

A Moodle Plugin for Socratic Questioning

By
Manish Chouhan
Roll No. 113050082

Under the guidance of
Prof. Sridhar Iyer

Outline

- Introduction
- Socratic Questioning
- Principle used in Socratic Method
- Moodle Integration
- Functional Requirements
- Implementation
- Functional Logic
- Limitation and Future work
- References

Introduction

Motivation

- On-line learning and distance education systems are gaining attention now a days, which can be available at any place and at any time.

Goal

- Create ITS Framework for Socratic Questioning as a plugin for moodle through which teacher can teach their course in distance learning environment.

Socratic Questioning

- Socratic questioning is basically a dialogue conversation between two teacher and student.
- The main purpose of this strategy is to fill the students' mind with questions;
- In this process they find answers on their own through a series of questions.
- This strategy based on the theory of Socrates "learning consists recollection of things that student already knows".

Objective of Socratic Questioning

- This method helps students to develop critical thinking attitude and inquiring attitude.
- Socratic Questioning helps the students to debug their own theories.
- This method uses questioning about student's existing beliefs. And lead their beliefs/ideas to a contradiction.

Types of Socratic Questioning

- Classic Socratic Method
- Modern Socratic Method

Classic Socratic Method [1]

- Classic Socratic Method uses questioning to lead on-going conversation towards contradiction of other person's belief.
- In this method neither questioner nor respondent (other person) knows the correct answer.
- The result of classic Socratic Method is failure to find a satisfactory answer to the primary question.
- If satisfactory answer is found, this method becomes the modern Socratic Method.
- This method is difficult to implement in real conversation because of handling the wide range of possibility of answers is impossible.

Modern Socratic Method [1]

- questioning is used to lead a student's mind to knowledge through small steps (series of questions).
- In this method teacher ask direct question that have predefines range of answers.
- Prompt student to answer correctly primary question before moving to the next question.

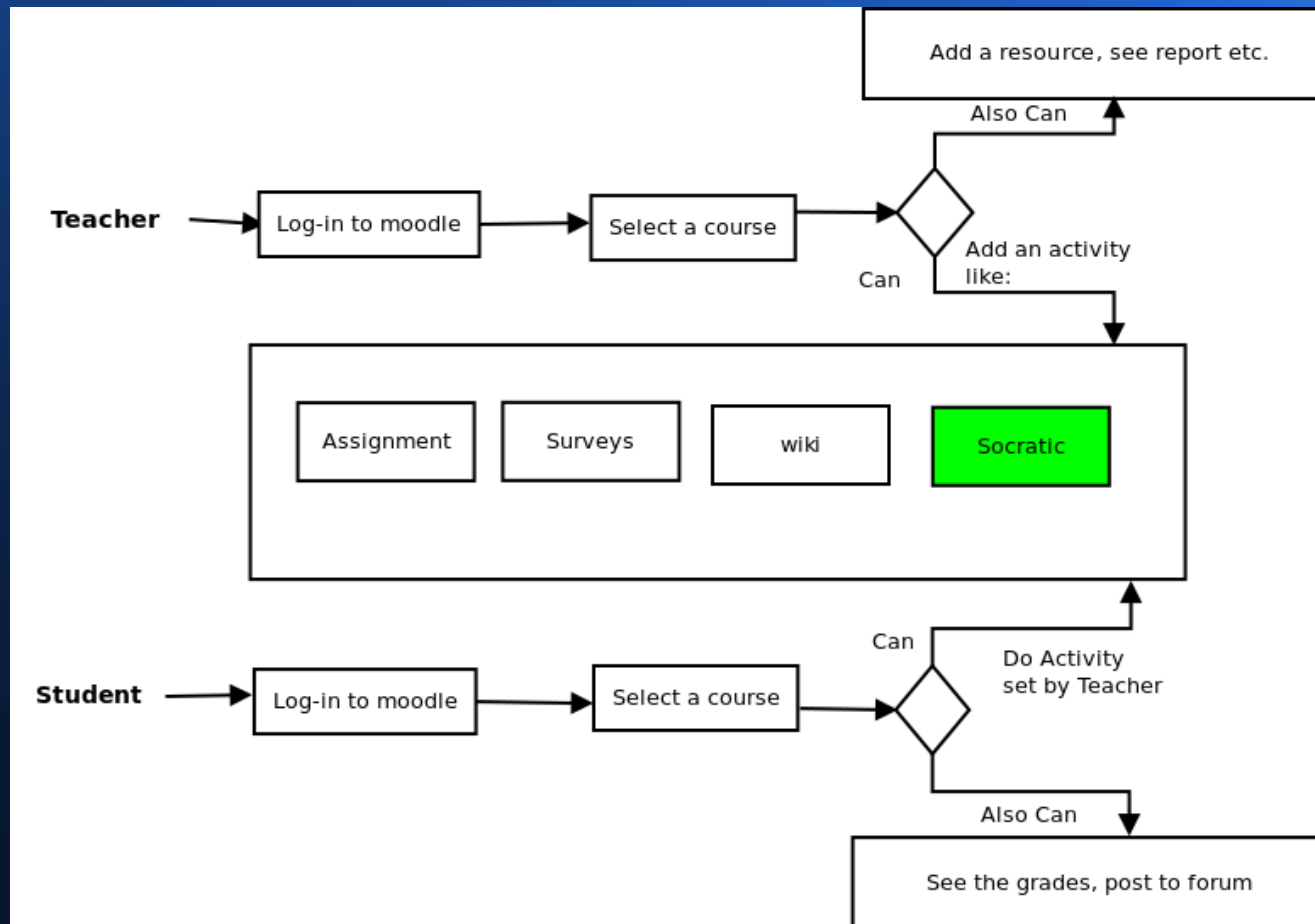
Principles used in Socratic questioning [3]

- Generate different examples to students
- Recall knowledge the student already has
- Determine students belief
- Ask for prediction
- Present counterexamples

Principles used in Socratic questioning [3]

- Testing of hypothesis formed by student.
- Entrap the student and lead him/her to contradiction. When he/she has not identified all the relevant factor.
- Question further to elaborate the concept.
- Help the student in establishing new rules.
- Ask the student to apply new rules.

Moodle Integration



Moodle Integration

The screenshot shows a Moodle course page titled "data structure" in a Mozilla Firefox browser. The browser's address bar shows the URL `localhost/moodle/course/view.php?id=2¬ifieditingon=1`. The course page includes a navigation sidebar on the left with various settings and administration options. A central calendar view displays dates from 20 September to 31 October. A modal dialog box titled "Add an activity or resource" is open in the foreground, listing various activities and resources. The "socratic" activity is selected, and a preview text is visible: "Use the socratic module for... | The socratic module allows...". The dialog box has "Add" and "Cancel" buttons at the bottom. The background page shows a search bar with the text "chouhan" and navigation controls like "Previous", "Next", "Highlight all", and "Match case".

Course: data structure - Mozilla Firefox

localhost/moodle/course/view.php?id=2¬ifieditingon=1

data structure

Home ▶ My courses ▶ ds

Settings

- Course administration
 - Turn editing off
 - Activity chooser off
 - Edit settings
 - Users
 - Unenrol me from ds
 - Filters
 - Grades
 - Backup
 - Restore
 - Import
 - Publish
 - Reset
 - Question bank
- Switch role to...
- My profile settings
- Site administration

20 September - 26 September

27 September - 3 October

4 October - 10 October

11 October - 17 October

18 October - 24 October

25 October - 31 October

Add a block

Add...

Find: chouhan

Previous Next Highlight all Match case

ACTIVITIES

- Assignment
- Chat
- Choice
- chouhan
- Database
- External Tool
- Forum
- Glossary
- Lesson
- newmodule
- newplug
- Quiz
- SCORM package
- socratic
- Survey
- Wiki
- Workshop

RESOURCES

- Book
- File
- Folder
- IMS content package

Use the socratic module for... | The socratic module allows...

Add Cancel

You are logged in as Admin User (Logout)

Turn editing off

Search forums

Go

Advanced search ?

Latest news

Add a new topic...

25 Sep, 12:58
Admin User
ds more...
Older topics ...

Upcoming events

There are no upcoming events

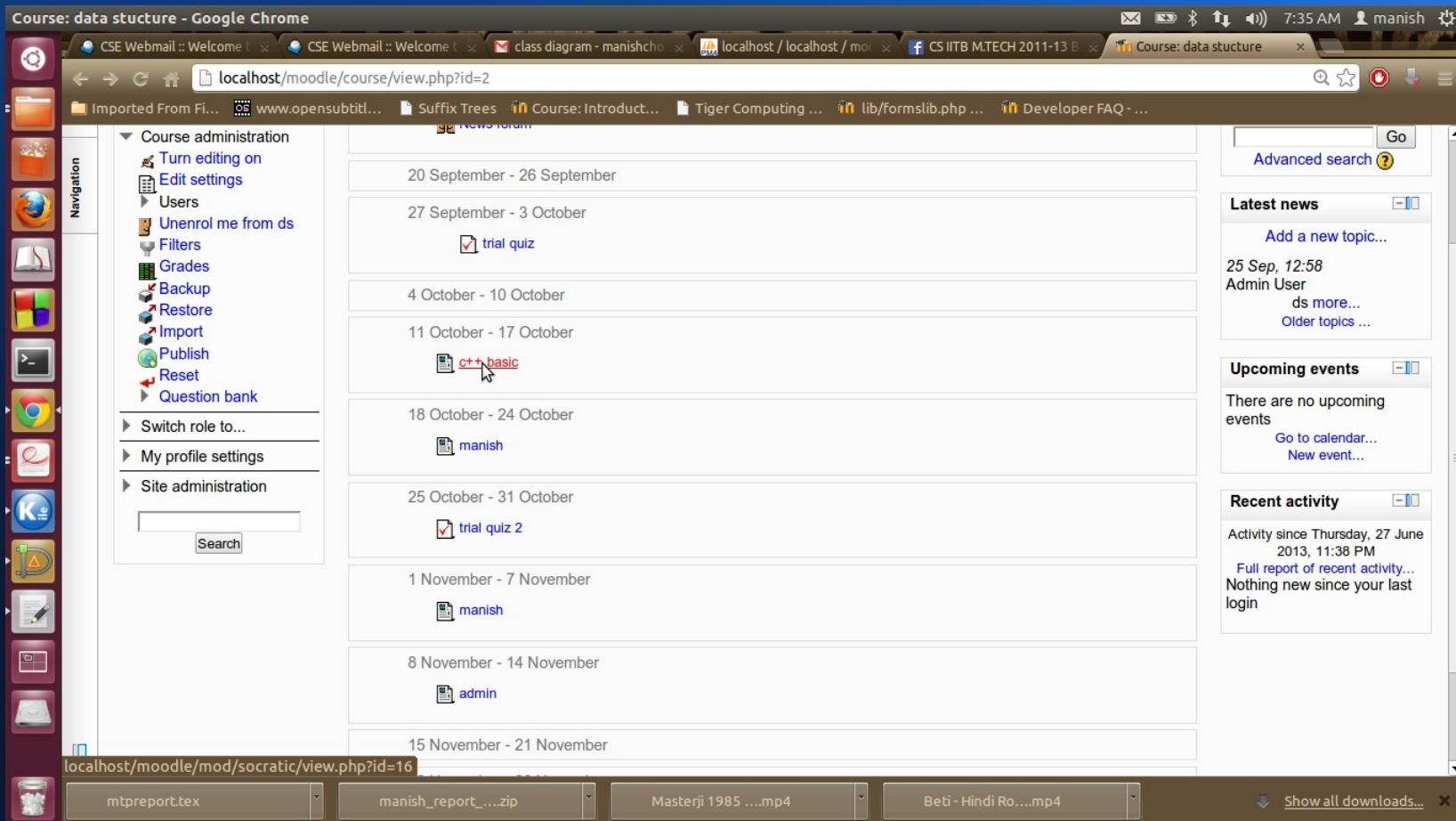
Go to calendar...
New event...

Recent activity

Activity since Sunday, 21 October 2012, 11:30 AM
Full report of recent activity...
Nothing new since your last login

Functional Requirement

1. User will select Socratic activity on course home page.



The screenshot shows a web browser window displaying a Moodle course page. The browser's address bar shows the URL `localhost/moodle/course/view.php?id=2`. The page content is organized into several sections:

- Course administration:** A sidebar menu on the left containing options like "Turn editing on", "Edit settings", "Users", "Unenrol me from ds", "Filters", "Grades", "Backup", "Restore", "Import", "Publish", "Reset", and "Question bank".
- Activity List:** A central table listing activities with columns for dates and activity names. The activities listed are:
 - 20 September - 26 September
 - 27 September - 3 October: trial quiz
 - 4 October - 10 October
 - 11 October - 17 October: **ct+basic** (highlighted with a mouse cursor)
 - 18 October - 24 October: manish
 - 25 October - 31 October: trial quiz 2
 - 1 November - 7 November: manish
 - 8 November - 14 November: admin
 - 15 November - 21 November
- Search and News:** A search bar at the top right with a "Go" button and an "Advanced search" link. Below it is a "Latest news" section with a link to "Add a new topic..." and a news item dated "25 Sep, 12:58" by "Admin User".
- Upcoming events:** A section stating "There are no upcoming events" with links to "Go to calendar..." and "New event...".
- Recent activity:** A section titled "Recent activity" showing "Activity since Thursday, 27 June 2013, 11:38 PM" and a link to "Full report of recent activity...".
- Downloads:** A bottom bar showing a list of downloaded files: `mtpreport.tex`, `manish_report_....zip`, `Masterji 1985mp4`, and `Beti - Hindi Ro....mp4`, with a "Show all downloads..." link.

Functional Requirement

2. System will prompt teacher to select one option either add question or edit question.

The screenshot shows a web browser window with the URL `localhost/moodle/mod/socratic/view.php?id=16`. The page title is "data structure" and the user is logged in as "Admin User (Logout)". The breadcrumb trail is "Home > My courses > ds > 11 October - 17 October > c++ basic". The main content area is titled "socratic" and contains two buttons: "Add question" and "edit question". The left sidebar shows a "Settings" menu with options for "socratic administration", "Course administration", "Switch role to...", "My profile settings", and "Site administration". The bottom of the browser window shows a download bar with several files.

Functional Requirement

3. If user will select add question then If previous Concept is completed.

The screenshot shows a web browser window with the URL `localhost/moodle/mod/socratic/mod_firstQuestion.php?id=16`. The page title is "Start a new concept". On the left, there is a "Settings" sidebar with a tree view containing "socratic administration" (with sub-items: Edit settings, Locally assigned roles, Permissions, Check permissions, Filters, Logs), "Course administration", "Switch role to...", "My profile settings", and "Site administration". A search box is located below the sidebar.

The main content area contains the following form fields:

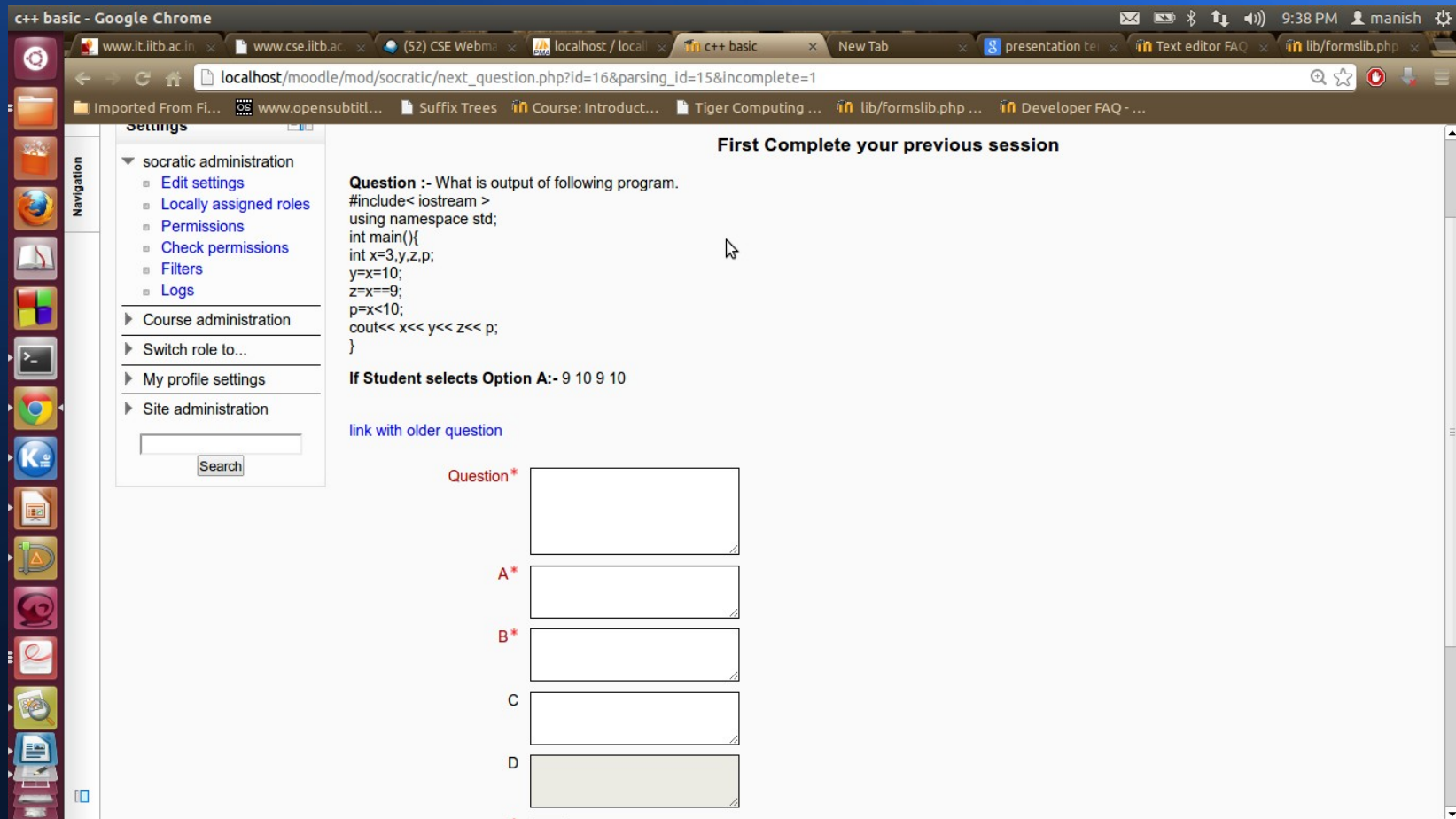
- Concept description***: A text input field.
- Question***: A text input field.
- A***: A text input field.
- B***: A text input field.
- C**: A text input field.
- D**: A text input field.
- correct***: A dropdown menu currently showing "A".

At the bottom of the form are "Save changes" and "Cancel" buttons. A red asterisk at the bottom right indicates "There are required fields in this form marked *".

The browser's taskbar at the bottom shows several open files: `mtpreport.tex`, `manish_report_...zip`, `Masterji 1985 ...mp4`, and `Beti - Hindi Ro...mp4`. The system tray shows the time as 7:39 AM and the user as manish.

Functional Requirement

4. If previous concept is not completed.



The screenshot shows a web browser window displaying a Moodle Socratic question page. The browser's address bar shows the URL: `localhost/moodle/mod/socratic/next_question.php?id=16&parsing_id=15&incomplete=1`. The page title is "First Complete your previous session".

On the left side, there is a navigation menu with the following items:

- socratic administration
 - Edit settings
 - Locally assigned roles
 - Permissions
 - Check permissions
 - Filters
 - Logs
- Course administration
- Switch role to...
- My profile settings
- Site administration

The main content area displays the following text:

Question :- What is output of following program.

```
#include< iostream >
using namespace std;
int main(){
int x=3,y,z,p;
y=x+10;
z=x+9;
p=x+10;
cout<< x<< y<< z<< p;
}
```

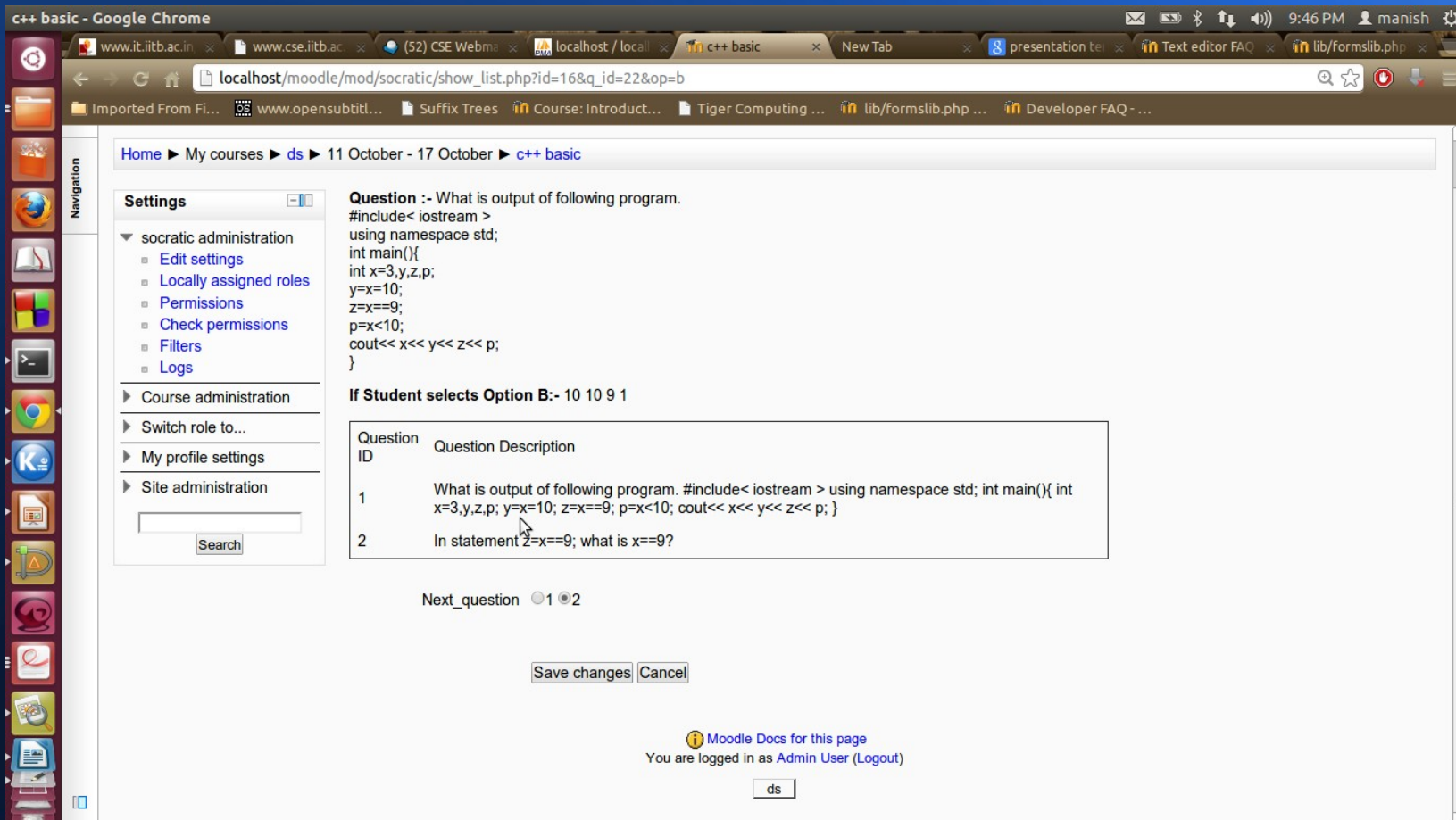
If Student selects Option A:- 9 10 9 10

Below the question, there is a "link with older question" link and a "Question*" label. The question is followed by four multiple-choice options, each with a text input field:

- A*
- B*
- C
- D

Functional Requirement

5. Teacher can link question with older question



The screenshot shows a Moodle Socratic question editor interface. The main content area displays a question with a C++ code snippet and a table of questions. The table has two columns: 'Question ID' and 'Question Description'. The first row (ID 1) contains the question text and the code snippet. The second row (ID 2) contains the text 'In statement $z=x=9$; what is $x=9$?'. Below the table, there is a 'Next question' section with radio buttons for selecting between question 1 and question 2. The 'Next question' section shows question 2 selected. At the bottom, there are 'Save changes' and 'Cancel' buttons, a link to Moodle Docs, and a user login status 'You are logged in as Admin User (Logout)'. A 'ds' button is also visible at the bottom center.

Question :- What is output of following program.

```
#include< iostream >
using namespace std;
int main(){
int x=3,y,z,p;
y=x=10;
z=x=9;
p=x<10;
cout<< x<< y<< z<< p;
}
```

If Student selects Option B:- 10 10 9 1

Question ID	Question Description
1	What is output of following program. #include< iostream > using namespace std; int main(){ int x=3,y,z,p; y=x=10; z=x=9; p=x<10; cout<< x<< y<< z<< p; }
2	In statement $z=x=9$; what is $x=9$?

Next_question 1 2

Save changes Cancel

[Moodle Docs for this page](#)
You are logged in as Admin User (Logout)

ds

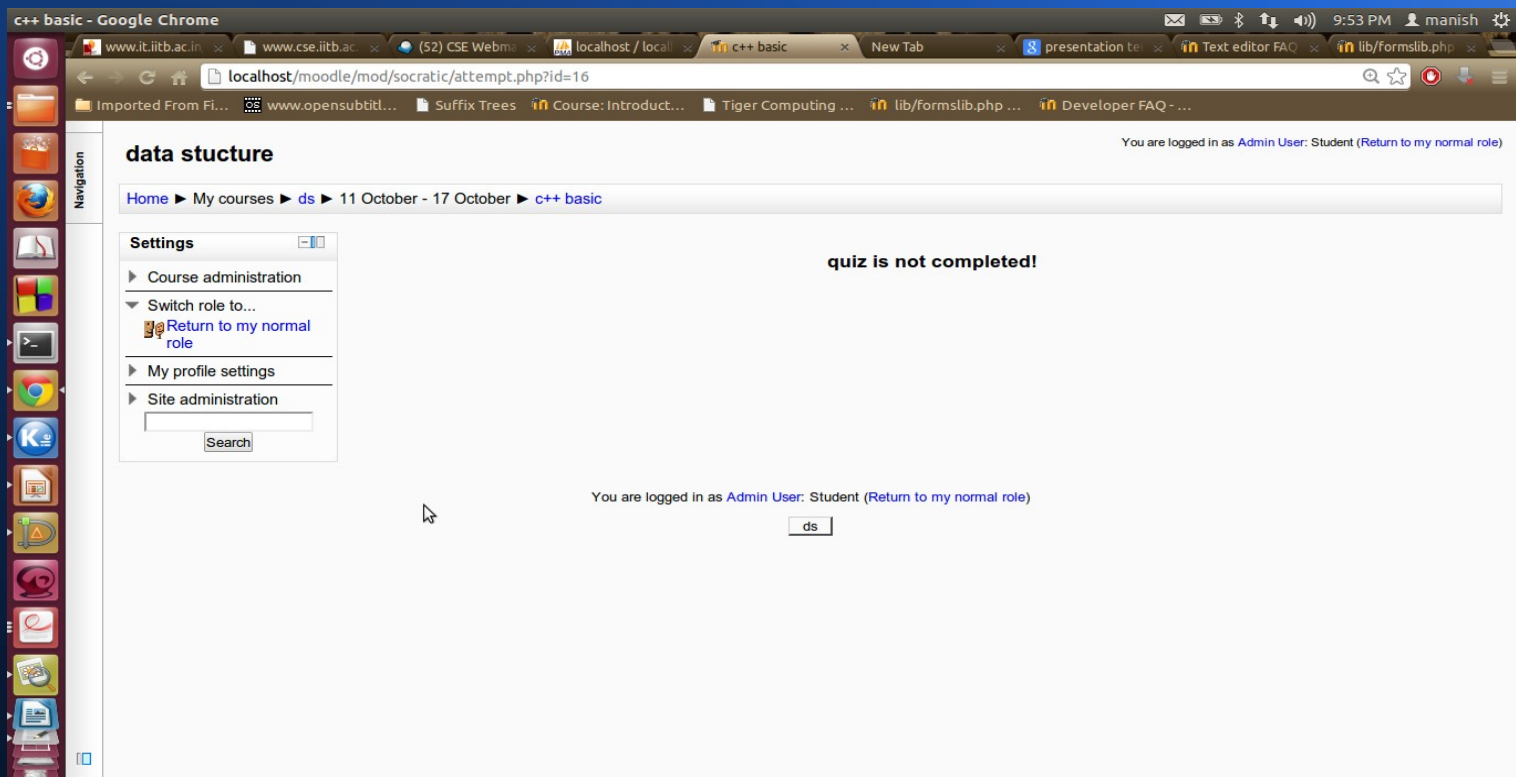
Functional Requirement

If a student click on activity then

The screenshot shows a web browser window with the URL `localhost/moodle/mod/socratic/view.php?id=16`. The page title is "data structure" and the user is logged in as "Admin User: Student (Return to my normal role)". The breadcrumb trail is "Home > My courses > ds > 11 October - 17 October > c++ basic". On the left, there is a "Settings" sidebar with options for "Course administration", "Switch role to..." (with a "Return to my normal role" link), "My profile settings", and "Site administration". The main content area is titled "socratic" and contains an "Attempt quiz" button. At the bottom, there is a "ds" button and a "Search" input field.

Functional Requirement

If there is an incomplete Concept in the quiz.



The screenshot shows a web browser window with the URL `localhost/moodle/mod/socratic/attempt.php?id=16`. The page title is "data structure" and it indicates the user is logged in as "Admin User: Student (Return to my normal role)". The breadcrumb trail is "Home > My courses > ds > 11 October - 17 October > c++ basic". On the left, there is a "Settings" sidebar with sections for "Course administration", "Switch role to..." (with a "Return to my normal role" link), "My profile settings", and "Site administration" (with a search box). The main content area displays the message "quiz is not completed!". At the bottom of the page, it says "You are logged in as Admin User: Student (Return to my normal role)" and there is a small button labeled "ds".

Functional Requirement

If there is no incomplete Concept in the quiz.

The screenshot shows a web browser window with the URL `localhost/moodle/mod/socratic/attempt.php?id=16`. The page title is "data structure". The user is logged in as "Admin User: Student". The breadcrumb trail is "Home > My courses > ds > 11 October - 17 October > c++ basic".

Settings

- Course administration
- Switch role to...
 - [Return to my normal role](#)
- My profile settings
- Site administration

Question : What would happen when this program is executed?

```
using namespace std;
int main(){
int a = 8,b;
cout<< a ;
}
```

8

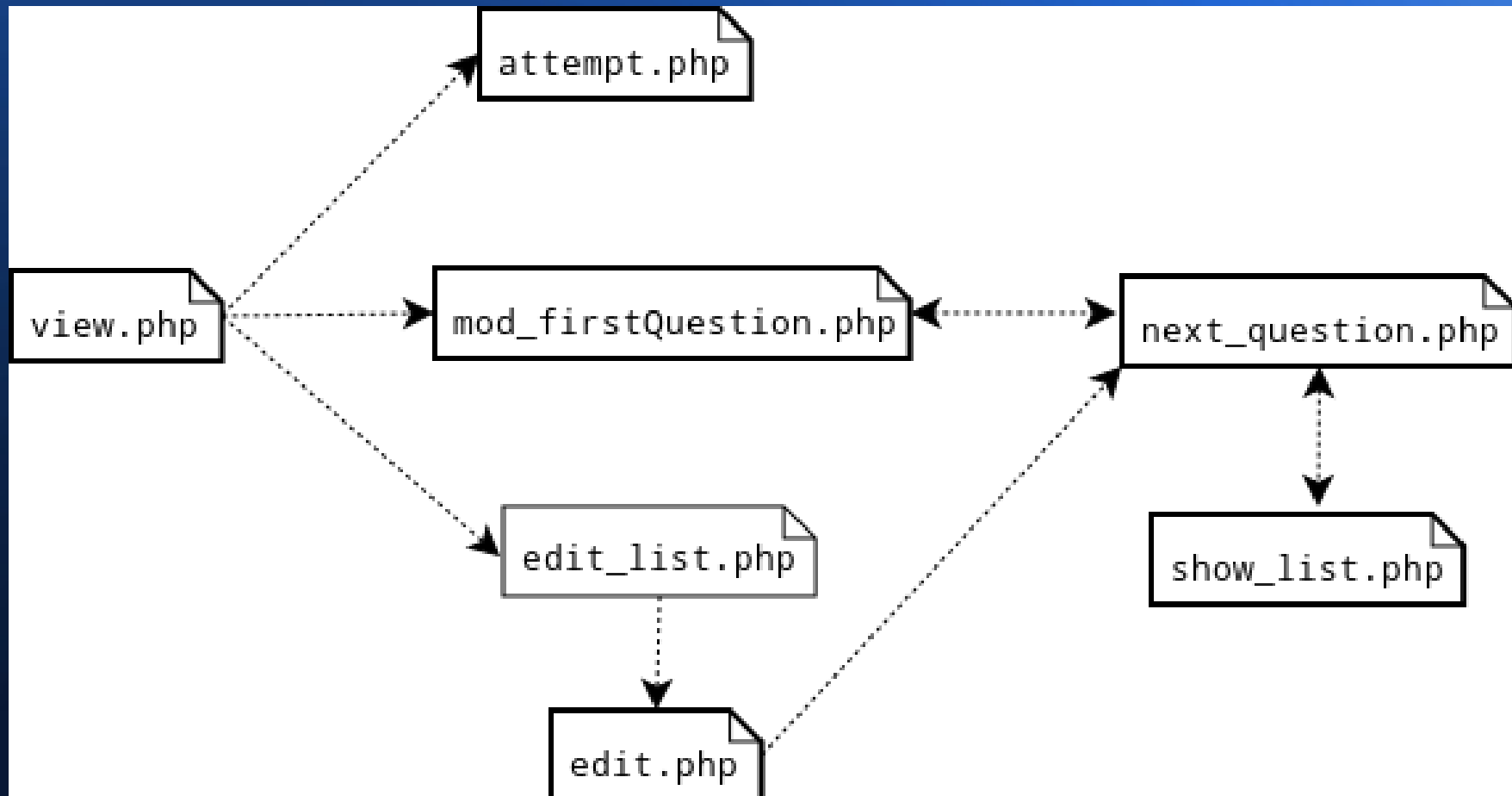
compiler error

You are logged in as Admin User: Student (Return to my normal role)

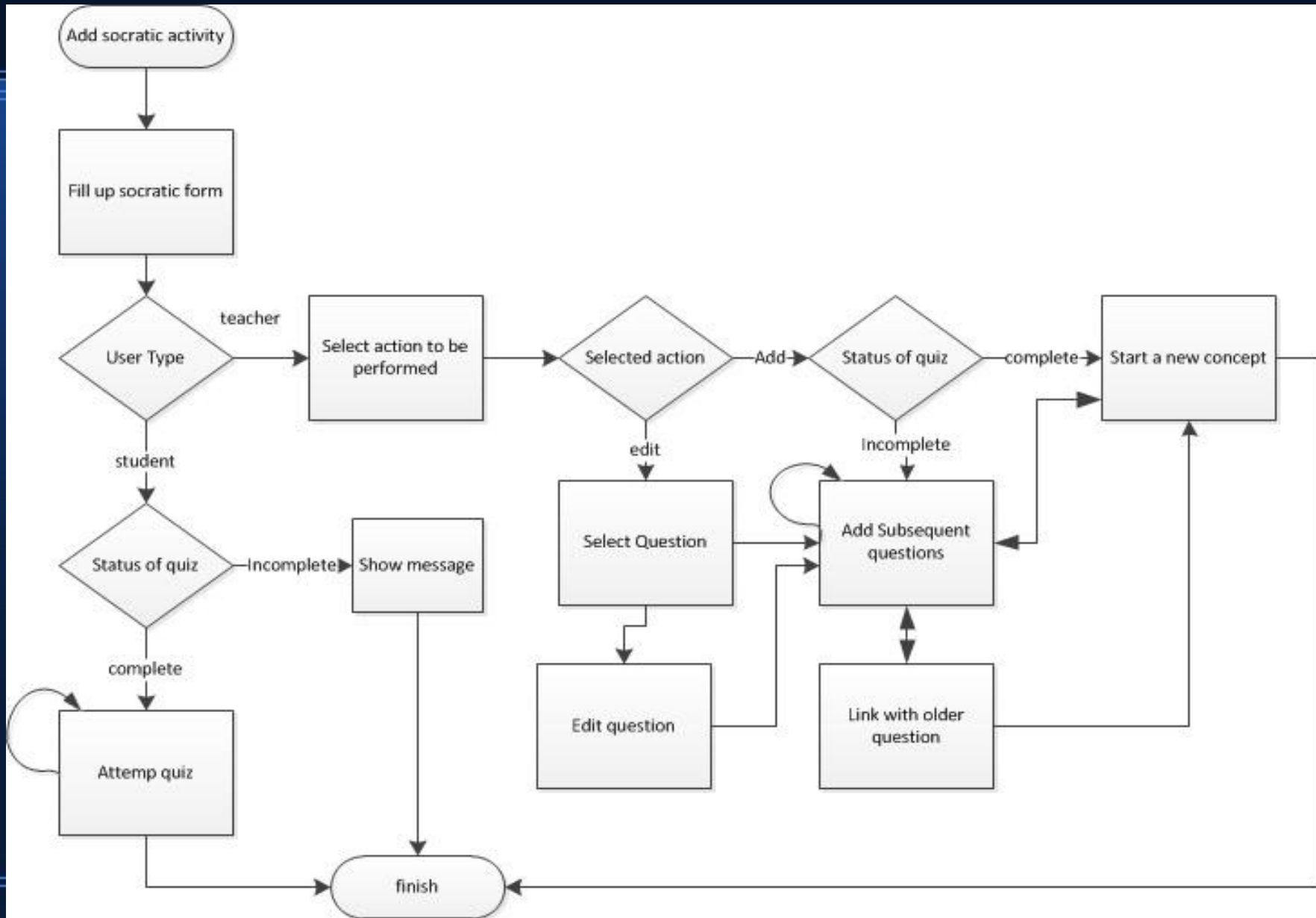
Implementation

File Name	Purpose
view.php	Home page of Socratic activity
mod_firstQuestion.php	For teacher to enter First question of a new concept.
next_question.php	For teacher to enter a connecting question for previously added question.
show_list.php	For teacher to connect an option of current question to a previous question from the list of questions.
edit_list.php	For teacher to select a question for editing.
edit.php	For teacher to edit the selected question
attempt.php	For student to attempt the Socratic quiz
version.php	This file contains the current version number of the module.
lib.php	Contains functions which are needed to integrate the module with Moodle.
db/access.php	Use to Define capabilities to restrict access of the user
db/upgrade.php	Use to upgrade database in moodle
db/install.xml	Define Database tables in xml format

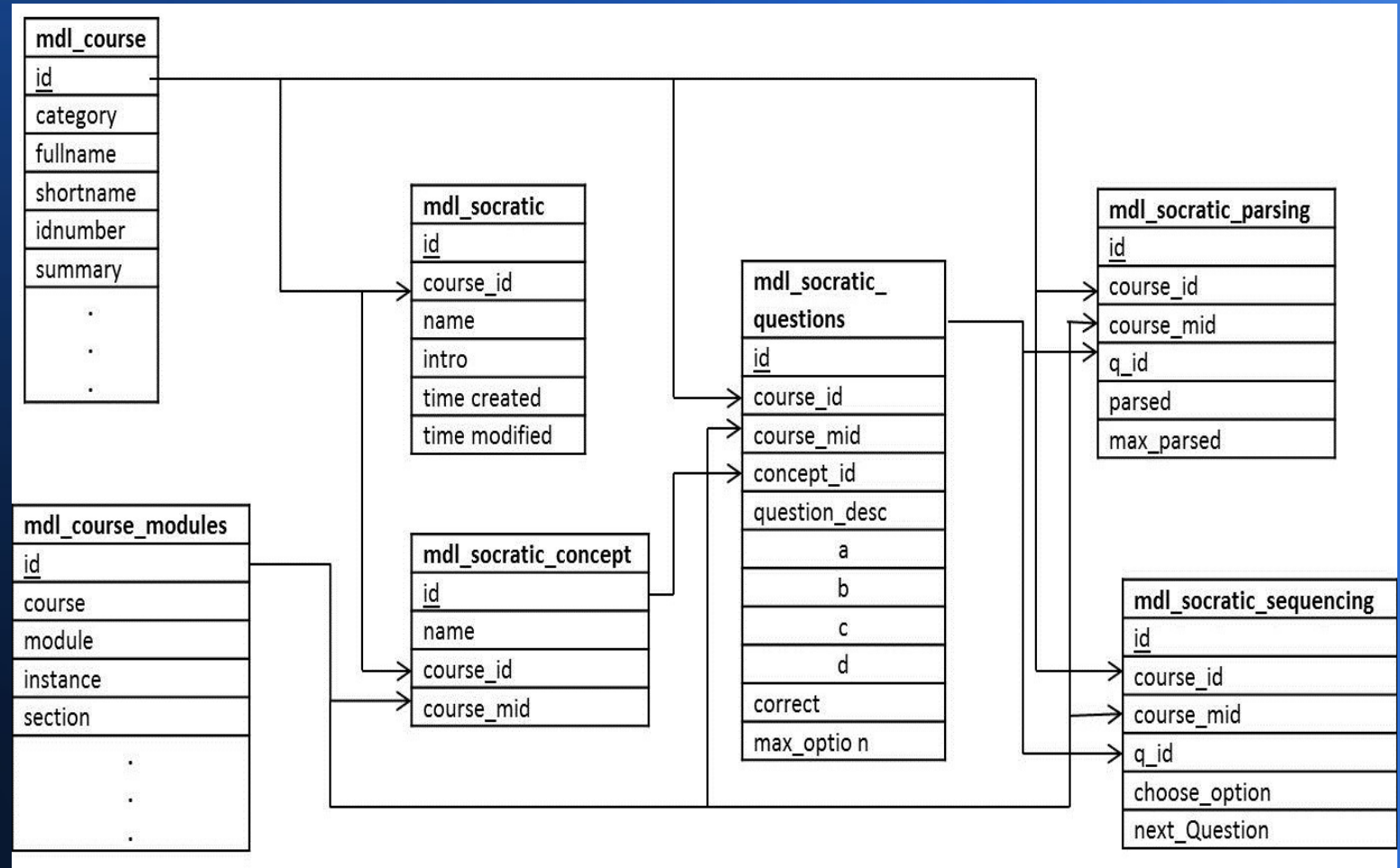
Component Diagram



Process Flow



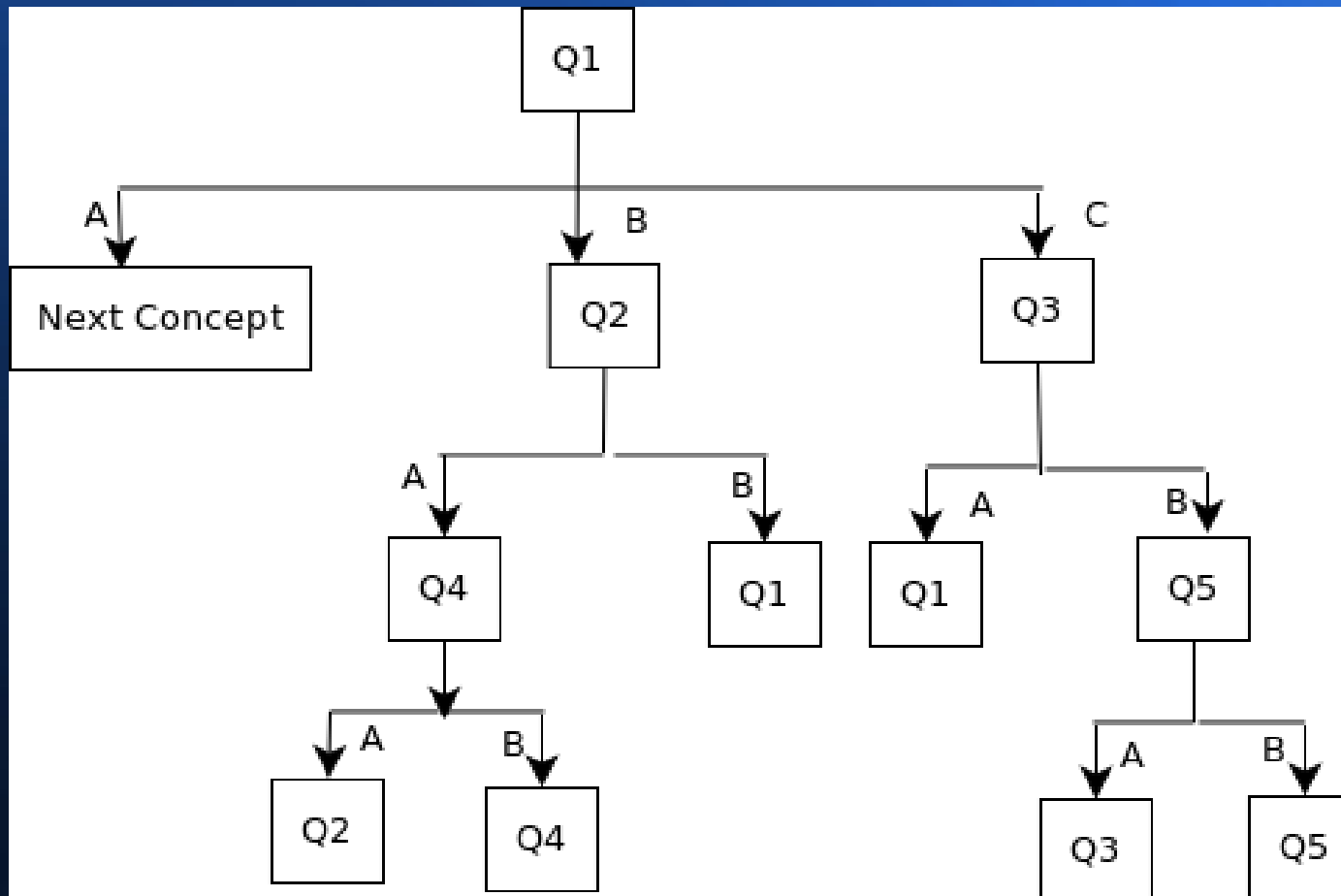
Database Structure



Functional logic

- Socratic quiz will be created concept wise.
- For each concept there will be a series of questions which forms a cycle.
- Every option of a question will be connected to some question.
- Correct option of first question will automatically connect to first question of next concept.

Cycle of questions



Status of concept in quiz?

```
SELECT *
FROM mdl_socratic_parsing`
LIMIT 0 , 30
```

Profiling [Inline]

Show : 30 row(s) starting from row # 0 in horizontal mode and

Sort by key: None

+ Options

	id	course_id	course_mid	q_id	parsed	max_parsed
<input type="checkbox"/> Edit Inline Edit Copy Delete	9	2	21	19	1	2
<input type="checkbox"/> Edit Inline Edit Copy Delete	12	2	22	21	0	3
<input type="checkbox"/> Edit Inline Edit Copy Delete	14	2	21	15	0	3

- When add a new question - Add an entry in socratic_parsing table for that question.
- When all option are parsed – delete this entry from socratic_parsing table.
- If there are no entry for a course module {Socratic activity} then there is no incomplete in that Socratic quiz activity.

Edit Question

The screenshot shows a web browser window with the URL `localhost/moodle/mod/socratic/edit_list.php?id=16`. The page title is "data structure" and the user is logged in as "Admin User". The breadcrumb trail is "Home > My courses > ds > 11 October - 17 October > c++ basic".

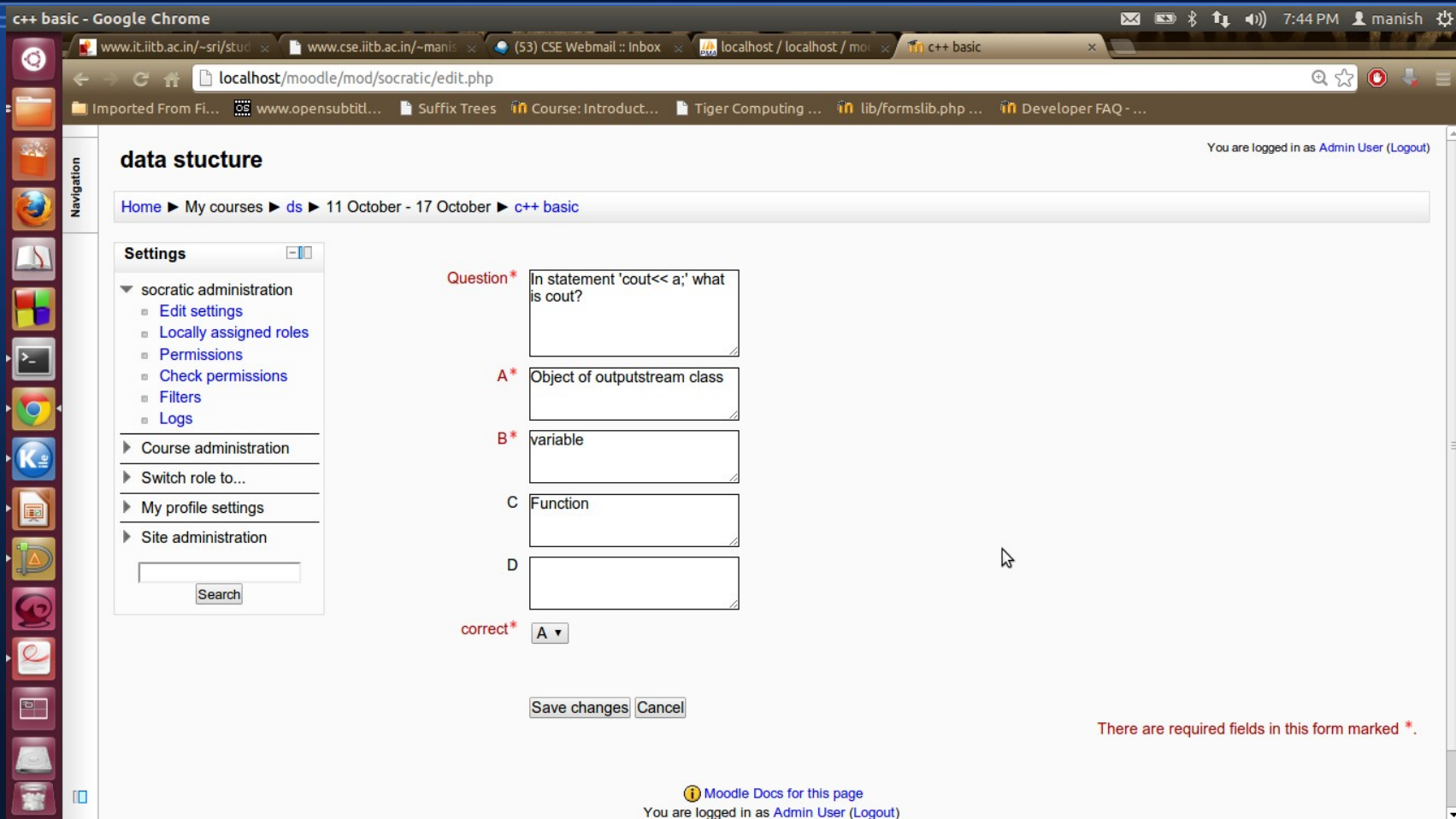
On the left, there is a "Settings" sidebar with a search box and a list of categories: "socratic administration", "Course administration", "Switch role to...", "My profile settings", and "Site administration". Under "socratic administration", there are links for "Edit settings", "Locally assigned roles", "Permissions", "Check permissions", "Filters", and "Logs".

The main content area is titled "Select Question you want to edit" and contains a table with the following data:

select	Question ID	Question Description
<input type="radio"/>	1	What would happen when this program is executed? using namespace std; int main(){ int a = 8,b; cout<< a ; }
<input checked="" type="radio"/>	2	In statement 'cout<< a;' what is cout?
<input type="radio"/>	3	Does our program contain this outputstream class?
<input type="radio"/>	4	Q4 Where is OutputStream class defined?
<input type="radio"/>	5	So before using cout should we include this library header file?

Below the table is a "submit" button. At the bottom of the page, there is a "Moodle Docs for this page" link, a "You are logged in as Admin User (Logout)" message, and a "ds" button.

Edit Question



The screenshot shows a web browser window with the Moodle Socratic question editor. The browser title is "c++ basic - Google Chrome". The address bar shows "localhost/moodle/mod/socratic/edit.php". The page title is "data structure". The user is logged in as "Admin User".

The navigation breadcrumb is: Home ► My courses ► ds ► 11 October - 17 October ► c++ basic

The left sidebar contains a "Settings" menu with the following items:

- socratic administration
 - Edit settings
 - Locally assigned roles
 - Permissions
 - Check permissions
 - Filters
 - Logs
- Course administration
- Switch role to...
- My profile settings
- Site administration

The main content area shows a question editor:

Question * In statement 'cout<< a;' what is cout?

A * Object of ostream class

B * variable

C Function

D

correct * A ▼

Buttons: Save changes, Cancel

Message: There are required fields in this form marked *.

Footer: Moodle Docs for this page, You are logged in as Admin User (Logout)

Edit Question

- When user clicks on save changes button all entry in sequencing table and parsing Table for this question_id will be deleted.
- New entry will be added to the parsing table with field parsed =0.
- Question entry in question table will be updated.

The screenshot displays a database management interface. At the top, a SQL query is shown: `SELECT * FROM mdl_socratic_sequencing LIMIT 0, 30`. Below the query, there are controls for displaying the results: "Show : 30 row(s) starting from row # 0 in horizontal mode and repeat headers after 100". A "Sort by key:" dropdown is set to "None". Below these controls, a table of data is displayed with columns: id, course_id, course_mid, q_id, chosen_option, and next_question. Each row includes interactive icons for Edit, Inline Edit, Copy, and Delete.

	id	course_id	course_mid	q_id	chosen_option	next_question
<input type="checkbox"/> Edit Inline Edit Copy Delete	26	2	16	11	a	11
<input type="checkbox"/> Edit Inline Edit Copy Delete	27	2	16	11	b	12
<input type="checkbox"/> Edit Inline Edit Copy Delete	28	2	16	12	a	13
<input type="checkbox"/> Edit Inline Edit Copy Delete	29	2	16	12	b	12
<input type="checkbox"/> Edit Inline Edit Copy Delete	30	2	16	12	c	12
<input type="checkbox"/> Edit Inline Edit Copy Delete	31	2	16	13	a	9
<input type="checkbox"/> Edit Inline Edit Copy Delete	32	2	16	13	b	10
<input type="checkbox"/> Edit Inline Edit Copy Delete	33	2	16	10	a	11
<input type="checkbox"/> Edit Inline Edit Copy Delete	34	2	16	10	b	10
<input type="checkbox"/> Edit Inline Edit Copy Delete	35	2	16	10	c	10
<input type="checkbox"/> Edit Inline Edit Copy Delete	42	2	21	14	a	19

Limitation & Future work

- At this time this plugin is limited to teach any subject with the help of multiple choice questions (MCQs) only.
- Teacher can't enter rich text content in question and option fields.
- Use of editor element instead of text area in moodle form. It allows user to enter rich text content.
- GUI may be improved.

References

1. <http://www.Socraticmethod.net>.
2. vikash kumar Development of Intelligent Tutoring System Framework: Using Socratic strategy
3. K. E. Chang, M. L. Lin, and S. W. Chen, Application of socratic dialogue on corrective learning of subtraction, Comput. Educ., vol. 31, no.1, pp. 5568, Aug. 1998.
4. <http://docs.moodle.org>.
5. <http://docs.moodle.org/23/en/Pedagogy>.

Thank You