is comprehensive evaluation of the programs, manual procedures, computer operations and controls.

System testing is the process of checking whether the developed system is working according to the objective and requirement. All testing is to be conducted in accordance to the test conditions specified earlier. This will ensure that the test coverage meets the requirements and that testing is done in a systematic manner.

It's the process of analyzing the software item to detect the differences between existing or required condition and evaluate the features of the software items. The thorough testing of the system before release of the software needs to be done vide the various test cases and modes so that the software becomes devoid of bugs and uses minimum space requirements as well as minimum time to perform. The test cases were selected beforehand with expected results defined and actual results recorded for comparison. The selection of test cases is done vide "White Box Testing" technique to check the internal programming logic and efficiency and vide "Black Box Testing" technique to check software requirement fulfillment with intension of finding maximum number of errors with minimum effort and time. Although test cases are a design by considering the cyclomatic complexity, conditional test, still the software code is not in its optional form, as all other possible alternative parts in the software are not considered. At the integration level, the software will be passing to the third party tests which would further enhance the software optimality and efficiency.

- **Black Box Testing**:

The method of Black Box Testing is used by the software engineer to derive the required results of the test cases:

a. Black Box Testing alludes to test that are conducted at the software interface.
b. Black Box Test examines some fundamental aspect of a system with little regard for the internal logic structure of the software.

c. A limited number of important logical paths can be selected and exercised.
d. Important data structure can be probed for validity.

Black box testing was performed to find errors in the following categories:-

- Incorrect or missing functions
- Variable error
- Performance error

- **White Box Testing:**

White Box Testing is sometimes called Glass Box Testing. Using White Box Testing methods the software engineer can derive the following test cases:

a. Guarantee that all independent paths within a module have been exercised at least once.
b. Exercise all logical decisions on their true and false sides.
c. Execute all loops at their boundaries and within their operational bounds.
d. Exercise internal data structures to ensure the validity.50

In White Box Testing efforts were made to handle the following:-

- Number of input parameters equal to number of arguments.

- Parameters and arguments attributes match.

- Number of arguments transmitted is called modules equal to attributes of parameters.

- Unit system of argument transmitted is called modules equal unit system of parameter.

- Number of attributes and order of arguments to build in functions correct.

- Any references to parameters not associated to build in functions correct.

- Input only arguments altered.

- Global variable definition consistent across module.

- **Unit Testing:**

The unit testing is performed to test the validity of the individual units. This is done in the coding phase with the interactive testing. Thus it itself constitutes a majority of functionality test for each logical unit.

- **Integrity Testing:**

When all the development of all the units or modules is completed and integrated the integrity test phase is started. In this phase the interface between the modules are tested. This phase basically verifies whether inter module exchange of information and events are as per required system behavior.

- **Validation Testing:**

Tests were performed to find conformity with the requirements. Plans and procedures were designed to ensure that all functional requirements are satisfied. The software was alpha-tested. There are two goals in preparing test plans. Firstly, a properly detailed test plan demonstrates that the program specifications are understood completely. Secondly, the test plan is used during program testing to prove the correctness of the program.

8. Conclusion and further works

While developing the system a conscious effort has been made to create and develop a software package, making use of available tools, techniques and resources – that would generate a proper system for ONLINE DISCUSSION FORUMING.

While making the system, an eye has been kept on making it as user-friendly. As such one may hope that the system will be acceptable to any user and will adequately meet his/her needs. As in case of any system development process where there are a number of short comings, there has been some shortcomings in the development of this system also.

There are some of the areas of improvement which couldn’t be implemented due to limited time. One such feature was online chat where members can chat with his friends through this website. For now direct message is partially possible. While searching for other user auto suggest mechanism can be made which requires Ajax to PHP integration and little bit of jQuery. Video upload feature can also be implemented and integrated likewise.

Other features such as rewarding user with most voted answer, displaying the post related to certain topics or fields which user wants, will surely enhance the prospects of this project.

References

1) https://docs.phalconphp.com/en/latest/reference/wamp.html

2) http://stackoverflow.com/questions/19803671/accessing-records-in-descending-order-in-database

3) http://php.net/manual/en/function.mail.php

4) https://docs.phalconphp.com/en/latest/reference/xampp.html

5) https://www.laravel.com/docs

6) https://getbootstrap.com/components

7) https://laravel-recipes.com/file

Table of contents