Palestine Polytechnic University

College of Administrative Sciences and Informatics

Department of Information Technology

Plagiarism Detection System

Aya Ahmad Abu Sabha

Nehad Ameen Hroub

Sabreen Faris Smeirat

Supervisor

Eng. Hani Salah

A final project submitted in partial fulfillment of the requirements

for the bachelor degree in information technology

June 2010

Acknowledgment

The team members advance deep thanks to their dear supervisor
Eng. Hani Salah

The team advances deep thanks to the Friends of Fawzi Kawash IT Center of Excellence (FFKITCE) for their support and help

To our dear teachers, friends, and everybody helped us… we can only say … **Thank You …**

Project team

Dedication

To our parents for their support and encouragement all the time

To the supervisor of the project Eng. Hani Salah

To all our teachers and friends.

Project team

الملخص

الانتحال في واجبات الطلبة هي مشكلة ذات نطاق واسع ومتزايدة في المراحل الأكاديمية. أن عملية الكشف عن الانتحال بواسطة الإنسان هي عملية بطيئة وغير موثوقة. لذلك سيتم في هذا المشروع بناء نظام محوسب يعتمد على الويب لكشف حالات الانتحال حتى يتمكن المدرسون في الجامعات من عمل كشف للانتحال بطريقة أسرع وأكثر دقة وموثوقية.

يوجد العديد من الخوارزميات لكشف الانتحال, لذلك تم دراسة هذا الخوارزميات, واختيار أحدها كأساس لتصميم وتنفيذ هذا النظام لمقارنة الواجبات التي يسلمها الطلبة على موقع التعليم الالكتروني المستخدم في جامعة بوليتكنك فلسطين, وإخراج تقارير تفيد بوجود حالات الانتحال ,ولقد تم استخدام دورة حياة تطور الانظمة (SDLC) كمنهجية معتمدة لبناء النظام.

وسيتم تطبيق هذا النظام على موقع التعليم الالكتروني في جامعة بوليتكنك فلسطين (موودل), باستخدام لغة PHP المستخدمة في برمجة نظام موودل.

مع العلم بوجود أنظمة تقوم بنفس عمل نظام هذا المشروع, إلا أن هذه الأنظمة (البرمجيات) غير مجانية وغير مفتوحة المصدر, بعكس النظام المقدم في هذا التقرير والذي سيوزع بشكل مفتوح المصدر.

Abstract

The plagiarism in student assignments is a widespread and growing problem in the academic process. The traditional manual detection of this kind of plagiarism by human is difficult, not accurate, and time consuming process. This project aims to create an online web-based plagiarism-detection system that can help university teachers to make a better judgment for student's work.

There is currently a number of existing detection algorithms. Some of them have been studied, analyzed, and compared. Based on this analysis and comparison, the most suitable one for our system (namely: AC) has been chosen for the implementation phase. Our system will reuse the selected algorithm, and integrate it to the E-learning platform that is used at PPU (namely: Moodle). The algorithm has been reprogrammed by PHP (the language used in Moodle programming), in this project we will use System Development Life Cycle (SDLC) as a methodology to implement this system.

Given that, there are already some existing systems (software) for plagiarism detection, but they are all not free and not open-source. On the other hand, the system that is presented in this report is free to use, integrateable with Moodle, and open-source.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter One

Introduction

1.1 Problem Statement

Plagiarism can be defined as "*the use or close imitation of the language and thoughts of another author and the representation of them as one's own original work*" [1].

Many students make (intentionally or unintentionally) some type of cheating and plagiarism in their assignments. Usually it is difficult for teachers to detect plagiarism in student assignments by hand. The detection process becomes easier, faster and more efficient if it's performed automatically (i.e. via a computerized system).

The plagiarism detection *"is the processes of locating instance of plagiarism within a document wither its text or code"* [2]. Detection can be either manual or computer-assisted. Manual detection requires effort, and it is impractical in cases where many documents to be compared.

Modular Object-Oriented Dynamic Learning Environment (Moodle) is an open source Course Management System (CMS) , it's used by educators as a dynamic web site for their students; it's best tool to manage learning. Moodle can support large deployments and hundreds of thousands of students, so it can be used for schools. It can be used to conduct fully online courses while others used it to augment face-to-face courses. Some teachers use Moodle to deliver material to students, assignments and quizzes, that makes learning more richly and collaborative.

1.2 Project Objective

The main goal of the project is to create an online plagiarism detection system that can be used by teachers at Palestine Polytechnic University (PPU) to detect cheating and plagiarism cases in student submitted assignments. The system should be integrated with the e-learning platform that is currently used at PPU (namely: Moodle).

1.3 Project Goals

The project objective mentioned in Section 1.2 can be achieved by specifying the following goals:

1. Studying, analyzing, and comparing existing open-source plagiarism detection software and algorithms.

2. Picking one of the analyzed algorithms to be used in this system, or designing a new one from the scratch. This goal includes coding the algorithm using PHP (the programming language used in Moodle).
3. Integrating the plagiarism-detection code in the existing PPU e-learning platform (Moodle).

Based on the goals stated above, we shall precede breaking down them into concrete activities, functions, and deliverables as follow:

1. **Studying, analyzing, and comparing the existing open-source plagiarism detection software and their algorithms.**
    a. Searching for related algorithms and open- source software.
    b. Studying and analyzing these algorithms.
    c. Comparing the algorithms and identifying their advantages and disadvantages.

2. **Picking one of the analyzed algorithms to be used in this system, or designing a new one from the scratch.**
    a. Based on the comparison among the analyzed algorithms, the best (most suitable / adaptable one to the system requirements) will be picked to be implemented in the system.

3. **Integrating the selected algorithm in the PPU e-learning platform (Moodle).**
    a. Learn the language that is used on construction of the Moodle (namely: PHP).
    b. Reprogramming the plagiarism-detection code using PHP, and adding it as a new feature to Moodle.

1.4 Methodology

The project team will use the traditional method of software engineering, which called System Development Life Cycle (SDLC), in the analysis, and development of the system.

We will start by studying, analyzing existing plagiarism system and algorithms, and comparing between them to choice the best and most suitable one between them to be uses on the project.

1.5 System Requirements

This section lists both functional and non-functional requirements of the system. Further details about each requirement will be discussed in Chapter 3.

1.5.1 Functional Requirements

a. Enable teachers to detect plagiarism and cheating in student submitted assignments.

b. Viewing visually aided cheating (similarity) reports.

1.5.2 Non-Functional Requirements

a. Compatibility.

b. Ease to use.

1.6 Time Schedule / Gantt Chart

To discuss the time schedule of the project, it is best to refer to the Gantt charts below in Table (1) for the first part of the project and Table (2) for the second part of the project where the key activities are stated and the time period for their completion is depicted.

For clearing up, the following is a brief description of the key activities stated in Table (1) (starting October 1st - December 20th, 2009-12-20)

- **Analysis:** establishes a high-level view of the project and its goal.
- **Requirement specification:** presents the functional and non-functional requirements.
- **Design:** describes desired features and operation in details.

Table (1): Gantt Chart 1 (First Semester: October 2009 - December 2009)

Key Activities	October	November	December
Analysis	🟦		
Requirement specification		🟦	
Design(Draft)			🟦
Documentation	🟦	🟦	🟦

<u>Important dates and milestones for 1st semester:</u>

- **October 1st:** Project plan.
- **October 20th:** Project outline.
- **November 30th:** literature review and Background.
- **December 10th:** Requirement specification.
- **December 20th:** Design.
- **December 24th:** Final documentation.

Below is a brief description of key a activities stated in Table (2) (starting February 1st to may 15th, 2010):

- **Programming:** coding by PHP.
- **Installation:** presents more technical details and show the installation environment.
- **Testing:** put the system under testing to ensure that it meets the design and functional requirements.

Table (2): Gantt Chart 2 (Second Semester: February 2010 –May 2010)

Key Activities	February	March	April	May
Design (Final)	▨			
Programming	▨	▨		
Installation		▨	▨	
Testing				▨
Documentation	▨	▨	▨	▨

<u>Important dates and milestones for 2nd semester:</u>

- **February 20th:** Design.
- **March 30th:** Programming.
- **May 5th:** Testing.
- **May 25th:** Final documentation.

1.7 The Risks

During the different phases of this project, some risks may appear and cause delaying, threaten the progress, or even affecting the outcomes of the project. Table (3) summarizes the most important expected risk events, access degree of probability (P) that the risk event happens (represented from 1-5 where five is the most likely), access degree of effect (E) upon project when it happens (represented from 1-5 where five is the worst), and the risk index which equal the product (P × E).

Table (3): Expected Risks

No	Risk event	Probability (P)	Effect(E)	Risk index (P ×E)
1	Member absence	4	2	8
2	Lack of time / schedule conflict between members	5	3	15
3	Equipment malfunction	2	5	10
4	Loss of data	2	5	10
5	Loss of equipment	2	5	10
6	Unexpected results	2	3	6
7	Communications problems between members	2	2	4
8	Installation problems	3	5	15

Table (4) summarizes the expected effect for each event on the project, action(s) could be taken for each event, and who is responsible for this action.

Table (4): Effects and Responsibilities

No	Risk	Effect on the project	Action	Responsibility
1	Member absence	Delaying	Each member May absence	ALL
2	Lack of time / schedule conflict between members	Delaying	Try to solve conflicts as possible	ALL
3	Equipment malfunction	Delaying/ partial or complete failure	Periodic maintenance and testing and preparing the damaged part	ALL
4	Loss of data	Delaying/ partial or complete failure	Backup	ALL
5	Loss of equipment	Delaying/ partial or complete failure	Provide enough security To save the equipment from theft	ALL
6	Unexpected results	Delaying/ partial failure	Perform requirement analysis carefully right from the beginning	ALL

Table (4): Cont.

7	Communications problems between members	Delaying	Perform timely meetings ,good communication	ALL
8	Installation problems	Delaying	Try to solve the problems as possible	ALL

Chapter Two

Background

In this project we will presents background about PHP, Moodel and Existing Cheating Detection Algorithms and Open Source Software.

2.1 PHP

Hypertext **P**reprocessor (**PHP**) "*is a widely-used general-purpose scripting language that is especially suited for web development and can be embedded into html*" [3].

2.1.1 Advantages of PHP

1. PHP is a powerful tool for making dynamic and interactive Web pages.
2. PHP is a server-side scripting language.
3. PHP supports many database management systems (MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC, etc.).
4. PHP is open source software.
5. PHP is free to download and use.

2.1.2 Why PHP

1. PHP runs on different platforms (Windows, Linux, UNIX, etc.).
2. PHP is compatible with almost all web servers used today (Apache, IIS, etc.)
3. PHP is easy to learn and runs efficiently on the server side.
4. PHP is the language used in Moodle programming.

2.2 Moodle

Moodle is *"a software package for producing internet-based courses and web sites. It is an ongoing development project designed to support a social constructionist framework of education"* [4].

2.2.1 Moodle Usage

Moodle is using by a variety of institutions and individuals, including universities, high and primary schools.

2.2.2 Moodle Management

Moodle is managed by an administrator who is identified during the setup (default can be edited during setup). The administrator can use the plug-in themes to customize the site colors, fonts, layout to suit need otherwise plug-in activity module can be added to Moodle installation. Plug-in language packs allow full localization to any language. These can be edited using a built-in web based editor. Currently there are language packs for over 70 languages.

2.2.3 Assignments in Module

- Assignments can be specified with specific date and a maximum grade.
- Students can upload their assignments (using any file format) to the server.
- Late assignments are allowed, but the amount of lateness is shown clearly to the teacher.
- For each particular assignment, the whole class can be assessed (grade and comment) on one page in one form.
- Teacher feedback is appended to the assignment page for each student, and notification is mailed out.
- The teacher can choose to allow resubmission of assignments after grading (for regarding)
- Advanced assignments can allow multiple files to be uploaded.

2.3 Existing Cheating Detection Algorithms and Open Source Software

Many algorithms for plagiarism detection systems have already been developed. This section introduces these algorithms and identifies their features. These algorithms can be roughly classified into two categories:

1. Attribute-counting Systems.
2. Structure-metric Systems.

2.3.1 Attribute-Counting Systems

This is the earliest type of plagiarism detection algorithms. These systems measure the level of similarity among assignments pairs, using four simple program statistics:

- Number of distinct operators.
- Number of distinct operands.
- Total number of operator occurrences, over all distinct types.
- Total number of operand occurrences, over all distinct types

2.3.2 Structure-metric Systems

This type of plagiarism detection algorithms introduces much larger number of metrics and notions of similarity for the resulting feature vector in order to improve performance (based on structure and metric comparison).

These algorithms are usually based on converting the program into a stream of tokens (thus ignoring easily changeable information such as space, line breaks, comments, etc.) and then comparing these token streams to find similarities among them. The most advanced systems in this category (in terms of plagiarism detection performance) are: SIM, MOSS, JPlag, AC and CodeMatch. The following is a brief description for these systems.

2.3.2.1 SIM

Software Similarity Tester (SIM) plagiarism detection system was developing in 1999 by Gitchell and Tran as a system for measuring the similarity between text written in C, Java, Pascal and natural language.

In SIM, each program is first parsed using the lexical analyzer to produce a sequence of integers (tokens), then compares token sequences using a dynamic programming string alignment technique. This technique first assigns each pair of characters in alignment a score. For example, a match scores 1, a mismatch scores -1. The score between two sequences is then defined to be the maximum score among all alignments, and tests similarity between texts written in C, Java, Pascal, and natural language. With this definition, a similarity measure between two sequences is defined as follows:

$$s = 2 * score(s,t)/(score(s,s) + score(t,t))$$

SIM work steps:

1. Read the program files: read the file and store it in sequence.

2. Determine the set of interesting runs: the algorithm determines match between two files.

3. Determine the line numbers of the interesting runs: finds the start and end line number for each chunk.

4. Print the contents of the runs in order: the stored match and display the analysis in chart.

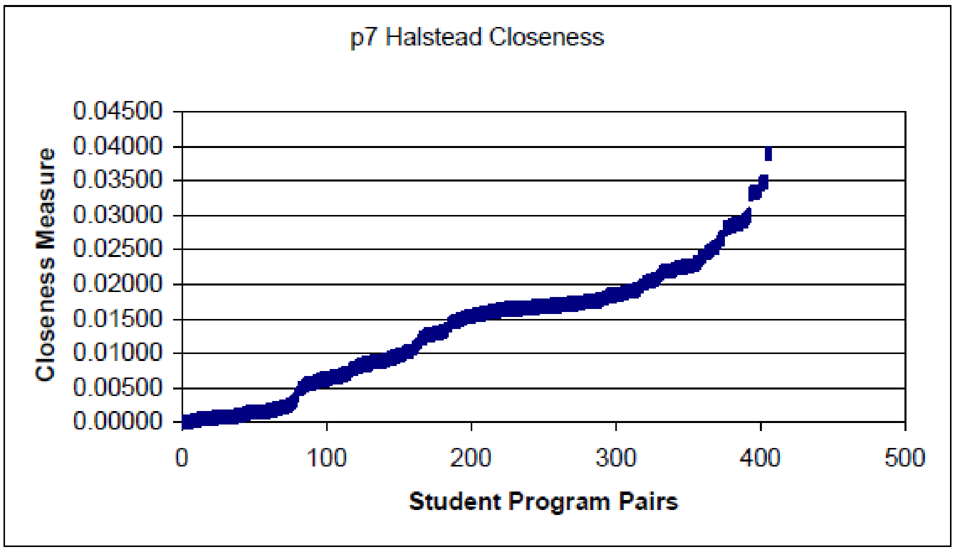


Figure (1): SIM Example-Student Program Pairs (Source: [1])

The main characteristics of SIM are the following:

1. Locality of the tools - the tool's software may be downloaded and the processing can be performed locally, or it may be web-based, with processing taking place on a remote server.
2. SIM are available at the web and source code did not include copyright or license information.

2.3.2.2 MOSS

Measure **o**f **S**oftware **S**imilarity (MOSS) was developed in 1994 by Alex Aiken at Berkeley as a system for measuring the similarity of source code written in C, C++, Java, or Pascal. MOSS tests the source code in real file be parse the source code, tokenizing it and apply comparison algorithm (MOSS) to the tokenized form of the code. And compare it with the source code in other files.

2.3.2.3 JPlag

The amount of information given about JPlag is very sparse. JPlag does not compare to the internet. It is designed to find similarities among the student assignments, which is usually sufficient for computer programs. However its main function is to convert the programs into token strings and comparing these strings.

The official JPlag website [8] summarizes its work a s follow: "*Similarities of 0% or 5% can be represented by the similarity value alone this clearly is no plagiarism. Likewise, similarities of 100% can also be represented by the similarity value alone — this clearly is a plagiarism. But what if the similarity is 40%? Such cases should usually be investigated by a human being for final judgment*."

The main characteristic of JPlag can be summarized as follow:

1. JPlag is available as a web service.
2. JPlag has a powerful user interface for understanding the results.
3. JPlag is resource-efficient and scales to large submissions.
4. JPlag has very good plagiarism detection performance

2.3.2.4 SID

Shared **I**nformation **D**istance or **S**oftware **I**ntegrity **D**etection (SID) detects similarity between programs (source code) by computing the shared information between them.

SID is easy to use software to detect plagiarism within source code and has shown to be the most effective at catching cheaters. SID currently supports Java and C++ source codes. For two programs to be compared, SID computes the shared amount of information between two programs, the shared information distance between two programs X and Y is defined as:

$$(x, y) = \frac{k(x) - k(x/y)}{k(xy)}$$

SID works in two phases:

1. In the first phase, source programs are parsed to generate token sequences by standard lexical analyzer.
2. In the second phase, Token Compress algorithm is used to compute the shared information metric d(x, y) between each program pair within the assignments.

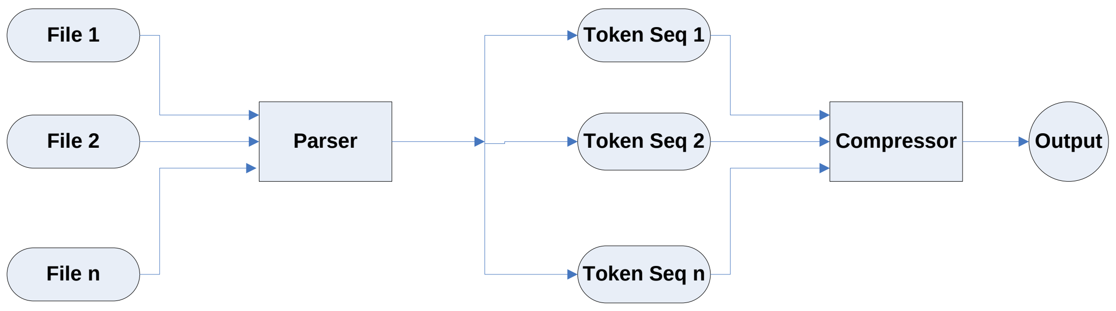


Figure (2): SID Phases (adapted from [2])

2.3.2.5 AC

AC presents a website to detect similarity between assignments or programs and can be used by any person free. This website provides statistical analysis and several graphical visualizations aid in the interpretation of analysis results. AC tools available for research and development at: http://tangow.ii.uam.es/ac.

AC performs the following steps to compare between students assignments:

1. **Distance integration**

   This stage put the characters in sequence and converting them into sequence of tokens after removing comments and spaces from the source file.

2. **Token counting similarity distance**

   This stage counts the tokens between two assignments using parser (compiler to compare the similarities between the two sequences) and gives the percentage of similarity.

The comparison process in AC can be further explained as follow:

1. Input the source code or characters are split in sequence.
2. Lexical analysis, create tokens by converting the sequence of characters into sequence of tokens to become meaningful symbols. For example, 12*(20+12)/40 it becomes after lexical analysis: 12,*, (, 20, +, 12,), /, 40.
3. Tokens are ready.
4. Syntactic analysis checks that the tokens form allowable expression.

5. Semantic parsing makes the actual parsing by comparing the sequences and gives of the outputs.

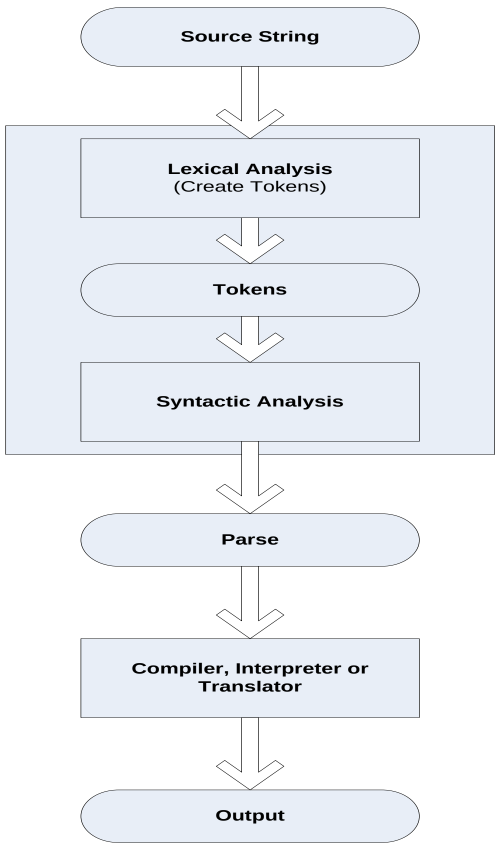


Figure (3): Parsing Steps (adapted from [3])

AC characteristics can be summarized as follow:

1. **Locality of the tool:** it may be downloading and using it locally, it is completely stand-alone, and can be run in any computer with a suitable Java runtime environment.
2. **Breadth:** number of programming languages and variants that the tool can process.
3. **Privacy:** It may only be available to its host, or it may be available for wide use.
4. **Documentation:** existence of documentation about the tool of AC, source code and associated programs, which facilitate the comprehension.
5. **Algorithms:** quality and variety of the similarity distances incorporated into the tool.
6. **Visualization:** existence graphical and chart tool.
7. **Support:** the availability of long-term technical support of the tool.

2.3.2.6 CodeMatch

CodeMatch compares thousands of source code files in multiple directories and subdirectories to determine which files are the most highly correlated. This can be used to significantly speed up the work of finding source code plagiarism, because it can direct the examiner to look closely at a small amount of code in a handful of files rather than thousands of combinations. CodeMatch is also useful for finding open source code within proprietary code, determining common authorship of two different programs, and discovering common, standard algorithms within different programs.

CodeMatch compares every file in one directory with every file in another directory, including all subdirectories if requested. CodeMatch produces a database that can then be exported to an HTML basic report that lists the most highly correlated pairs of files. You can click on any particular pair listed in the HTML basic report see an HTML detailed report that shows the specific items in the files (statements, comments, identifiers, or instruction sequences) that caused the high correlation.

It’s currently supports the following programming languages BASIC, C, C++,C#, Delphi, Flash, Java, JavaScript, MASM, Pascal, Perl, PHP, PowerBuilder, Ruby, SQL, VHDL.

The Algorithms

CodeMatch uses several algorithms to determine similarity between two source code files. When multiple files are compared, file pairs are given a correlation score between 0 and 100.

Statement Matching

The CodeMatch Statement Matching algorithm compares each functional line of source code from both files, excluding comment lines. Also, all whitespace removed. For source lines to be considered matches, they must contain at least one non-keyword such as a variable name or function name.

Comment Matching

The CodeMatch Comment Matching algorithm compares each line of comments from both files. Sequences of whitespace are converted to single spaces so that the syntax structure of the line is preserved. The entire comment is compared, regardless of whether there are keywords in the comment or not.

Identifier Matching

For each file pair, the CodeMatch Identifier Matching algorithm counts the number of matching identifiers that are not programming language keywords. In order to determine whether an identifier is a programming language keyword, comparison is done with a list of programming language keywords. Only non-keywords are compared in order to find matching function names, variable names, and other identifiers.

Correlation Score

Finally, a single correlation score is given for the similarity of the file pairs. If a file pair has a higher score, it implies that these files are more similar and may be plagiarized from each other. CodeMatch reduces the effort needed by the expert by allowing him to narrow his focus from hundreds of thousands of lines of code in hundreds of files to dozens of lines of code in dozens of files.

2.4 Conclusions

Based on the information presented in section 2.3, AC is nominated to be used for the implementation of the project. The reasoning of this selection can be explained as follow:

1. AC is free and open source.
2. AC supports both programming (source-code) and other natural language files.

These points in addition to AC features stated in section 2.3.2.5 make it a good candidate for this project. In this project we cannot use the AC code as it because it is difficult to integrate it with model code, so we analysis the code and reprogramming the main algorithms by using PHP to use it in the plagiarism detection system.

Table (5): Algorithms Comparison

	JPlag	Moss	SID	CodeMatch	SIM	AC
System Started From	1997	1994	1994	2005	2003	2007
Language (s) Supported	Java, C#, C, C++, Scheme and natural language text.	C, C++, Java, C#, Python, Visual Basic, Javascript, FORTRAN, ML, Haskell, Lisp, Scheme, Pascal, Modula2, Perl, TCL, Matlab, VHDL, Verilog, Spice, MIPS Assembly 8086, HCL2.	Java, C++.	BASIC, C, C++, C#, Delphi, Flash ActionScript, Java, JavaScript, MASM, Pascal, Perl, PHP, PowerBuilder, Ruby, SQL, Verilog, VHDL.	C, Java, Pascal and natural language.	C, Java, natural language.
Cost	Free but user must create an account	Free but user must create an account	Free and open sourced	Commercial tool, free on any code where the total of all files being examined is less than 1	Free and open sourced	Free and open sourced

				megabyte		
Service	Web service	Internet service	Standalone application	Standalone application	Standalone application	Standalone application
Interface	GUI	GUI	GUI	GUI	GUI	
Requirements	Web browser, Java Runtime Environment (JRE), Java 1.5 or higher	A submission script for either UNIX or Windows	JDK 1.4 or later		JDK 1.4 or later	Java runtime environment
Security	User id and e-mail needed	User id and e-mail needed	Runs locally	Runs locally	Runs locally	
Submission Methods	Standalone Java software application that can be deployed over the network	Command line	Standalone Java application	Standalone application	Standalone application	Standalone application
Source Data	Files, or a directory with or without subdirectories	Files, or a directory with or without subdirectories	A directory containing subdirectories, each containing one or more zip archives or submissions	Files, or a directory with or without subdirectories	Files, or a directory with or without subdirectories	Files, or a directory with or without subdirectories

Table (5): Cont.

Speed	Fast	Fast	Large sets of files will require long amounts of time to analyze	Large sets of files will require long amounts of time to analyze	Fast	Fast
Algorithms	Greedy String Tiling	Winnowing Algorithm	Token based matching algorithm for source-code, string matching for natural language texts	String matching	Greedy String Tiling	Token based matching algorithm for source-code, string matching for natural language texts

Table (6): Result Display

	JPlag	**Moss**	**SID**	**CodeMatch**	**SIM**	**AC**
Result storage	Local	Remote	Local	Local	Local	Local
Overview results display method	Histogram, statistics about the files that were analyzed	Ordered list	Matching pair Tree	Ordered list	Ordered list	Ordered list
Display of Results	Powerful graphical interface for presenting results	Powerful graphical interface for presenting results	Powerful graphical interface for presenting results	Powerful graphical interface for presenting results	Results displayed in a dialogue	Powerful graphical interface for presenting results

Table (6): Cont.

Visualization of Results	Cross linked listings, scroll bars		2/4 scroll bars, simultaneous scrolling	Table showing the similar lines of code detected among file pairs.	List of results showing the detected code fragments between file pairs.	existence graphical and chart tool.
Visual display of matched file pairs	Yes	Yes	Yes	No	No	Yes
Metrics Produced	Percentage similarity, token matches	Percentage similarity, token matches, lines matched	Percentage similarity, token matches	Percentage similarity, lines matched	Token matches, lines matched	Percentage similarity, token matches.

Chapter Three

System Requirements

This chapter first presents both functional and nonfunctional requirements of the system. It then discusses the feasibility study of the system.

3.1 System Requirements

This section lists both functional and non-functional requirements of the system.

3.1.1 Functional requirements

1. **Enable teachers to detect plagiarism and cheating in student submitted assignments.**
   - The system reads the submitted assignments and enters them to the algorithm to find the degree of similarity between them.

2. **Viewing visually aided cheating (similarity) reports.**
   - Teachers can display cheating (plagiarism) report, which contains all submitted assignments and the percentage of similarity of each assignment with others.

     The main functions such a registration, login, create courses are already exist in the Moodel (it is not new functions to be added in this project).

3.1.2 Non-Functional requirements

1. Compatibility

- System should be compatible and integrateable with Moodle because it will be added as new feature to Moodle.

2. Ease to use

- Teachers will interact with the system to generate plagiarism report through a user-friendly graphical user interface. Furthermore, the generated reports will contain both textual and visual (bars, charts, etc.) representation for the results.

3.2 Feasibility Study

This section shows the feasibility study for system development and operation.

3.2.1 Development requirements

1. **Development hardware resources**

Table (7) below shows the hardware resources that are needed during the development phase along with their costs.

Table (7): Hardware Development Resources and Costs[1]

Hardware Component	Cost
HP EliteBook, Core 2 Duo SL9400 / 1.86 GHz LV Centrino 2 with vPro, RAM 3 GB - HDD 120 GB - GMA 4500MHD, Win XP Pro - 12.1" Widescreen TFT 1280 x 800.	**1609$**
Total	**1609$**

2. **Development software resources**

Table (8) below shows the software resources that are needed during the development phase along with their costs.

Table (8): Software Development Resources and Costs [1]

Software Component	Cost
Windows XP professional	**159 $**
PHP	**0**
MySQL Server	**0**
Apache	**0**
Microsoft Office 2007	**103$**
Total	**262 $**

[1](http://www.amazon.com, accessed: 10/10/09)

3. **Development human resources**

The cost of development is zero because the developers of the system are the project group team and the system development is part of their graduation requirements.

3.2.2 Operational requirements

1. **Operational hardware resources**

This system will be hosted on a server that is already located at the computer center, table (9) below shows the hardware resources that are needed during the implementation phase along with their costs.

Table (9): Operational Hardware Resources and Costs

Hardware Component	Cost
Server HP ProLiant DL 380G5 1 CPU, Intel ® 5000P Chipset, RAM DDR2-667 32 GBs	**0 $**
Total	**0$**

2. **Operational software resources and costs**

Table (10) below shows the software resources that are needed during the implementation phase along with their costs.

Table (10): Operational Software Resources and Costs[2]

Software Component	Cost
PHP	**0**
MySQL Server	**0**
Apache	**0**
Total	**0**

[2](http://www.php.net/ ,accessed: 10/10/09)

3. Operational human resources

The human operational cost is zero, because the e-learning supervisor and network supervisor are employees at Palestine polytechnic university and they already paid salaries for their jobs.

3.2.3 Total cost

Table (11) below shows the total costs for the hardware, software, and human resources that are required for both system development and implementation.

Table (11): Total Costs

Development Costs	Operational Costs	Total
1871$	**0$**	**1871 $**

Chapter Four

System Specification

4.1 Use Case

The Use Case is used to describe the system as whole. As shown in Figure (4), the system has a database and two user types; namely: teacher and student.

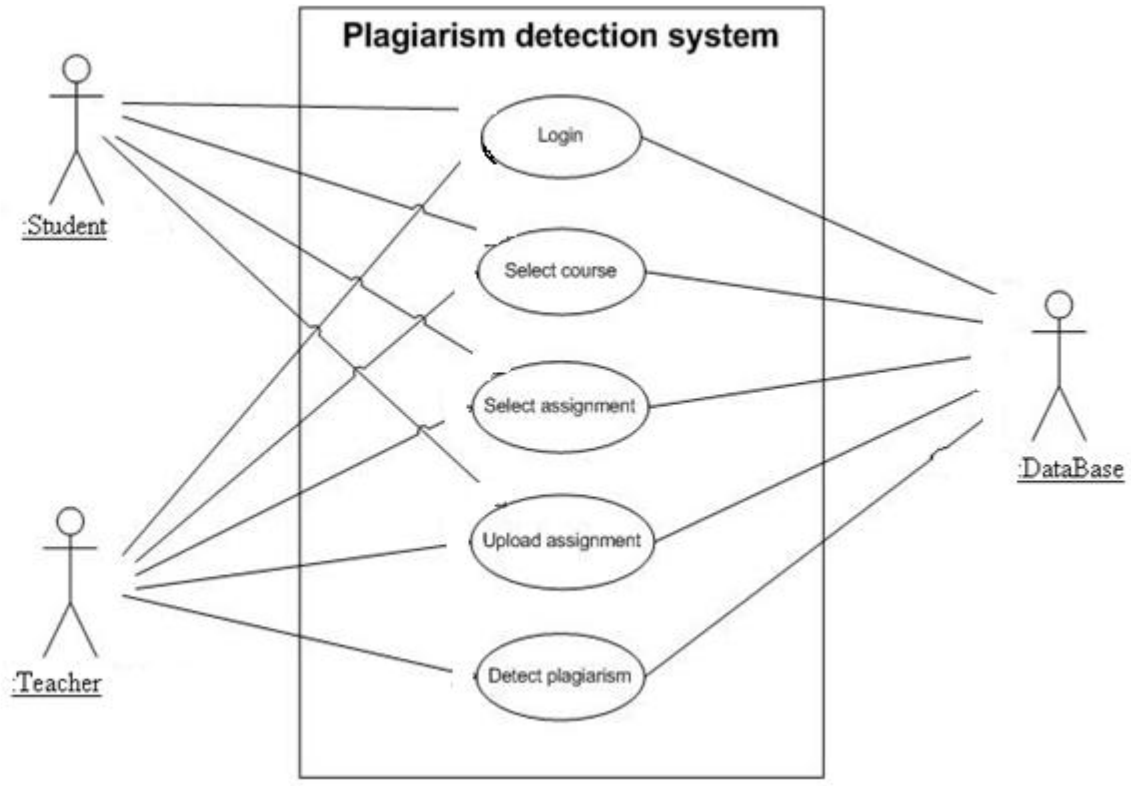


Figure (4): Use Case

4.2 Class Diagram

The Class Diagram is used to describe the main classes and their roles in the system. This diagram is shown in Figure (5).

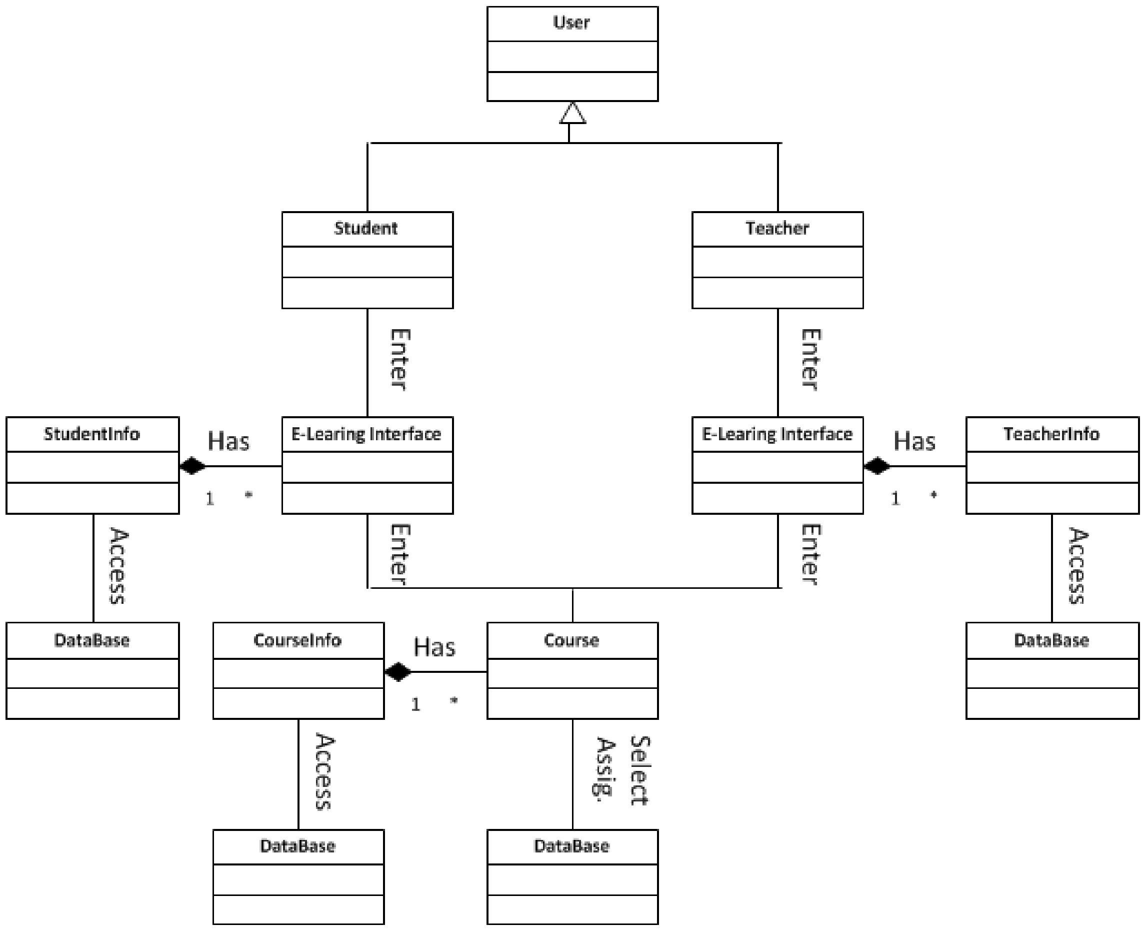


Figure (5): Class Diagram.

4.3 Object Diagram

The Object Diagram is used to describe the main objects in the system as shown in the Figure (6).

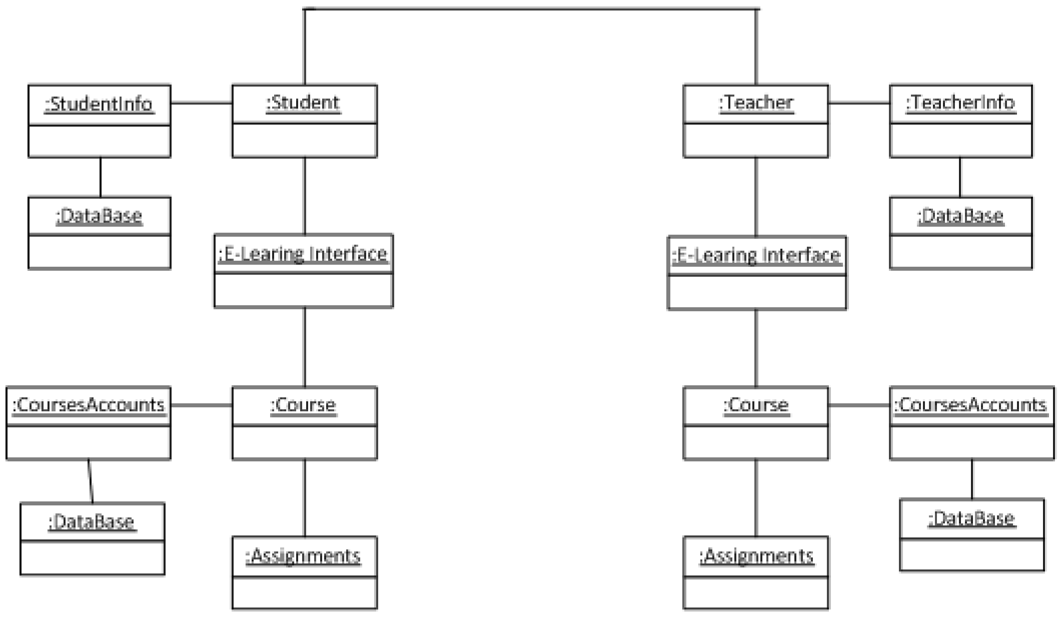


Figure (6): Object Diagram

4.4 Sequence Diagram

The Sequence Diagram is used to describe the system sequence. For this system, the system sequencing has been depicted in two scenarios: teacher and student (as shown in the Figure (7) and Figure (8) respectively).

4.4.1 Teacher scenario

In this scenario

- The teacher can interacts with system by enter username and password.
- The teacher enters to e-learning interface.
- Select course from group of courses, select the submitted assignments.
- Click on button plagiarism the system detect the plagiarism among submitted assignments and view a report about the cases of plagiarism between the students assignments.

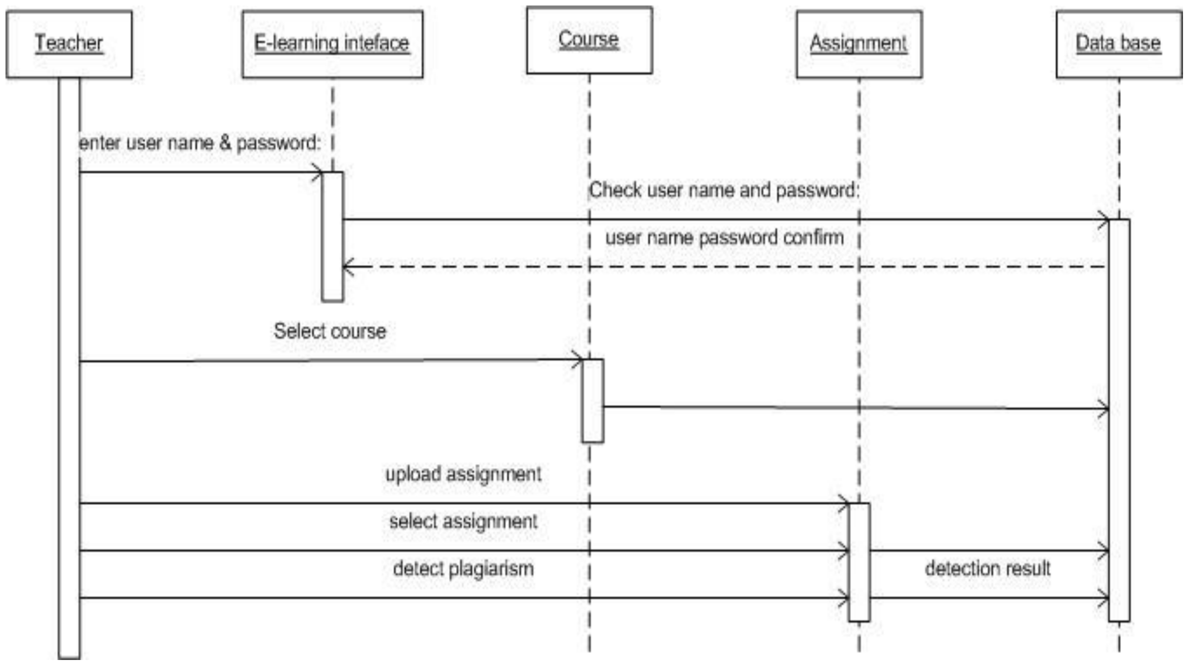


Figure (7): Teacher Sequance Diagram

4.4.2 Student scenario

In this scenario

- The student interacts with the system by enter username and password.
- The student enters to e-learning interface.
- Select course from group of courses and then select the assignment.
- Upload this assignment, and this assignment becomes submitted.

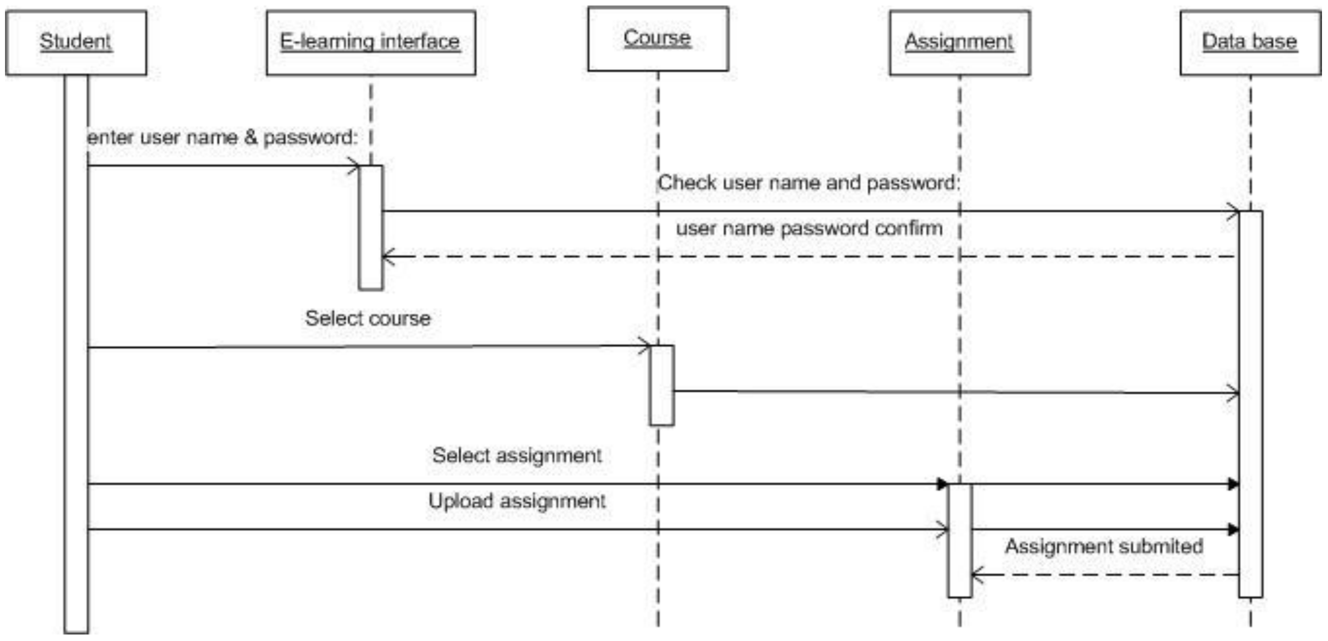


Figure (8): Student Sequence Diagram

4.5 Collaboration Diagram

4.5.1 Teacher collaboration diagram

Figure (9) show the data that flow between object in teachers operation.

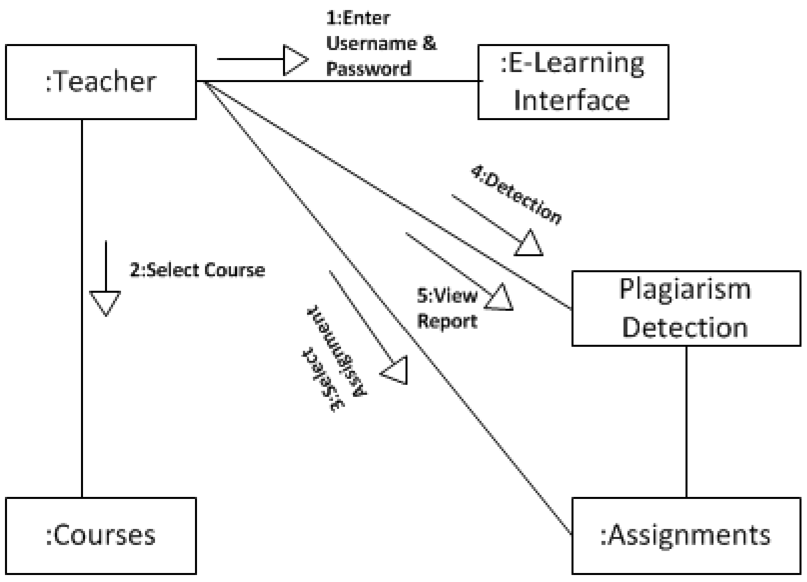


Figure (9): Teacher Collaboration Diagram

4.5.2 Student collaboration diagram

Figure (10) shows the data that flow between object in students operation.

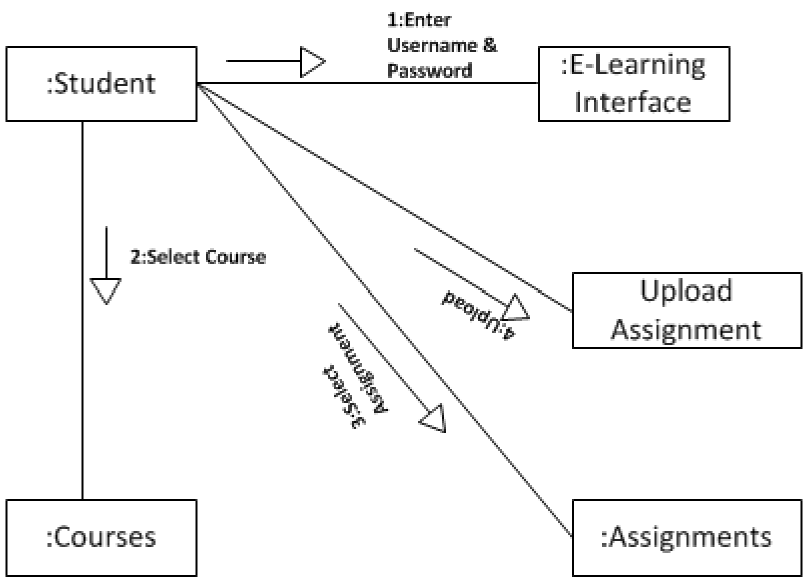


Figure (10): Student Collaboration Diagram

4.6 State Diagram

4.6.1 Teacher state diagram

Figure (11) shows the deferent state for objects depend on its attributes during teacher operation.

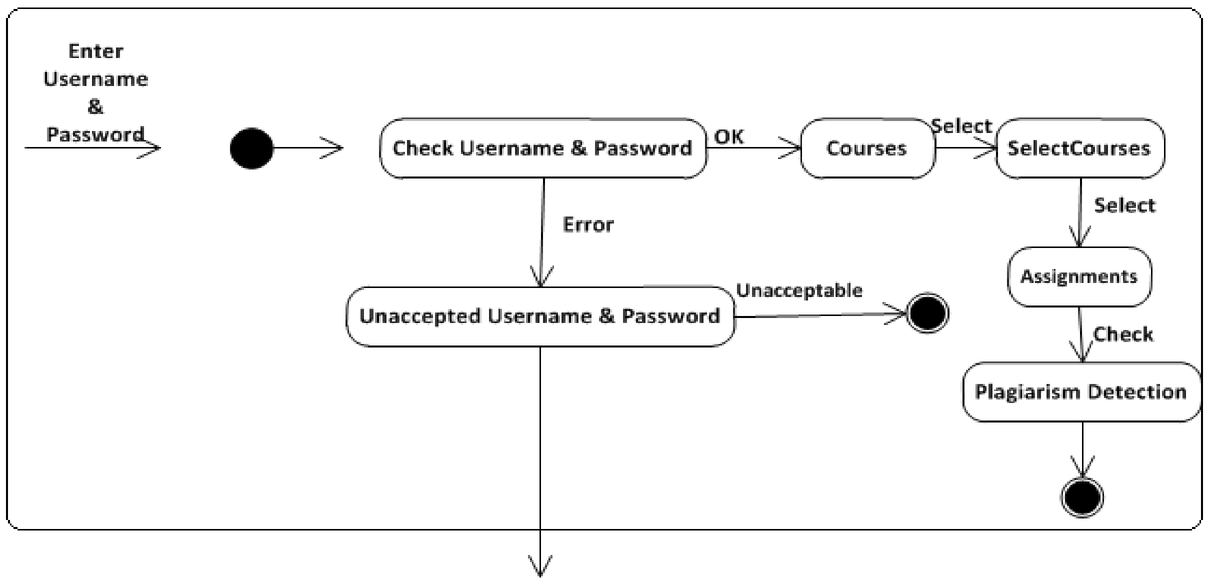


Figure (11): Teacher State Diagram

4.6.2 Student state diagram

Figure (12) shows the different state for objects depend on its attributes during student operation.

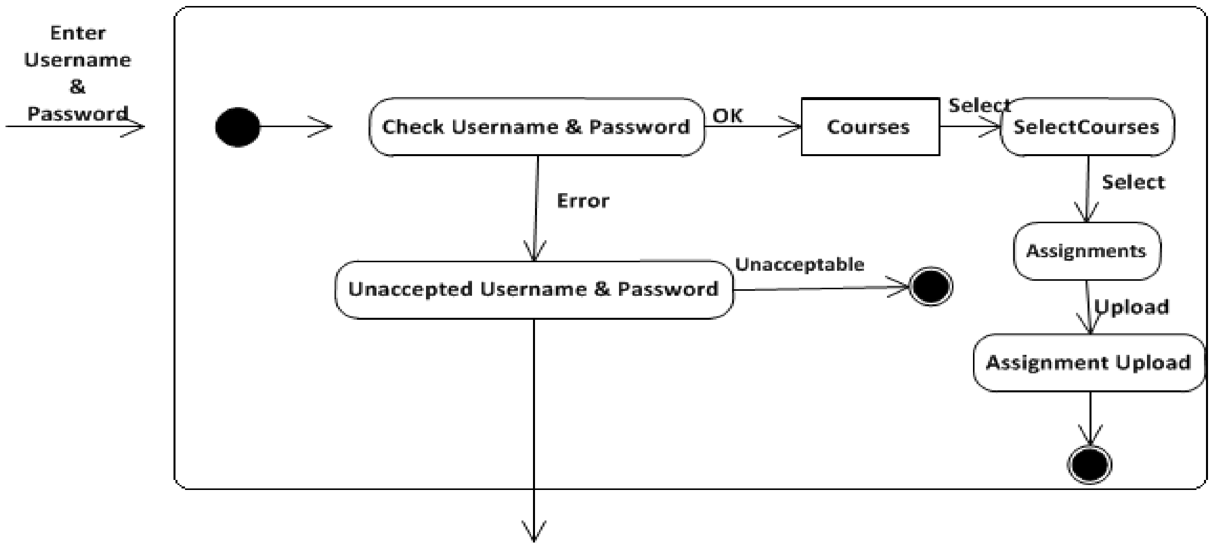


Figure (12): Student State Diagram

4.7 Activity Diagram

4.7.1 Teacher activity diagram

Figure (13) presents the deferent activity for teacher.

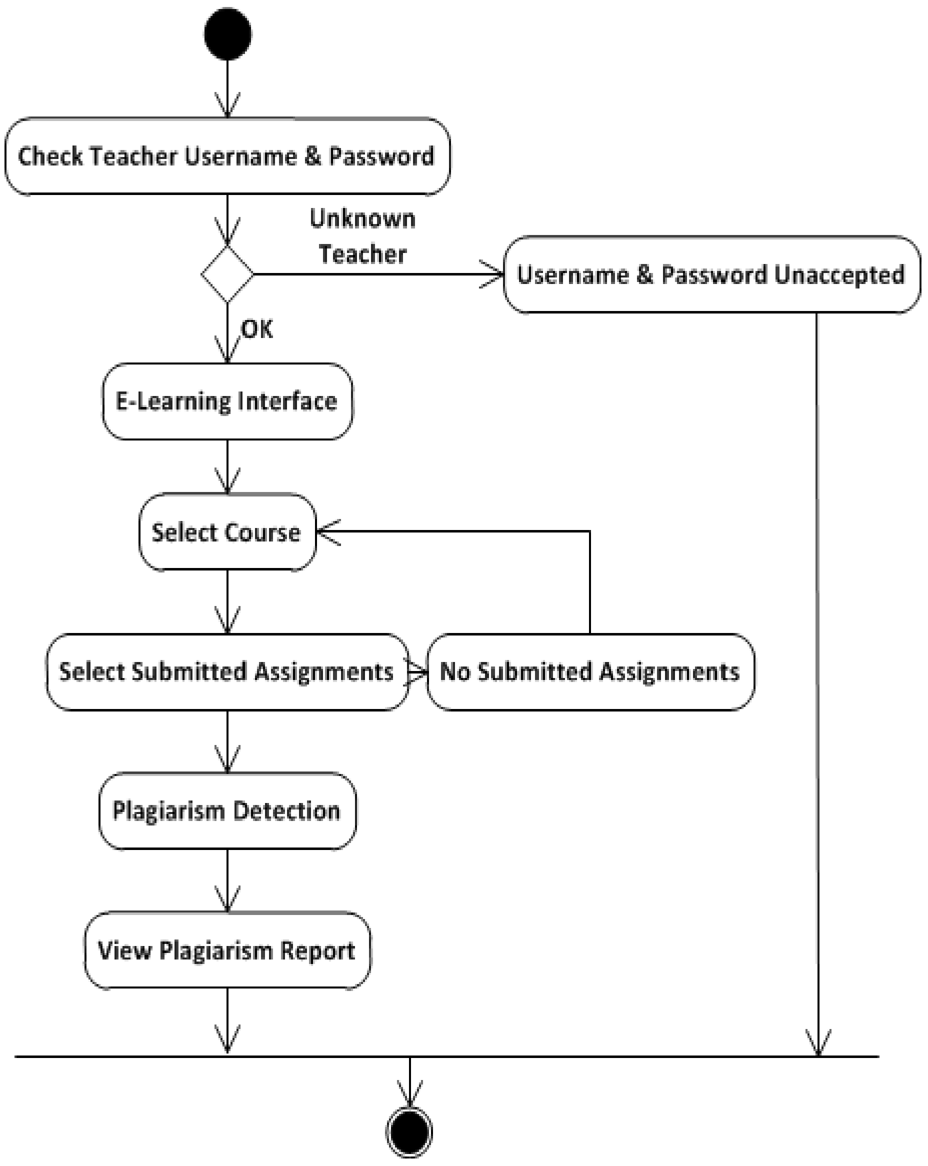


Figure (13): Teacher Activity Diagram

4.7.2 Student activity diagram

Figure (14) presents the deferent activity for teacher.

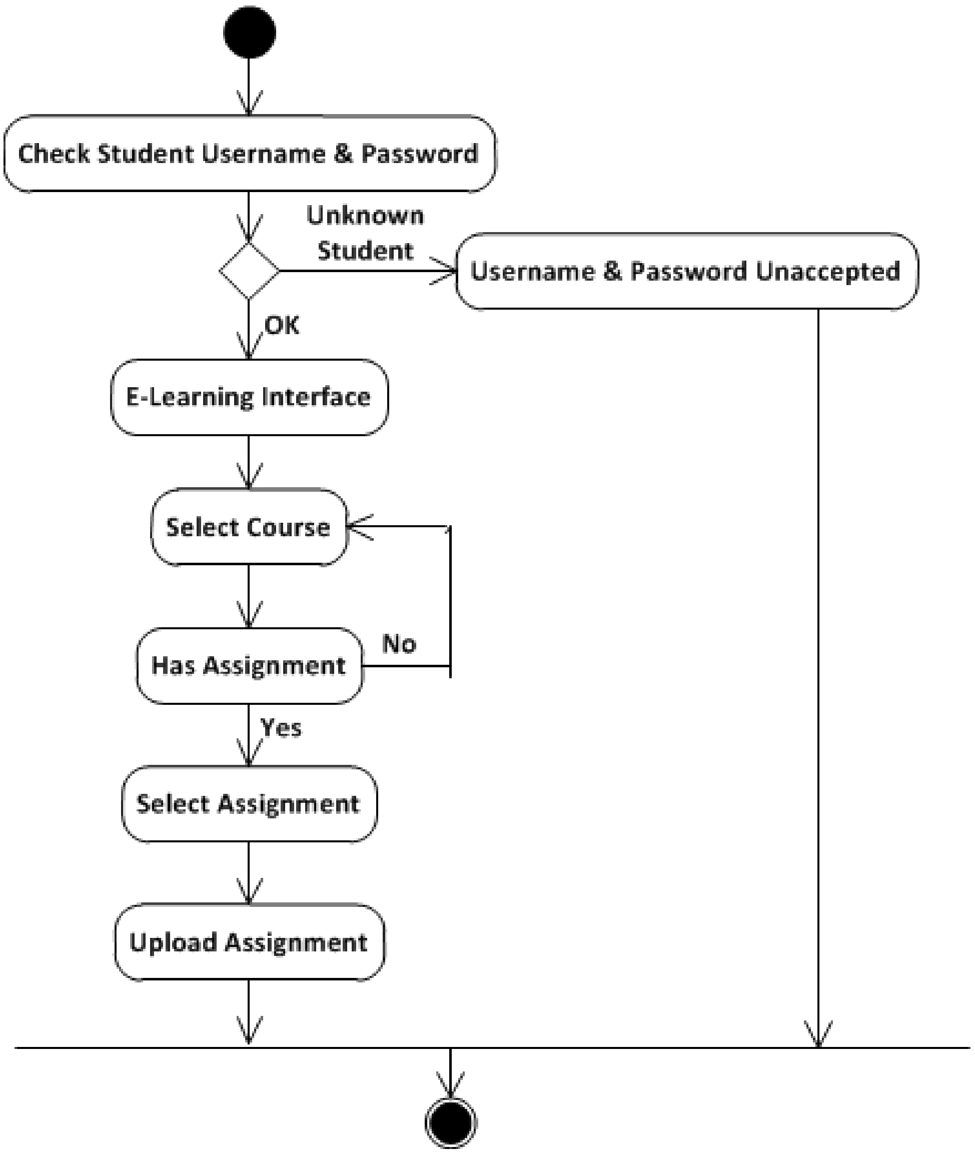


Figure (14): Student Activity Diagram

Chapter Five
Design

5.1 Database Design [3]

This section presents the most relevant tables that are needed to create this system; these tables are already built in Model.

- **Role_Assignment table**

Table (12): Role_Assignment Table

Role_assignment
id: INT roleid: INT contextid: INT userid: INT hidden: INT timestart: INT timeend: INT timemodified : INT modifierid :INT enroll: VARCHAR(20) sortorder :INT

- **Context table**

Table (13): Context Table

Context
id :INT contextlevel: INT instance: INT path:VARCHAR(255) depth:INT

- **Part of User table**

Table (14): User Table

User
id :INT name: VARCHAR(255) major: VARCHAR(255) address: VARCHAR(255) phone: INT

[3] (These tables are subset of the complete database of Moodle; a subset of the schema can be found in Appendix (B)).

5.2Interface Design

- **Login form**

This form is used to authenticate users. The user (teacher or student) enters his / her credentials in the login form (see Figure (15)). Based on successful authentication, the user will be redirected to the *course page*.

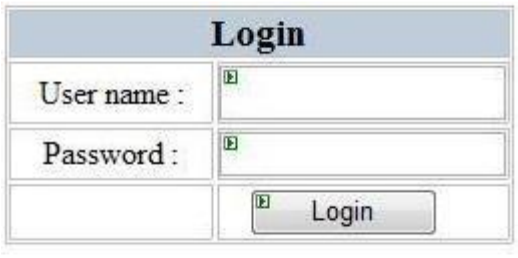
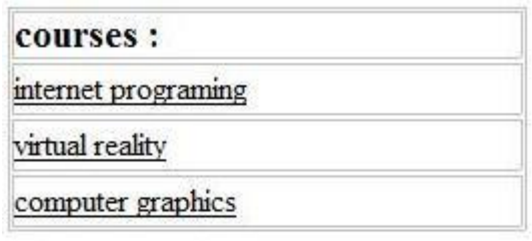


Figure (15): Login Form

- **Select course**

Through this form, the user selects a course from the list of courses he / she was registered in (see Figure (16)).



courses :
internet programing
virtual reality
computer graphics

Figure (16): Select Course

- **Select assignment**

Through this form, the user selects an assignment (see Figure (17)).

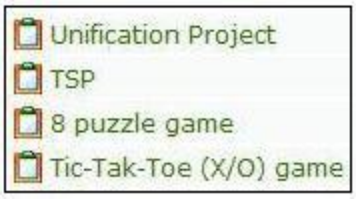


Figure (17): Select Assignment

- **Submit Assignment**

Finally, the student submits his / her solution on the upload form as shown in Figure (18).

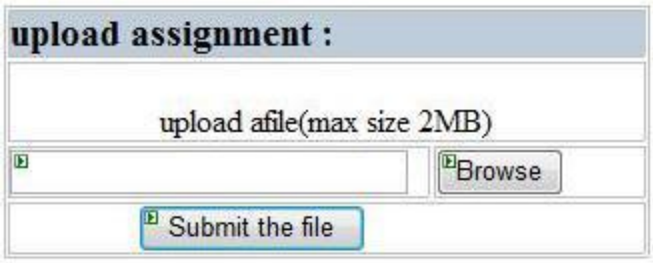


Figure (18): Submit Assignment

- **View Submitted Assignment**

Figure (19) below presents the "*view submitted assignments*" link that let the teacher to browse all submitted assignments.

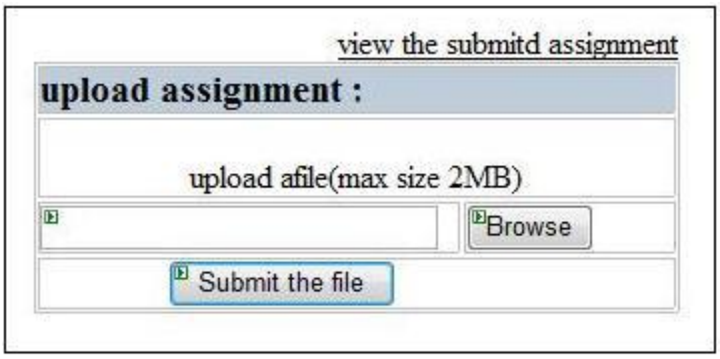
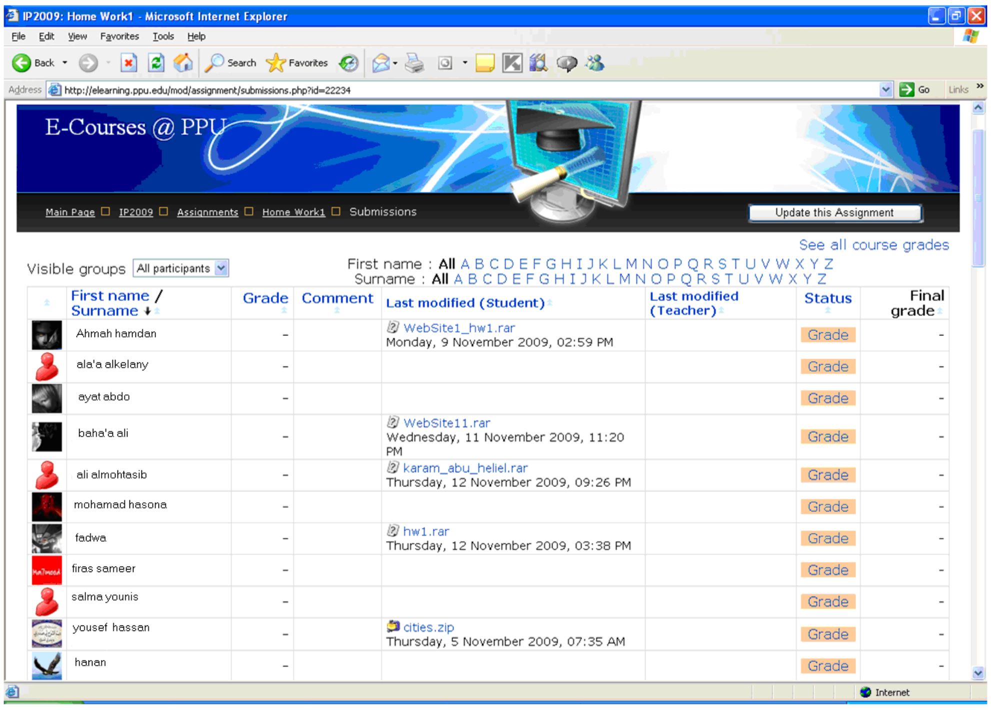


Figure (19): View Submitted Assignment

- **Submitted Assignment**

After the teacher press on the link above, the submitted assignments will be displayed as shown in Figure (20).

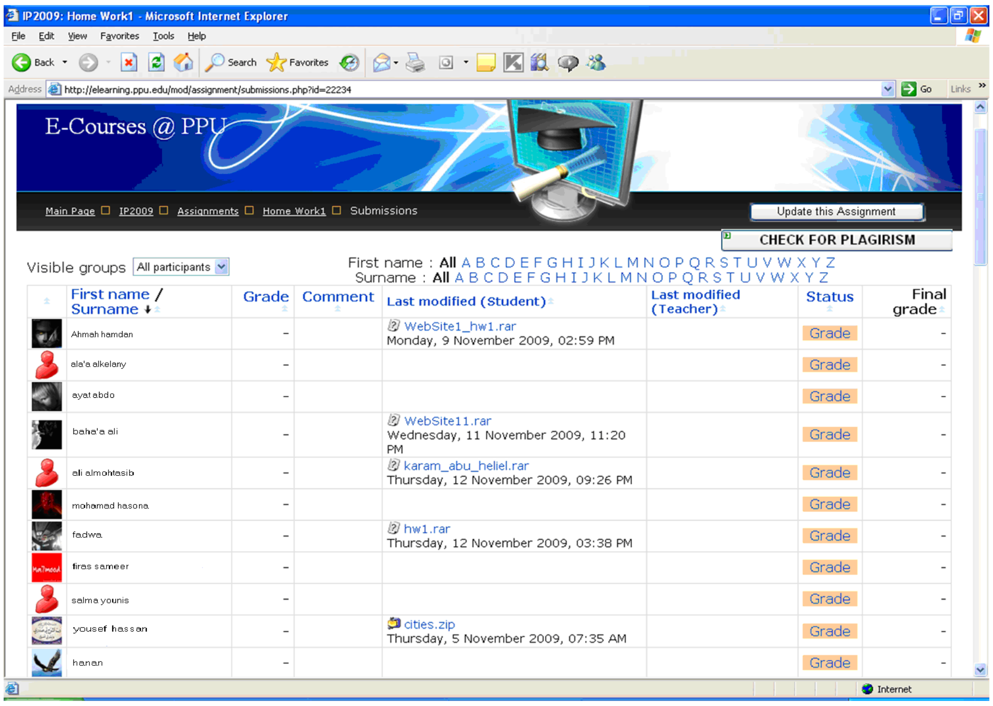

IP2009: Home Work1 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://elearning.ppu.edu/mod/assignment/submissions.php?id=22234
Go Links
E-Courses @ PPU
Main Page IP2009 Assignments Home Work1 Submissions
Update this Assignment
See all course grades
Visible groups All participants
First name : All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Surname : All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Internet

	First name / Surname	Grade	Comment	Last modified (Student)	Last modified (Teacher)	Status	Final grade
	Ahmah hamdan	-		WebSite1_hw1.rar Monday, 9 November 2009, 02:59 PM		Grade	-
	ala'a alkelany	-				Grade	-
	ayat abdo	-				Grade	-
	baha'a ali	-		WebSite11.rar Wednesday, 11 November 2009, 11:20 PM		Grade	-
	ali almohtasib	-		karam_abu_heliel.rar Thursday, 12 November 2009, 09:26 PM		Grade	-
	mohamad hasona	-				Grade	-
	fadwa	-		hw1.rar Thursday, 12 November 2009, 03:38 PM		Grade	-
	firas sameer	-				Grade	-
	salma younis	-				Grade	-
	yousef hassan	-		cities.zip Thursday, 5 November 2009, 07:35 AM		Grade	-
	hanan	-				Grade	-

Figure (20): Submitted Assignments

- **Plagiarism Detection Button**

This project aims to add a new "*CHEACK FOR PLAGIARISM*" button; the teacher can press on the button to browse plagiarism result. This new button an be seen in Figure (21).

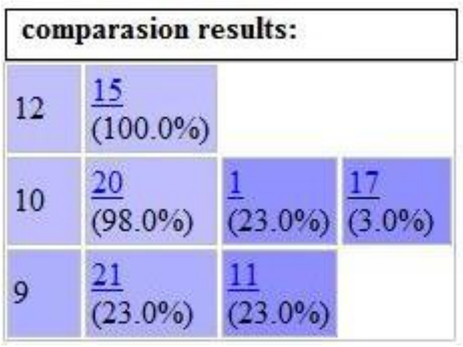

IP2009: Home Work1 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://elearning.ppu.edu/mod/assignment/submissions.php?id=22234
Go Links
E-Courses @ PPU
Main Page IP2009 Assignments Home Work1 Submissions
Update this Assignment
CHECK FOR PLAGIRISM
Visible groups All participants
First name : All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Surname : All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Internet

	First name / Surname	Grade	Comment	Last modified (Student)	Last modified (Teacher)	Status	Final grade
	Ahmah hamdan	-		WebSite1_hw1.rar Monday, 9 November 2009, 02:59 PM		Grade	-
	ala'a alkelany	-				Grade	-
	ayat abdo	-				Grade	-
	baha'a ali	-		WebSite11.rar Wednesday, 11 November 2009, 11:20 PM		Grade	-
	ali almohtasib	-		karam_abu_heliel.rar Thursday, 12 November 2009, 09:26 PM		Grade	-
	mohamad hasona	-				Grade	-
	fadwa	-		hw1.rar Thursday, 12 November 2009, 03:38 PM		Grade	-
	firas sameer	-				Grade	-
	salma younis	-				Grade	-
	yousef hassan	-		cities.zip Thursday, 5 November 2009, 07:35 AM		Grade	-
	hanan	-				Grade	-

Figure (21): Plagiarism Detection Button

- **Comparison Result**

  Figure (22) shows a sample cheating (comparison) results.



comparasion results:

12	15 (100.0%)		
10	20 (98.0%)	1 (23.0%)	17 (3.0%)
9	21 (23.0%)	11 (23.0%)	

Figure (22): Comparison Results

5.3 Flowcharts

Student operation

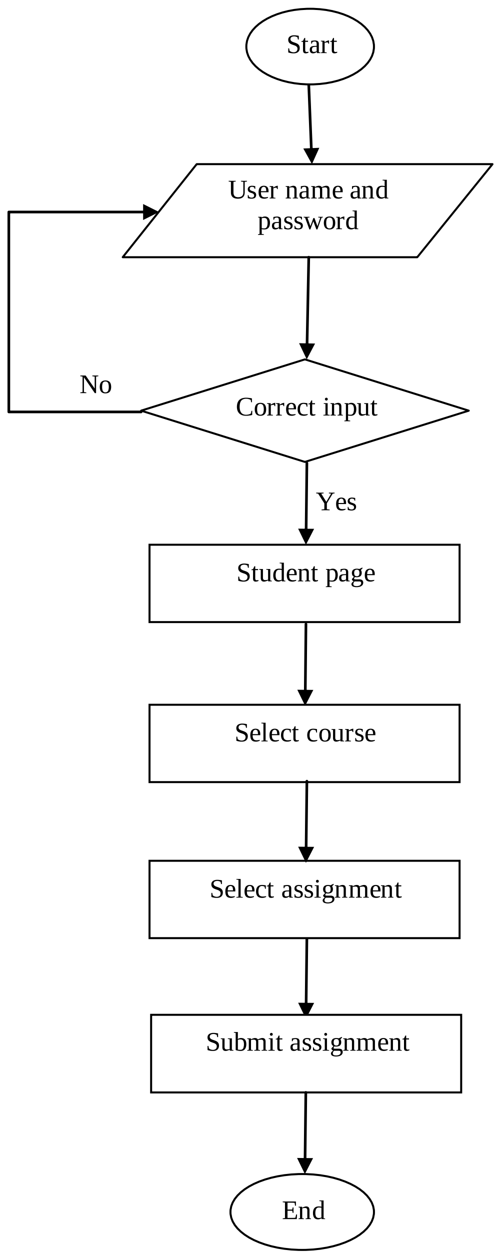


Figure (23): Student Operation

Teacher operation

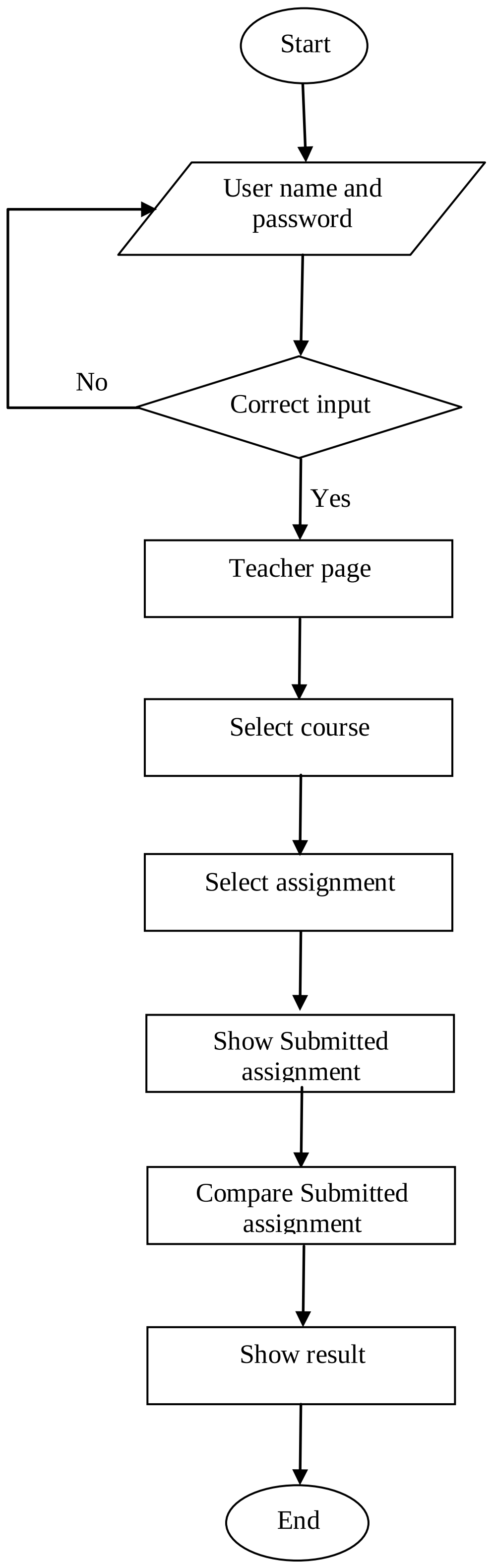


Figure (24): Teacher Operation

Chapter Six

Implementation

6.1 Installation Environment

The man installation requirements of Moodle include: Apache, MySQL, and PHP. It's regularly tested with Windows XP/2000/2003, Mac OS X and Netware 6 operating systems. However, the following is a more detailed listing of Moodle hardware and software requirements.

Moodle requirements

1. **Hardware**
    1. Disk space: 160MB free (min). To store the teaching materials it requires more free space.
    2. Memory: 256MB (min), 1GB (recommended). Moodle can support 50 concurrent users for every 1GB of RAM, but this will vary depending on the specific hardware and software combination.
    3. The capacity can limit the number of users that Moodle site can handle.

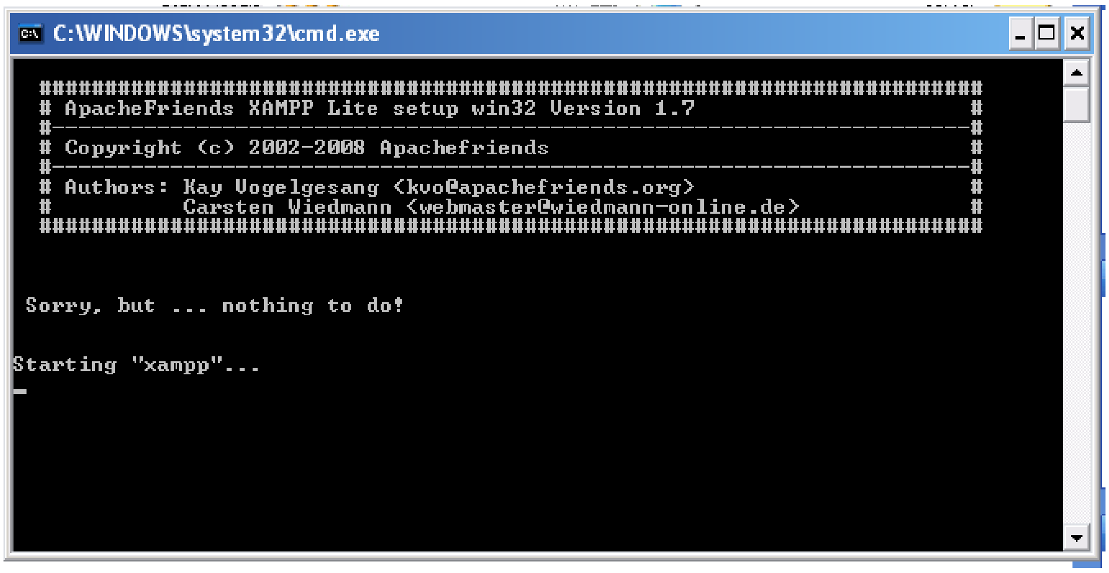
2. **Software**
    1. Web server software: Moodle can work fine under any web server that supports PHP, such as IIS on Windows platforms.
    2. PHP scripting language. There are currently two versions of PHP available: PHP4 and PHP5.

Start Moodle installation

The Moodle package is available at: http://download.moodle.org/. The package is free for download. After extracting the ZIP file, the resulted folder will contain the sever sub-folder, Start Moodle.exe, Stop Moodle.exe and readme file.

To install Moodle:

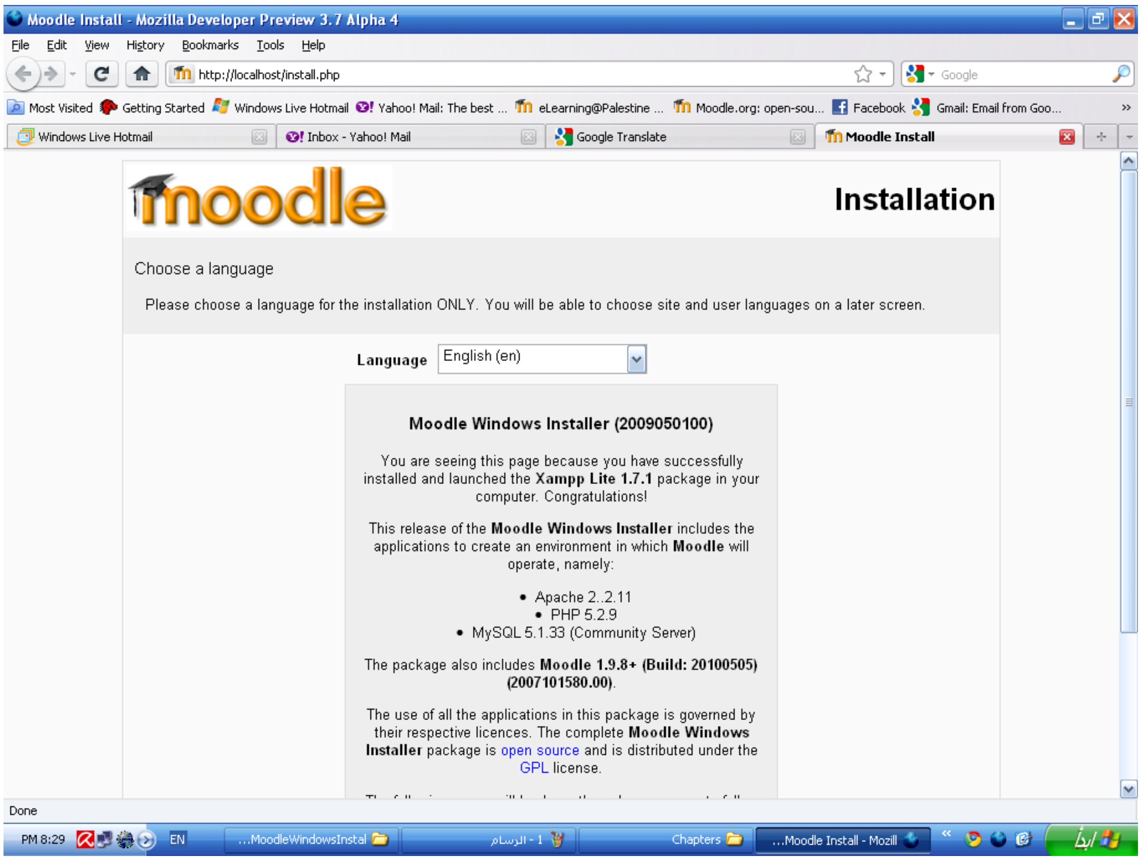
- Click on the Start Moodle.exe button, after you click on this button, you see a CMD prompt interface; this interface displays Moodle version and other information; it shows two choices the 0 for exit and 1 for refresh; to start enter 1, and XAMPP sever will start. The screen is shown in Figure (25).



```
C:\WINDOWS\system32\cmd.exe

 ###############################################################################
 # ApacheFriends XAMPP Lite setup win32 Version 1.7                            #
 #-----------------------------------------------------------------------------#
 # Copyright (c) 2002-2008 Apachefriends                                       #
 #-----------------------------------------------------------------------------#
 # Authors: Kay Vogelgesang <kvo@apachefriends.org>                            #
 #          Carsten Wiedmann <webmaster@wiedmann-online.de>                    #
 ###############################################################################


 Sorry, but ... nothing to do!

Starting "xampp"...
_
```

Figure (25): Start Moodle

- After running the XAMPP server, write http://localhost on the browser address, then the Moodle language selection page will appear as shown in Figure (26).



Moodle Install - Mozilla Developer Preview 3.7 Alpha 4

http://localhost/install.php

Installation

Choose a language

Please choose a language for the installation ONLY. You will be able to choose site and user languages on a later screen.

Language: English (en)

Moodle Windows Installer (2009050100)

You are seeing this page because you have successfully installed and launched the **Xampp Lite 1.7.1** package in your computer. Congratulations!

This release of the **Moodle Windows Installer** includes the applications to create an environment in which **Moodle** will operate, namely:

- Apache 2..2.11
- PHP 5.2.9
- MySQL 5.1.33 (Community Server)

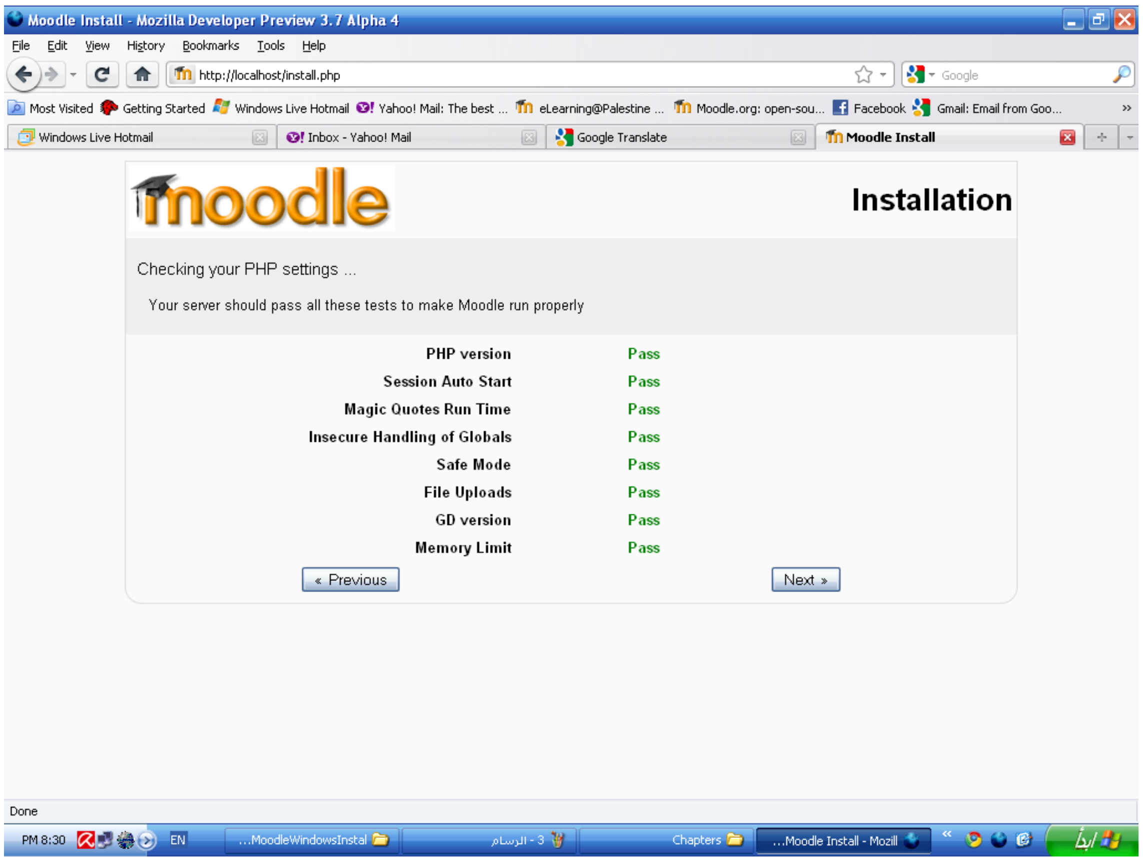
The package also includes **Moodle 1.9.8+ (Build: 20100505) (2007101580.00)**.

The use of all the applications in this package is governed by their respective licences. The complete **Moodle Windows Installer** package is open source and is distributed under the GPL license.

Done

Figure(26): Moodle Language Selection Interface

- After selecting the language, you will be redirected to the PHP settings screen as show in Figure (27).


Moodle Install - Mozilla Developer Preview 3.7 Alpha 4
http://localhost/install.php
Installation

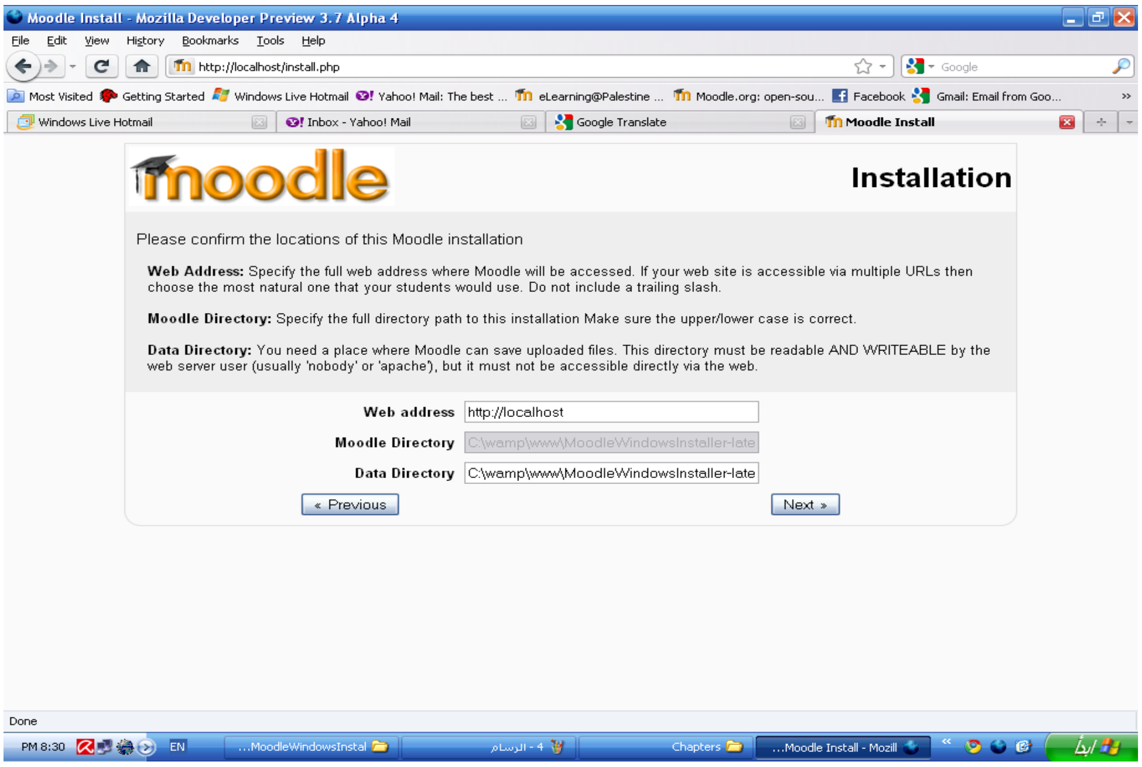
Checking your PHP settings ...

Your server should pass all these tests to make Moodle run properly

PHP version	Pass
Session Auto Start	Pass
Magic Quotes Run Time	Pass
Insecure Handling of Globals	Pass
Safe Mode	Pass
File Uploads	Pass
GD version	Pass
Memory Limit	Pass

Figure (27): Checking PHP Setting

- The next screen (see Figure (28)) will ask the administrator to fill-in the following information:
    1. Web address: specify the full web address where model will be accessed.
    2. Moodel directory: specify the full directory path to this installation.
    3. Data directory: a place where Moodel can save uploaded files.


Please confirm the locations of this Moodle installation

Web Address: Specify the full web address where Moodle will be accessed. If your web site is accessible via multiple URLs then choose the most natural one that your students would use. Do not include a trailing slash.

Moodle Directory: Specify the full directory path to this installation Make sure the upper/lower case is correct.

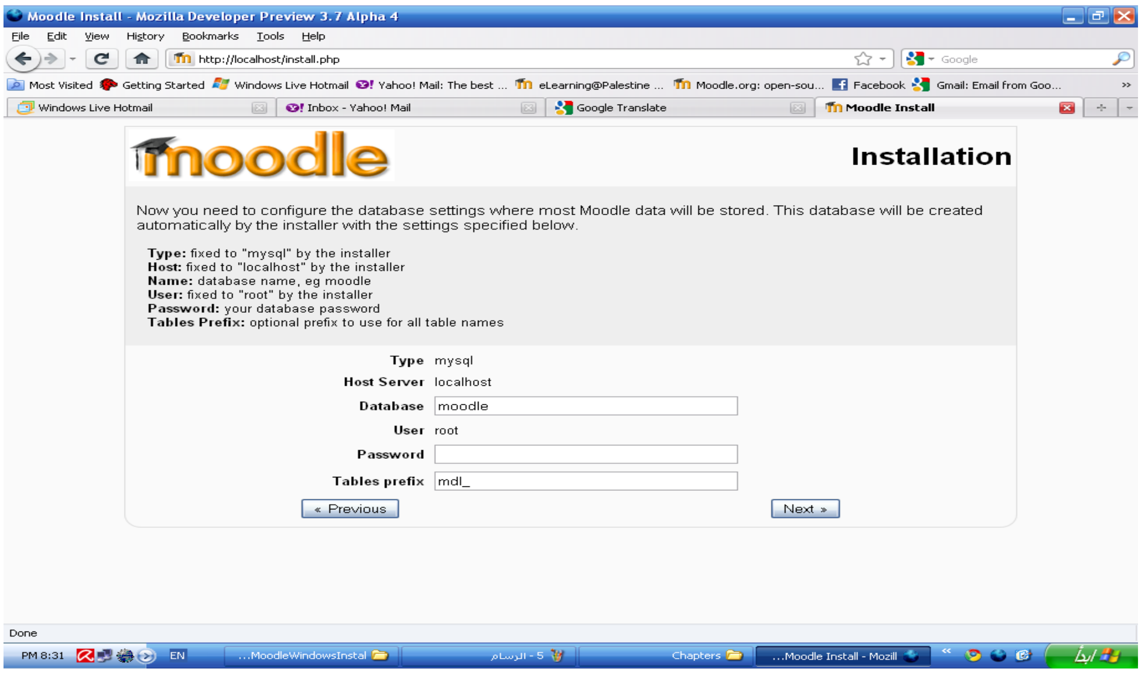
Data Directory: You need a place where Moodle can save uploaded files. This directory must be readable AND WRITEABLE by the web server user (usually 'nobody' or 'apache'), but it must not be accessible directly via the web.

Web address	http://localhost
Moodle Directory	C:\wamp\www\MoodleWindowsInstaller-late
Data Directory	C:\wamp\www\MoodleWindowsInstaller-late

Moodle Install - Mozilla Developer Preview 3.7 Alpha 4
http://localhost/install.php
Installation
« Previous
Next »
Done

Figure (28): Confirm Moodel Location

- The next step is to configure Moodle database settings. The installer as shown in Figure (29) creates this database automatically.


Now you need to configure the database settings where most Moodle data will be stored. This database will be created automatically by the installer with the settings specified below.

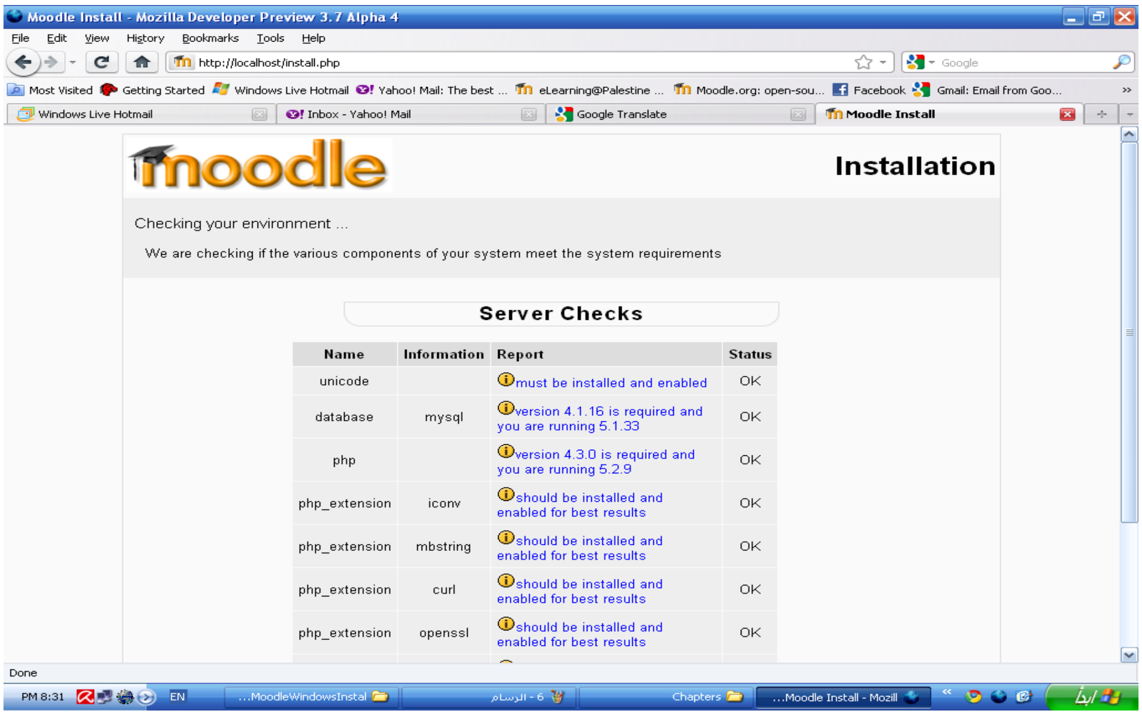
Type: fixed to "mysql" by the installer
Host: fixed to "localhost" by the installer
Name: database name, eg moodle
User: fixed to "root" by the installer
Password: your database password
Tables Prefix: optional prefix to use for all table names

Type	mysql
Host Server	localhost
Database	moodle
User	root
Password	
Tables prefix	mdl_

Moodle Install - Mozilla Developer Preview 3.7 Alpha 4
http://localhost/install.php
Installation
« Previous
Next »
Done

Figure (29): Configure Database Setting

- The next step (Figure (31)) is to check if the various components of the system meet the system requirements.


Moodle Install - Mozilla Developer Preview 3.7 Alpha 4
http://localhost/install.php
moodle
Installation

Checking your environment ...

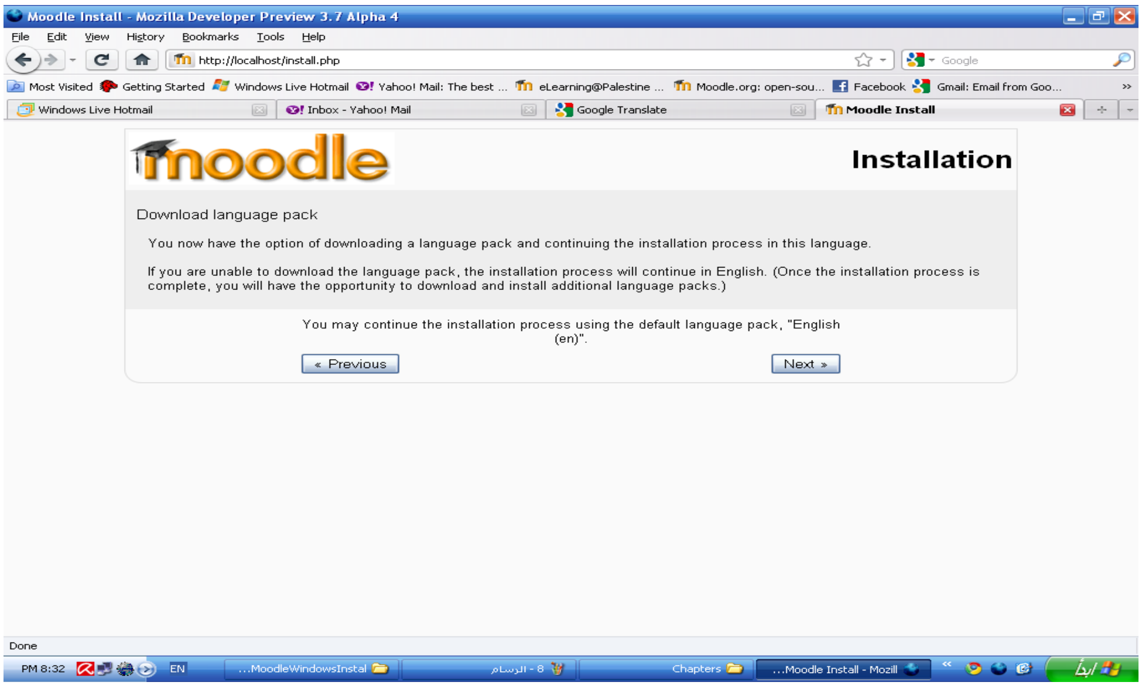
We are checking if the various components of your system meet the system requirements

Server Checks

Name	Information	Report	Status
unicode		must be installed and enabled	OK
database	mysql	version 4.1.16 is required and you are running 5.1.33	OK
php		version 4.3.0 is required and you are running 5.2.9	OK
php_extension	iconv	should be installed and enabled for best results	OK
php_extension	mbstring	should be installed and enabled for best results	OK
php_extension	curl	should be installed and enabled for best results	OK
php_extension	openssl	should be installed and enabled for best results	OK

Figure (30): Server Checks

- If you want to change the installation language, you need to download the language pack as shown in Figure (31).


Moodle Install - Mozilla Developer Preview 3.7 Alpha 4
http://localhost/install.php
moodle
Installation
« Previous
Next »

Download language pack

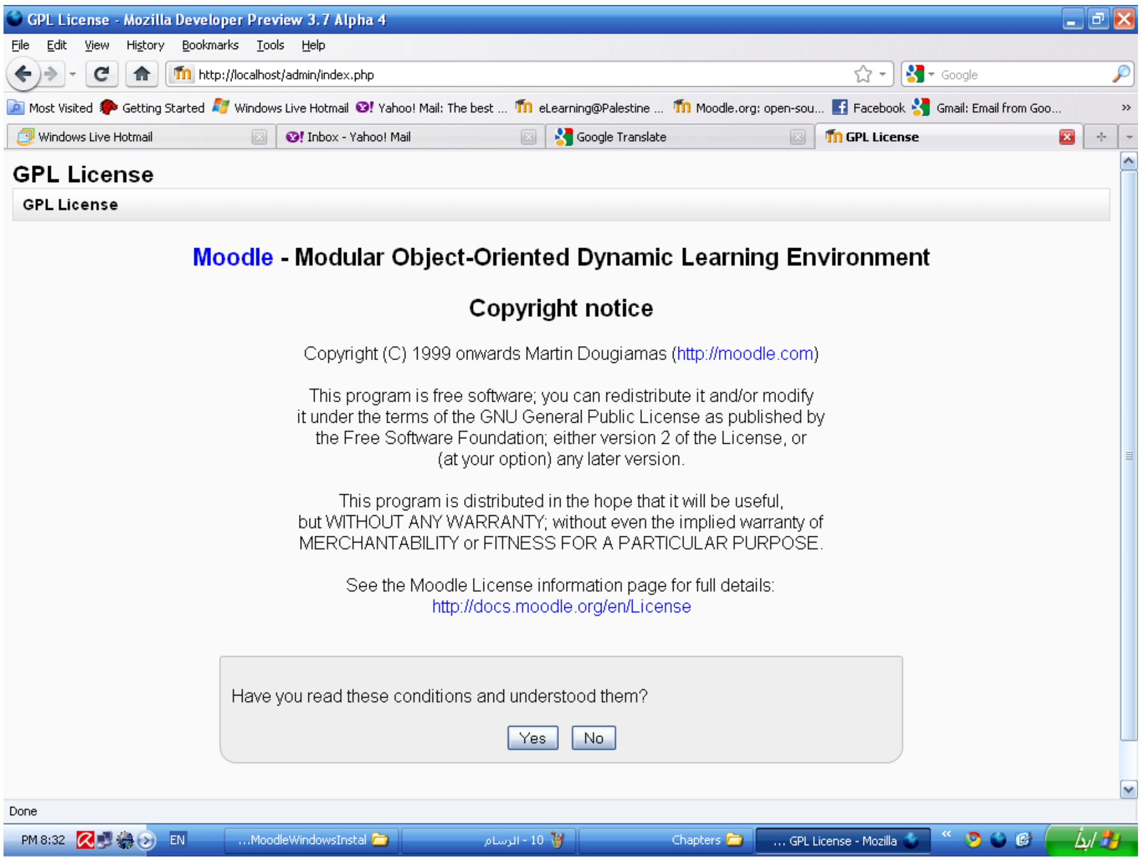
You now have the option of downloading a language pack and continuing the installation process in this language.

If you are unable to download the language pack, the installation process will continue in English. (Once the installation process is complete, you will have the opportunity to download and install additional language packs.)

You may continue the installation process using the default language pack, "English (en)".

Figure (31): Language Pack

- Next, the GPL license will appear as can be seen in Figure (32).


GPL License - Mozilla Developer Preview 3.7 Alpha 4
http://localhost/admin/index.php
GPL License
GPL License
Yes
No
Done

Moodle - Modular Object-Oriented Dynamic Learning Environment

Copyright notice

Copyright (C) 1999 onwards Martin Dougiamas (http://moodle.com)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

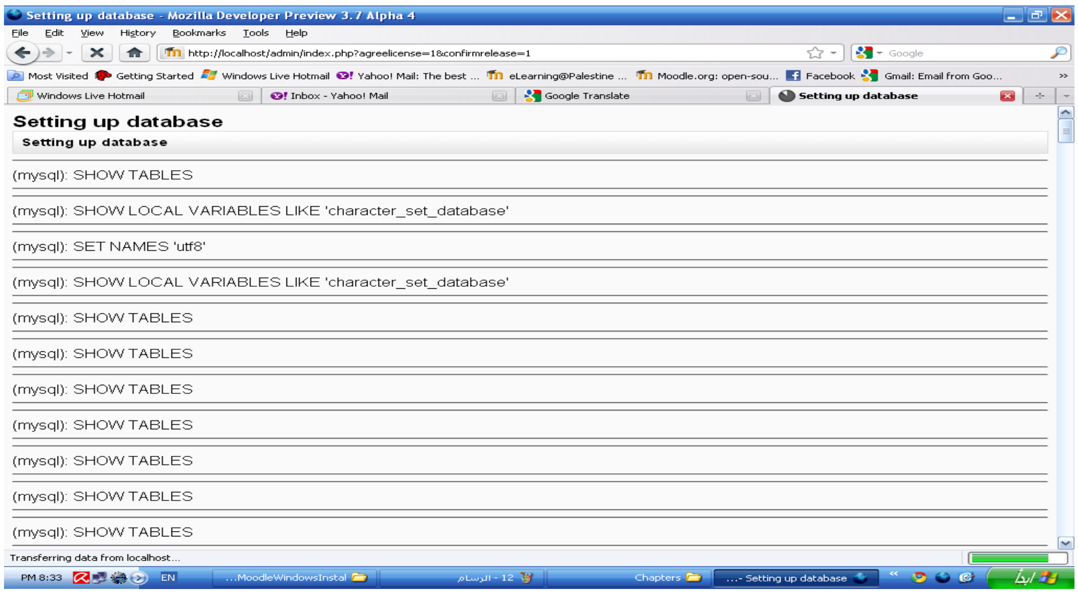
This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

See the Moodle License information page for full details: http://docs.moodle.org/en/License

Have you read these conditions and understood them?

Figure (32): GPL license

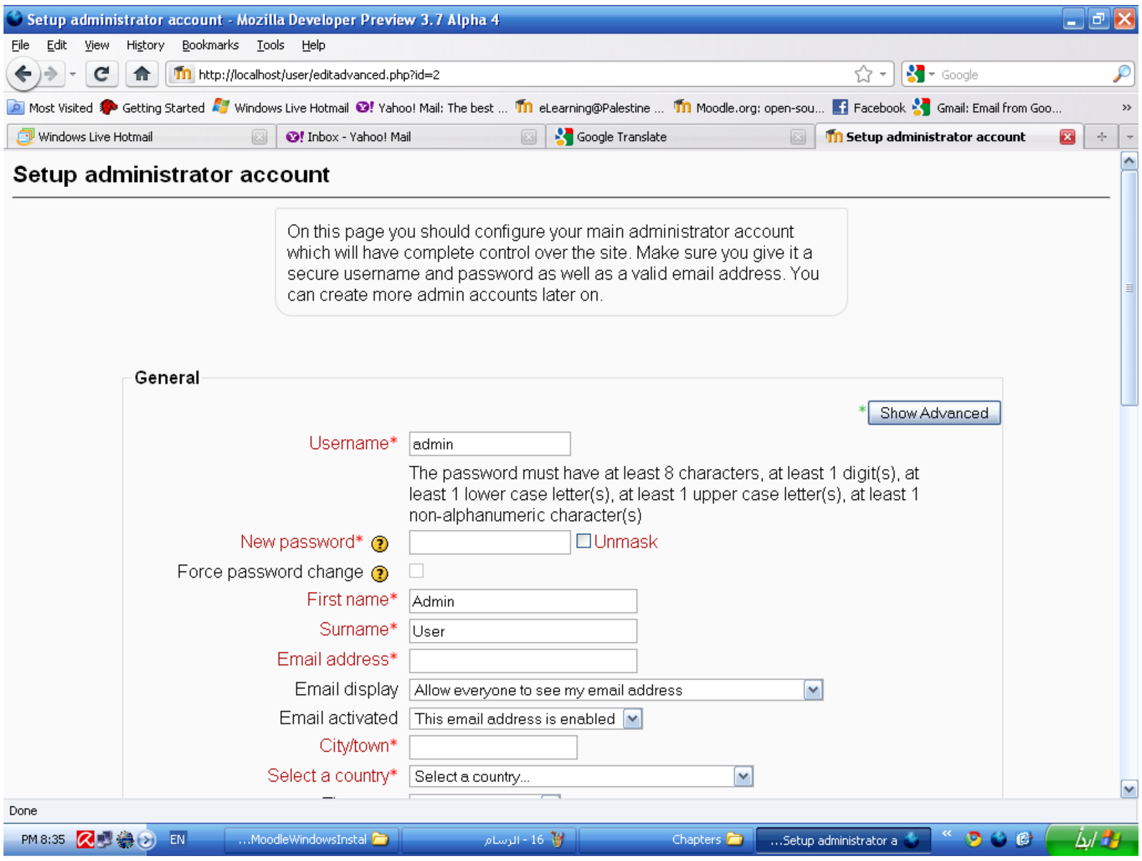
- The next step (Figure (33)) is to configure the database tables.


Setting up database - Mozilla Developer Preview 3.7 Alpha 4
http://localhost/admin/index.php?agreelicense=1&confirmrelease=1
Setting up database
Setting up database
Transferring data from localhost...

```
(mysql): SHOW TABLES
(mysql): SHOW LOCAL VARIABLES LIKE 'character_set_database'
(mysql): SET NAMES 'utf8'
(mysql): SHOW LOCAL VARIABLES LIKE 'character_set_database'
(mysql): SHOW TABLES
(mysql): SHOW TABLES
(mysql): SHOW TABLES
(mysql): SHOW TABLES
(mysql): SHOW TABLES
(mysql): SHOW TABLES
(mysql): SHOW TABLES
```

Figure (33): Database Tables Configuration

- Then configure an account for the main administrator who has a complete control over the site. You need to give him a secure username and password (see Figure (34)).


Setup administrator account - Mozilla Developer Preview 3.7 Alpha 4
File Edit View History Bookmarks Tools Help
http://localhost/user/editadvanced.php?id=2
Windows Live Hotmail
Inbox - Yahoo! Mail
Google Translate
Setup administrator account
Show Advanced
Done

Setup administrator account

On this page you should configure your main administrator account which will have complete control over the site. Make sure you give it a secure username and password as well as a valid email address. You can create more admin accounts later on.

General

Username*	admin
	The password must have at least 8 characters, at least 1 digit(s), at least 1 lower case letter(s), at least 1 upper case letter(s), at least 1 non-alphanumeric character(s)
New password*	☐ Unmask
Force password change	☐
First name*	Admin
Surname*	User
Email address*	
Email display	Allow everyone to see my email address
Email activated	This email address is enabled
City/town*	
Select a country*	Select a country...

Figure (34): Administrator Account

6.2 Server Information and Configuration

The server (hardware) is found and run at the Computer Center of Palestine Polytechnic University. Therefore, we do not need to configure it. Whenever the Computer Center wishes to deploy or test our system, they simply can host it as a new site on the server.

6.3 Deployment Tiers

6.3.1 Three-tier system

In this section, we display three-tier systems that contain clients specific to user (in this project users are teachers and students), internet server (s), and database server(s). A three-tier system design divides the business or organization objects and their operations or business rules from both the user interface, from the database, and from any other legacy applications that might be used.

In this project we need to apply it on Moodle in Palestine Polytechnic University, so this is three-tier system specific for Palestine Polytechnic University.

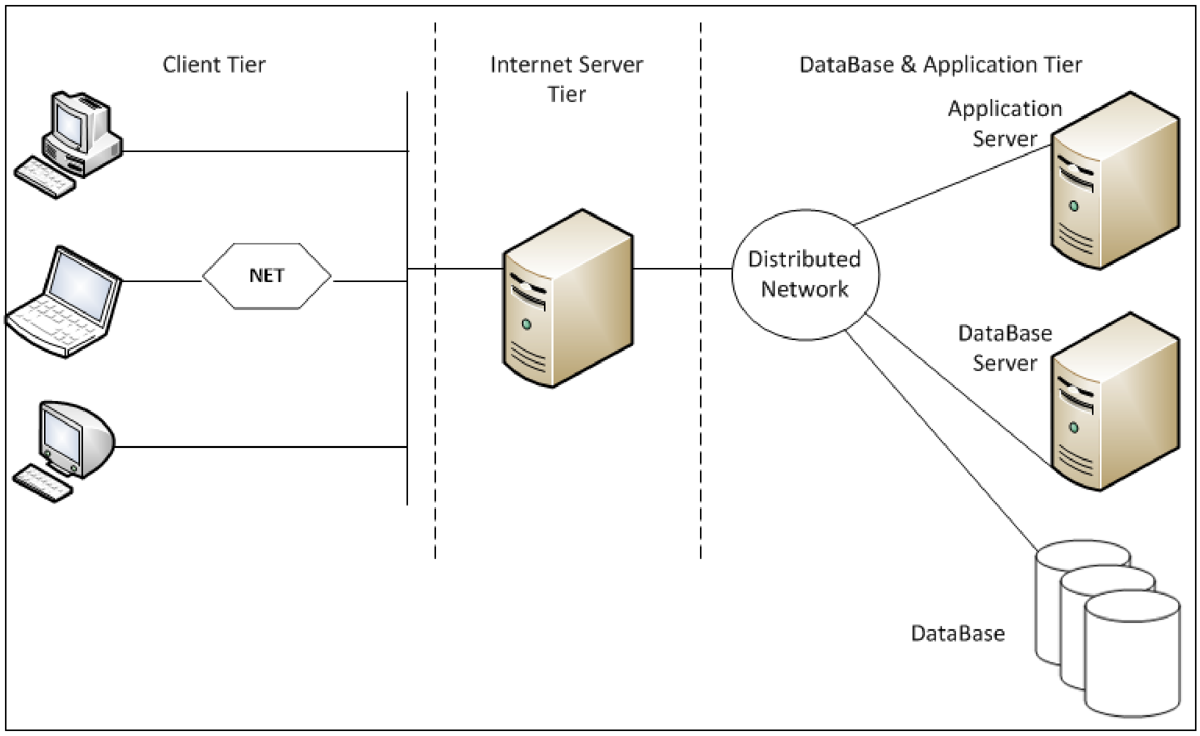


Figure (35): Three-Tier System[4]

6.3.2 Three-Tier Class Diagram

Before considering adding infrastructure classes to project class diagram, we might want to modify project class diagram by dividing it into three-tier system section to indicate which tier each class will reside in.

[4] The Computer Center in PPU

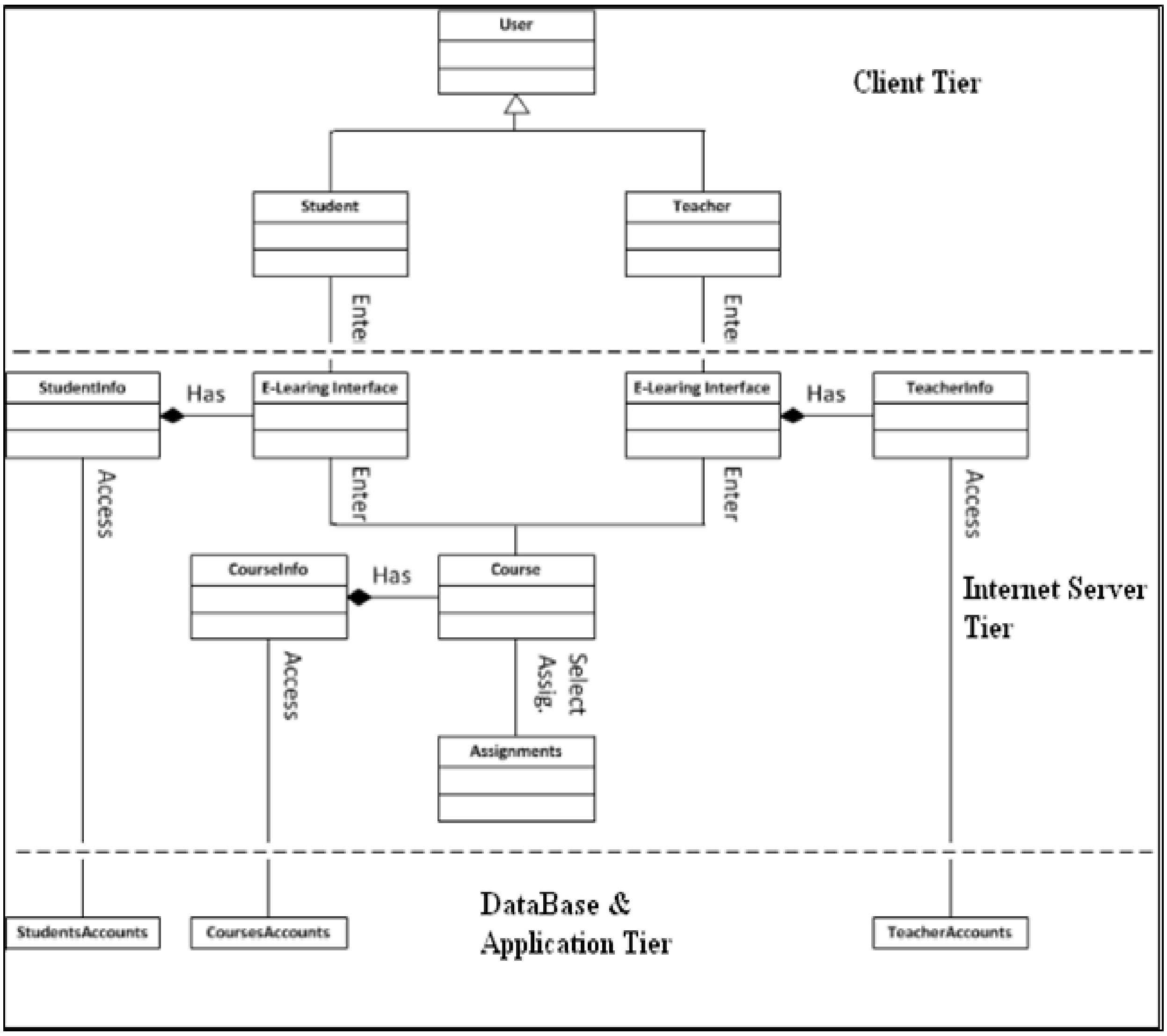


Figure (36): Three-Tier Class Diagram

Chapter Seven
Testing

In this chapter, we will test our system step-by-step (starting from the point when the teacher requests the main page of Moodle, going by intermediate steps such as course selection, assignment selection, plagiarism detection, to finally showing the plagiarism report). This test will be performing in order to ensure that the system meets its specifications and requirements. The testing phases that will be discussed in this chapter are:

1. System units and module testing.
2. Integration testing.
3. System testing.
4. Acceptance testing.

7.1 System Unit and Module Testing

The unit and module testing has been performed using the black box testing method; we suggest some possible problems and test the system performance against these problems.

Table (15): Shows Some Problems to be Testing on Teacher Object

Inputs	Expected values	Actual values	Notes
Valid Teacher Username and Invalid Password	Error Message	Error Message	Match
Invalid Teacher Username and Valid Password	Error Message	Error Message	Match
Invalid Teacher Username and Invalid Password	Error Message	Error Message	Match
Valid Teacher Username and Password	Open Main Page	Open Main Page	Match
Select Course That The Teacher Doesn’t Register In It	Open Enrollment Key Page	Open Enrollment Key Page	Match

Table (15): Cont.

Select Course That The Teacher Does Register In It	Open Course Main Page	Open Course Main Page	Match
Select The Assignment	Open Assignment Main Page	Open Assignment Main Page	Match
Select No Submitted Assignments	Error Message (No Submitted Assignments)	Error Message (No Submitted Assignments)	Match
Select # Submitted Assignments	Open The List Of Students Names Whose Submitted Assignment	Open The List Of Students Names Whose Submitted Assignment	Match
Check The Plagiarism Detection	View Plagiarism Report	View Plagiarism Report	Match

7.2 Integration Testing

All sub-units and sub-modules were integrated with each other and this integration is tested to show if there were defects that appear upon their integration. Testing here demonstrates on the interfaces among all sub-units of the system, and the functionality of the integrated sub-parts. After testing the integration of all sub-units of the system, the results indicated that all sub-units and sub-modules work together properly.

7.3 System Testing

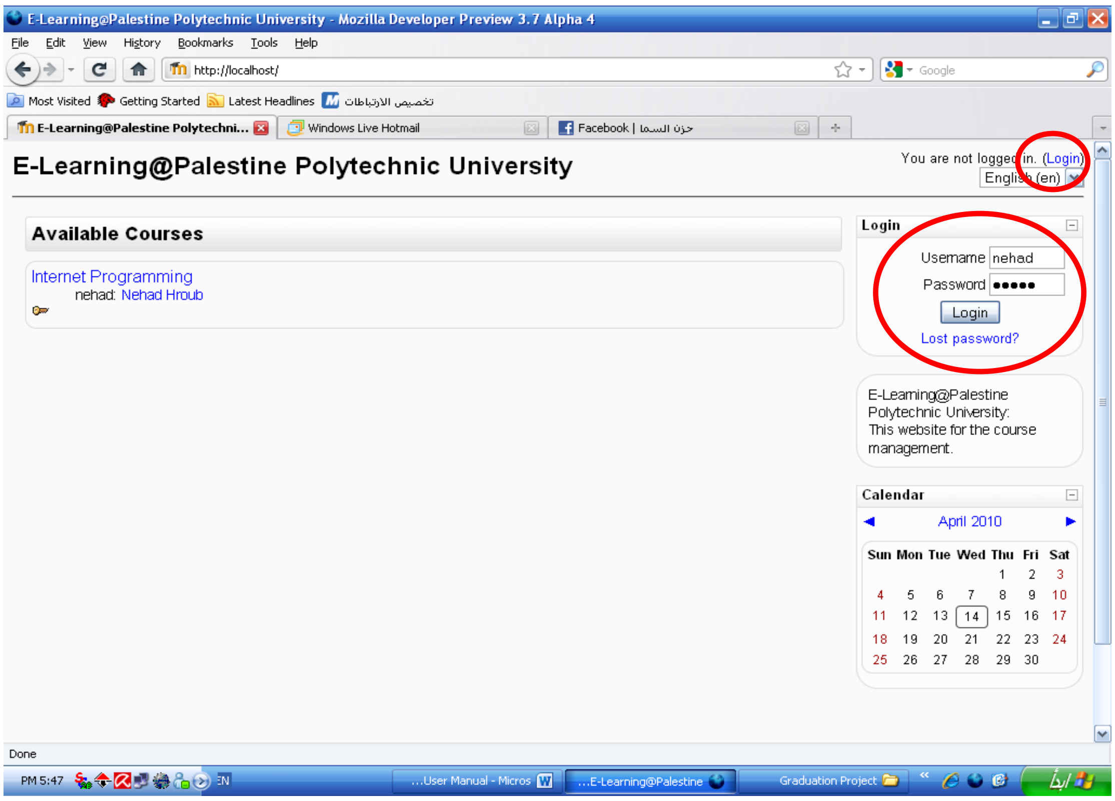
We have tested the system under several conditions and there were some errors and problems. Upon these results, all identified problems have been resolved.

7.4 Acceptance Testing

After testing the system against its requirements, we could determine that it achieves its functional requirements, and t is ready to be deployed in the real world (university). The system currently allows the teachers to perform plagiarism detection in rabid and easy way, and provides them with logical results.

7.5 Interface Testing

This section simulates (tests) how all system screens will be used by a teacher. The sequence and displayed results show that all main system screens are working as expected. First, the teacher enters the Moodle Website (http://elearning.ppu.edu/) and he will see the main page of Moodle that contains interactive objects such as the organization name (PPU in our case), the course names, and the website description. To enter the system, the teacher clicks the button, or enters his / her own username and password in the login form directly (see Figure (37)).


E-Learning@Palestine Polytechnic University - Mozilla Developer Preview 3.7 Alpha 4
File Edit View History Bookmarks Tools Help
http://localhost/
Google
Most Visited Getting Started Latest Headlines تخصيص الارتباطات
E-Learning@Palestine Polytechni...
Windows Live Hotmail
Facebook | حزن السما
Available Courses
Internet Programming
Login
Lost password?
Calendar
April 2010
Sun Mon Tue Wed Thu Fri Sat
Done
PM 5:47
...User Manual - Micros
...E-Learning@Palestine
Graduation Project

E-Learning@Palestine Polytechnic University

You are not logged in. (Login)

English (en)

nehad: Nehad Hroub

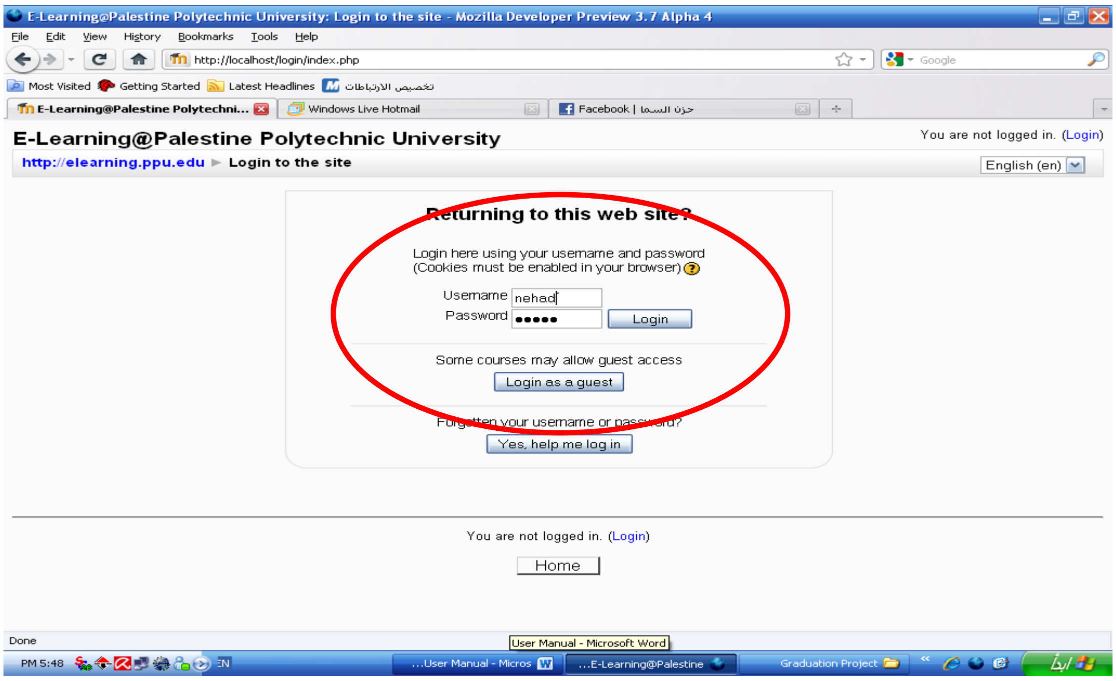
Username: nehad

Password: ●●●●●

E-Learning@Palestine Polytechnic University: This website for the course management.

Figure (37): Login Form

If he / she clicked on the button, he / she will be redirected to the login page; in this page, the teacher should enter his / her own username and password correctly.


E-Learning@Palestine Polytechnic University: Login to the site - Mozilla Developer Preview 3.7 Alpha 4

http://localhost/login/index.php

E-Learning@Palestine Polytechnic University

You are not logged in. (Login)

http://elearning.ppu.edu ► Login to the site

English (en)

Returning to this web site?

Login here using your username and password
(Cookies must be enabled in your browser)

Username nehad

Password •••••

Login

Some courses may allow guest access

Login as a guest

Forgotten your username or password?

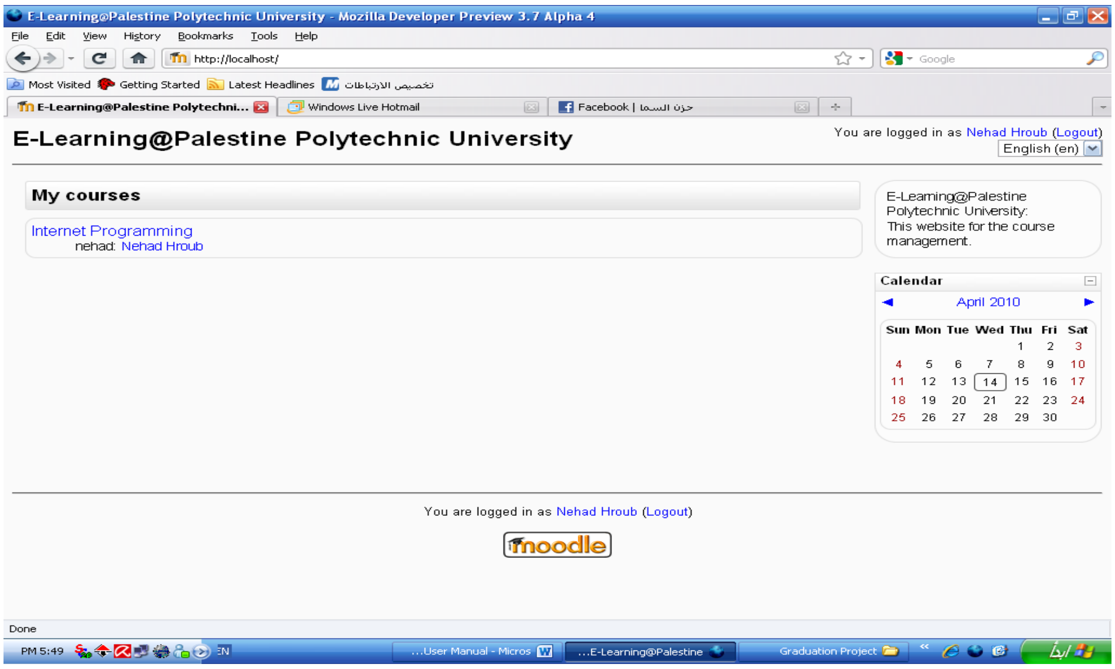
Yes, help me log in

You are not logged in. (Login)

Home

Figure (38): Login to the Site

If the entered username and password were correctly, Moodle will redirect the teacher to the main page which contains the names of courses that he / she is teaching, course description, and the name of teacher (teachers) that he/she (they) teaching each course as shown in the Figure (39).


E-Learning@Palestine Polytechnic University - Mozilla Developer Preview 3.7 Alpha 4

http://localhost/

E-Learning@Palestine Polytechnic University

You are logged in as Nehad Hroub (Logout)

English (en)

My courses

Internet Programming
nehad: Nehad Hroub

E-Learning@Palestine Polytechnic University: This website for the course management.

Calendar

April 2010

Sun	Mon	Tue	Wed	Thu	Fri	Sat
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

You are logged in as Nehad Hroub (Logout)

Figure (39): Main Page

When the teacher clicks on a course name, he / she will be redirected to the main page of the selected course.

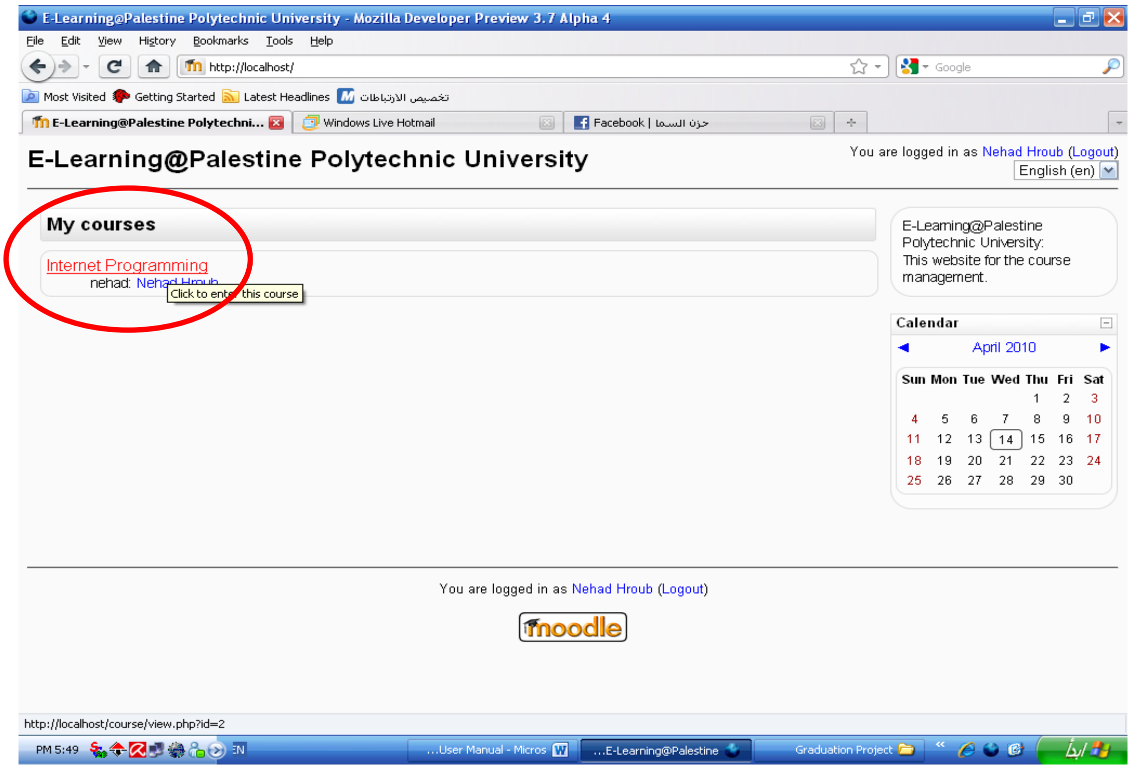

E-Learning@Palestine Polytechnic University - Mozilla Developer Preview 3.7 Alpha 4
File Edit View History Bookmarks Tools Help
http://localhost/
E-Learning@Palestine Polytechnic University
You are logged in as Nehad Hroub (Logout)
English (en)
My courses
Internet Programming
nehad: Nehad Hroub
Click to enter this course
E-Learning@Palestine Polytechnic University: This website for the course management.
Calendar
April 2010
http://localhost/course/view.php?id=2

Figure (40): Select a Course in the Main Page

After clicking on the name of the selected-course, the main page of that course will be displayed. This page contains course name, course participants, course assignments and other activities. The user then can click on the name of the available assignments (if any) as shown in Figure (41).

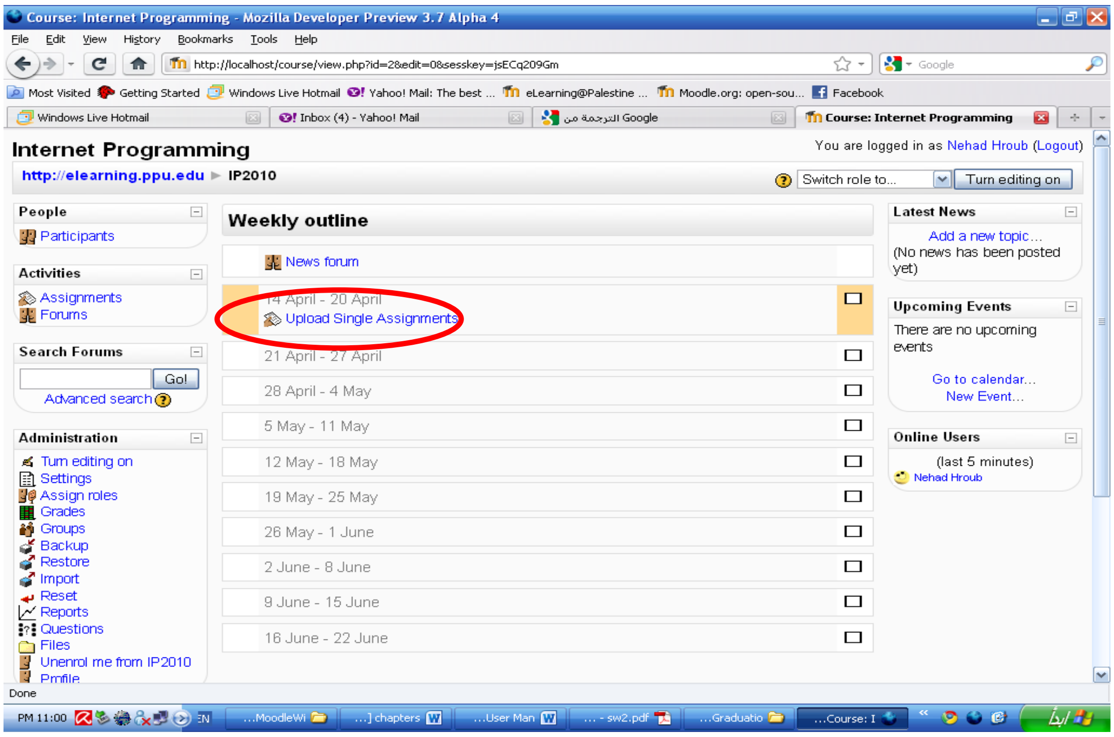
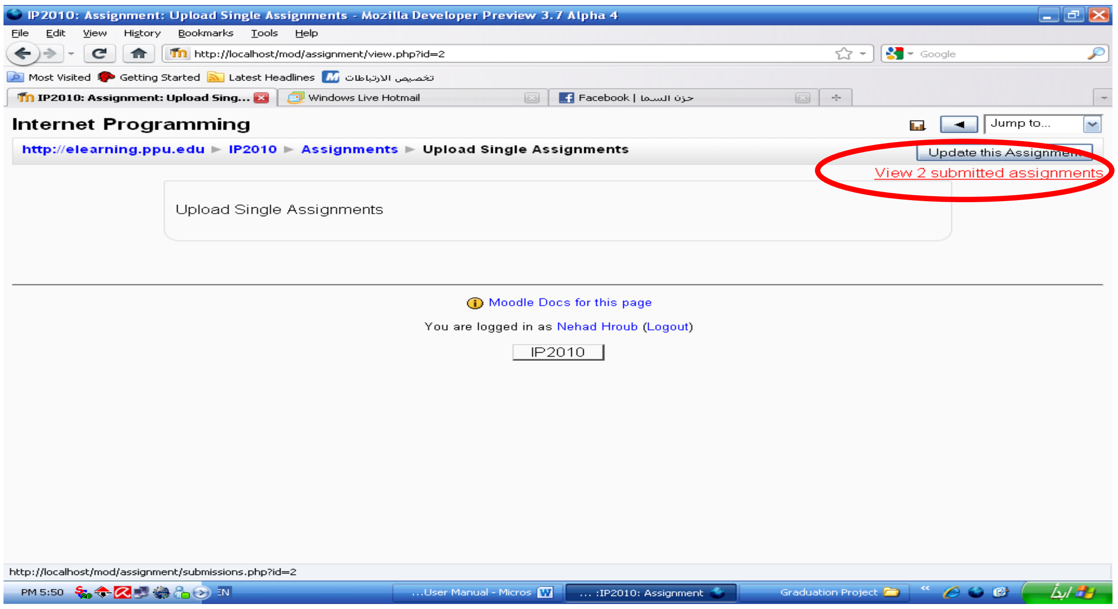


Figure (41): Select an Assignment

The assignment page contains assignment description, number of students who submitted assignments and the link to update this assignment, to view the submitted assignments you can click the link as can be seen in Figure (42).

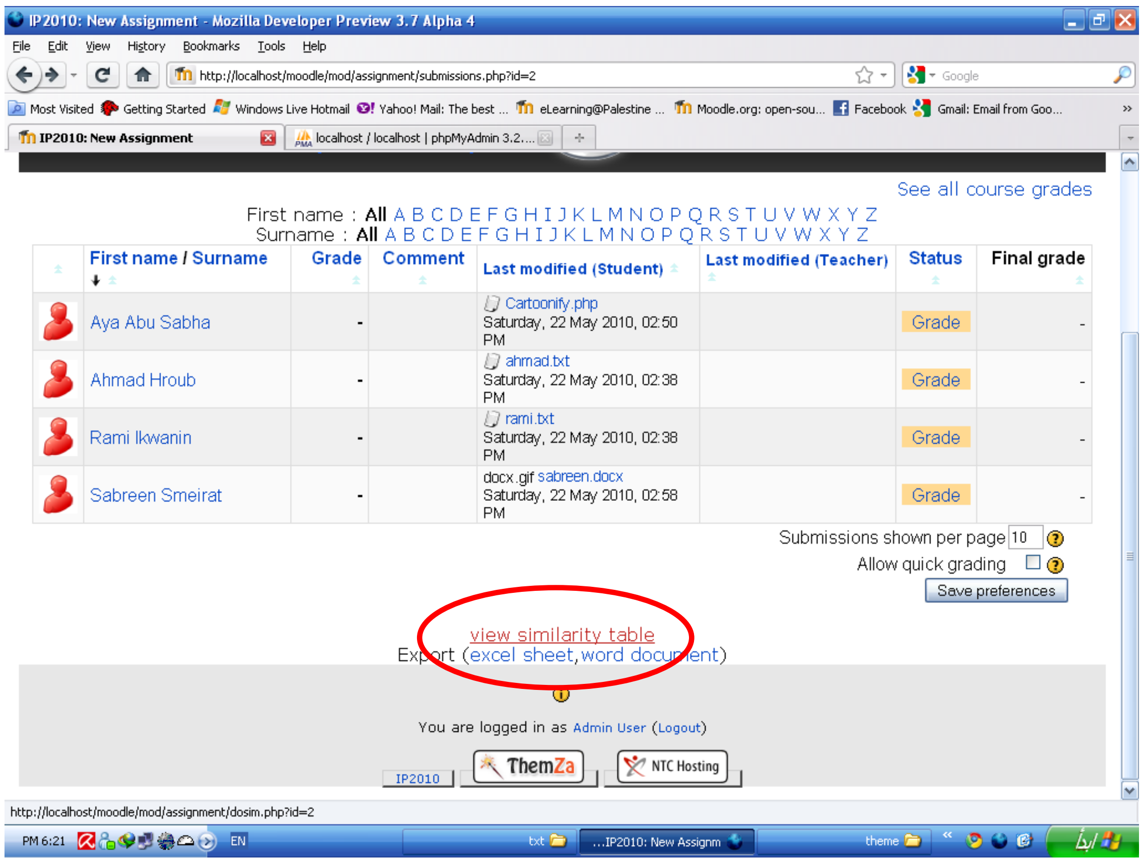


IP2010: Assignment: Upload Single Assignments - Mozilla Developer Preview 3.7 Alpha 4
http://localhost/mod/assignment/view.php?id=2
Internet Programming
Jump to...
http://elearning.ppu.edu ▶ IP2010 ▶ Assignments ▶ Upload Single Assignments
Update this Assignment
View 2 submitted assignments
Upload Single Assignments
Moodle Docs for this page
You are logged in as Nehad Hroub (Logout)
IP2010
http://localhost/mod/assignment/submissions.php?id=2

Figure (42): Assignment Page

After clicking on the link, the teacher will be redirected to the submission page that contains submitted assignments; this page contains the names of students who submitted the assignments and some other information.

The teacher can view these assignments, add comments, add the assignment grade, and update this assignment. When click on the link, the teacher can view the plagiarism report that compares the assignments with each other's.

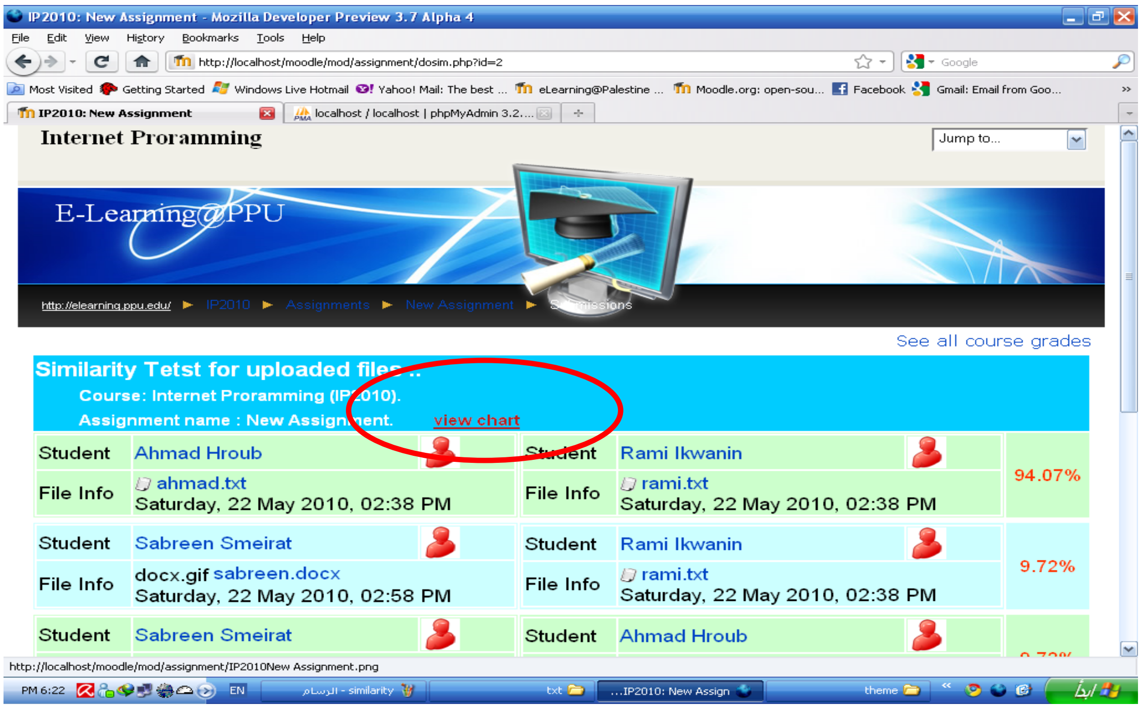

IP2010: New Assignment - Mozilla Developer Preview 3.7 Alpha 4
File Edit View History Bookmarks Tools Help
http://localhost/moodle/mod/assignment/submissions.php?id=2
Google
Most Visited
Getting Started
Windows Live Hotmail
Yahoo! Mail: The best ...
eLearning@Palestine ...
Moodle.org: open-sou...
Facebook
Gmail: Email from Goo...
IP2010: New Assignment
localhost / localhost | phpMyAdmin 3.2....
See all course grades
First name : All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Surname : All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

	First name / Surname	Grade	Comment	Last modified (Student)	Last modified (Teacher)	Status	Final grade
	Aya Abu Sabha	-		Cartoonify.php Saturday, 22 May 2010, 02:50 PM		Grade	-
	Ahmad Hroub	-		ahmad.txt Saturday, 22 May 2010, 02:38 PM		Grade	-
	Rami Ikwanin	-		rami.txt Saturday, 22 May 2010, 02:38 PM		Grade	-
	Sabreen Smeirat	-		docx.gif sabreen.docx Saturday, 22 May 2010, 02:58 PM		Grade	-

Submissions shown per page 10
Allow quick grading
Save preferences
view similarity table
Export (excel sheet, word document)
You are logged in as Admin User (Logout)
IP2010
ThemZa
NTC Hosting
http://localhost/moodle/mod/assignment/dosim.php?id=2
PM 6:21
EN
txt
...IP2010: New Assignm
theme

Figure (43): Submission Page

When the teacher clicks on the view similarity table, he/she will be redirected to the plagiarism form that shows the result of comparing between the submitted assignments as shown on the Figure (44).



IP2010: New Assignment - Mozilla Developer Preview 3.7 Alpha 4
File Edit View History Bookmarks Tools Help
http://localhost/moodle/mod/assignment/dosim.php?id=2
Internet Proramming
Jump to...
E-Learning@PPU
http://elearning.ppu.edu/ ► IP2010 ► Assignments ► New Assignment ► Submissions
See all course grades

Similarity Tetst for uploaded files ..

Course: Internet Proramming (IP2010).

Assignment name : New Assignment. view chart

Student	Ahmad Hroub	Student	Rami Ikwanin	94.07%
File Info	ahmad.txt Saturday, 22 May 2010, 02:38 PM	File Info	rami.txt Saturday, 22 May 2010, 02:38 PM	
Student	Sabreen Smeirat	Student	Rami Ikwanin	9.72%
File Info	docx.gif sabreen.docx Saturday, 22 May 2010, 02:58 PM	File Info	rami.txt Saturday, 22 May 2010, 02:38 PM	
Student	Sabreen Smeirat	Student	Ahmad Hroub	

http://localhost/moodle/mod/assignment/IP2010New Assignment.png

Figure (44): Similarity Page

The teacher can view the visual report when he/she click on the view chart link as shown on the previous Figure (45).

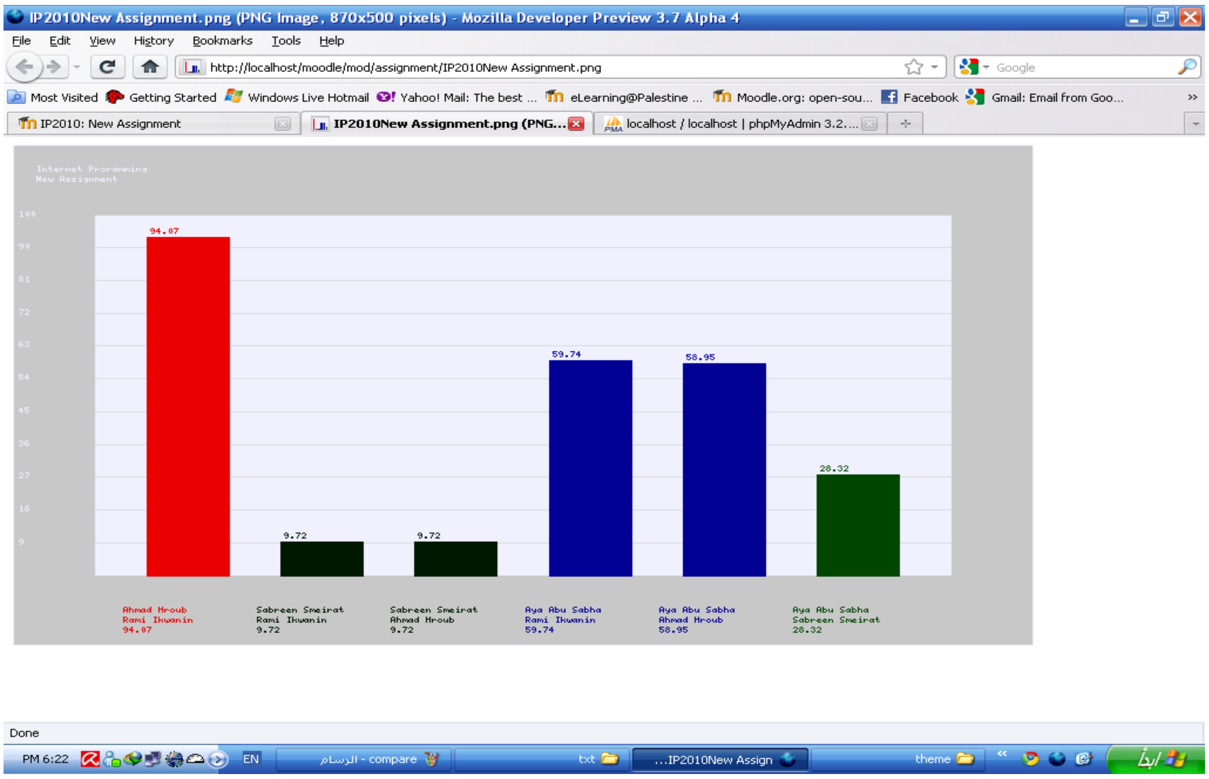
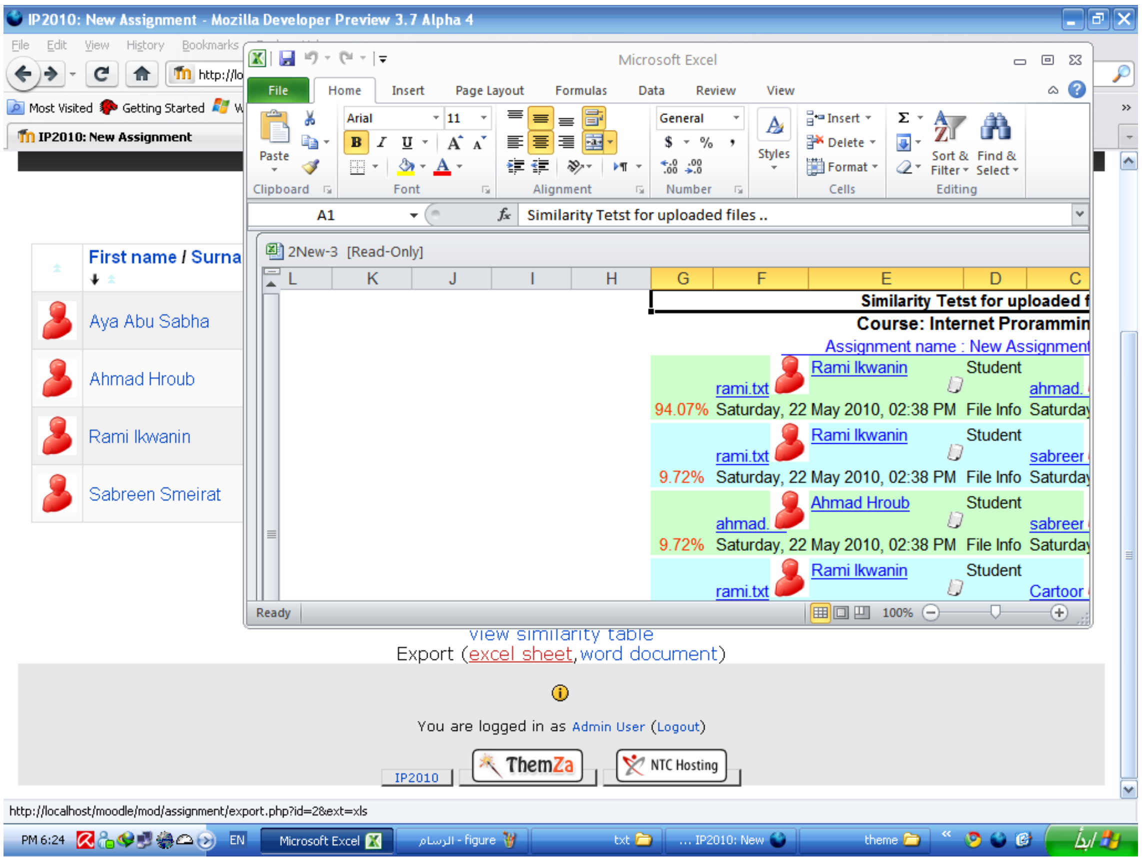


Figure (45): Visual Cheating Report

The teacher can export the results of the cheating report to an Excel sheet or Word document by clicking on the Excel sheet, Word document link as shown in Figure (46).


IP2010: New Assignment - Mozilla Developer Preview 3.7 Alpha 4

Microsoft Excel

A1 Similarity Tetst for uploaded files ..

2New-3 [Read-Only]

G	F	E	D	C
		Similarity Tetst for uploaded f		
		Course: Internet Prorammin		
		Assignment name : New Assignment		
	rami.txt	Rami Ikwanin	Student	ahmad.
94.07%	Saturday, 22 May 2010, 02:38 PM		File Info	Saturday
	rami.txt	Rami Ikwanin	Student	sabreen
9.72%	Saturday, 22 May 2010, 02:38 PM		File Info	Saturday
	ahmad.	Ahmad Hroub	Student	sabreen
9.72%	Saturday, 22 May 2010, 02:38 PM		File Info	Saturday
	rami.txt	Rami Ikwanin	Student	Cartoon

First name / Surna
Aya Abu Sabha
Ahmad Hroub
Rami Ikwanin
Sabreen Smeirat

view similarity table
Export (excel sheet, word document)

You are logged in as Admin User (Logout)

http://localhost/moodle/mod/assignment/export.php?id=2&ext=xls

File Edit View History Bookmarks
Most Visited
Getting Started
File Home Insert Page Layout Formulas Data Review View
Paste
Clipboard Font Alignment Number Cells Editing
Arial 11
General
Styles
Insert Delete Format
Sort & Filter
Find & Select
L K J I H
Ready
100%
IP2010
ThemZa
NTC Hosting
PM 6:24 EN Microsoft Excel
IP2010: New
theme

Figure (46): Save as Excel Sheet or Word Document

Chapter Eight

Conclusions

This system aimed to upgrade Moodle (the well-known open-source e-learning system). The presented system allows the teachers to detect the plagiarism among student's submitted assignments. This functionality has been successfully embedded into Moodle, and it is now ready to use by university teachers.

8.1 Conclusions

1. The system upgrades on Moodle and its open-source.
2. The plagiarism detection functionality of the system is based on a well-known algorithm in this field called AC.
3. The system compares among student's submitted-assignments and gives the rate of convergence among them.
4. The system generates cheating reports in both tabular and bar chart formats. It also enables the teacher to export cheating results to Excel spreadsheets or Word documents.
5. The system currently supports limited types of files; for the natural languages, it supports ***.text***, ***.doc***, and ***.docx*** files. For source code assignments, the system supports ***.php***, ***.C***, ***.C++***, and ***.java*** files.

8.2 Directions for Future Work

The following are some possible directions for future on the system:

1. Comparing among assignments containing compressed files and figures.
2. Comparing the student's submitted-assignments with other documents published on the web.
3. Add more intelligent techniques to the system in order to detect some advanced cheating tricks.

References

[1] Sanjay Goel, Deepak Rao et al.: Plagiarism and its Detection in Programming Languages, December 15, 2005.

[2] Sanjay Goel, Deepak Rao et al.: Plagiarism and its Detection in Programming Languages, December 15, 2005.

[3] http://www.php.net , (accessed: 23/10/2009)

[4] http://docs.moodle.org/en/Talk:About_Moodle, (accessed: 23/10/2009)

[5] Manuel Freire, Manuel Cebrian and Emilio del Rosal: *An Integrated Source Code Plagiarism Detection Environment*, Escuela Politecnica Superior, Universidad Autonoma de Madrid, 28049 Madrid, Spain.

[6] Shared Information and Program Plagiarism Detection, Xin Chen, Brent Francia, Ming Li, Brian Mckinnon, Amit Seker, University of California, Santa Barbara, December 13, 2003.

[7] Paul Clough: *Plagiarism in natural and programming languages: an overview of current tools and technologies*, Department of Computer Science, University of Sheffield, July 2000.

[8] *JPlag: Finding plagiarisms among a set of programs*, University at Karlsruhe D-76128 Karlsruhe, Germany

[9] Manuel Freire, Manuel Cebrifn and Emilio del Rosal Escuela: *AC: An Integrated Source Code Plagiarism Detection Environment*, Polit_ecnica Superior, Universidad Aut_onoma de Madrid.

[10] http://software.bioinformatics.uwaterloo.ca/SID/servlets/IndexPage , (accessed: 18/11/2009)

Figure References

- **Figure (1):**

  http://www.cis.famu.edu/~ejones/papers/stephens-plagiarism-ccscsc.pdf, (accessed: 23/11/2009)

- **Figure(2):**

  http://en.wikipedia.org/wiki/Parsing, (accessed: 20/11/2009)

- **Figure (3):**

  Xin Chen, Brent Francia, Ming Li, Brian Mckinnon, and Amit Seker: *Shared Information and Program Plagiarism Detection*, University of California, Santa Barbara, December 13, 2003.

Appendices

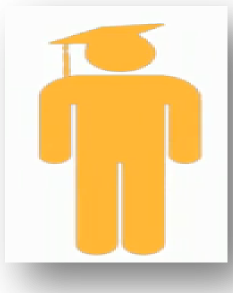
Appendix (A): User manual

Appendix (B): Part of Moodle Database Schema

Appendix (A)

User Manual

User Sub-Manual For Using Plagiarism Detection System[*]



Moodle Version: 1.9.8

June 2010

Plagiarism Detection System is an open-source software upgrading on Moodle to detect the plagiarism cases among assignments that student's submitted on Moodle and generates plagiarism report the plagiarism cases.

[*]This user sub-manual is for our project only, and we aim to integrate it to the original user manual of Moodle once we integrate our system with Moodle.

1.1 Introduction

Plagiarism can be defined as the use or close imitation of the language and thoughts of another author and the representation of them as one's own original work**.**

Many students make (intentionally or unintentionally) some type of cheating and plagiarism in their assignments. Usually it is difficult for teachers to detect plagiarism in student's assignments by hand. The detection process becomes easier, faster, and more efficient if it has performed automatically (i.e. by a computerized system).

The Plagiarism Detection is the process of locating instance of plagiarism within a document withers its text or code. Detection can be either manual or computer-assisted. Manual detection requires effort, memory and it is impractical in cases where many documents to be compared.

So, in this project we will applied this system on Moodle, this is Moodle using in E-Learning at Palestine Polytechnic University (PPU), It has become very popular among educators around the world as a tool for creating online dynamic web sites for their students. To work, it needs to be installed on a web server somewhere, either on one of your own computers or one at a web hosting company, and it is open source software package for producing Internet-based courses and web sites. It is a global development project designed to support a social constructionist framework of education.

Moodle is provided freely as Open Source software under the GNU General Public License (GPL) Version 2, June 1991. Basically this means Moodle is copyrighted, but that you have additional freedoms. You are allowed to copy, use, and modify Moodle.

Plagiarism Detection System is open-source software as an upgrade on Moodle; it aims to detect the plagiarism cases among student's submitted assignments and report the plagiarism cases.

1.2 System Features

The system presents open-source software to be integrated with an existing open-source e-learning platform (namely: Moodle). The user does not need to know how this system works; him / her only needs to know how to use Moodle as a teacher.

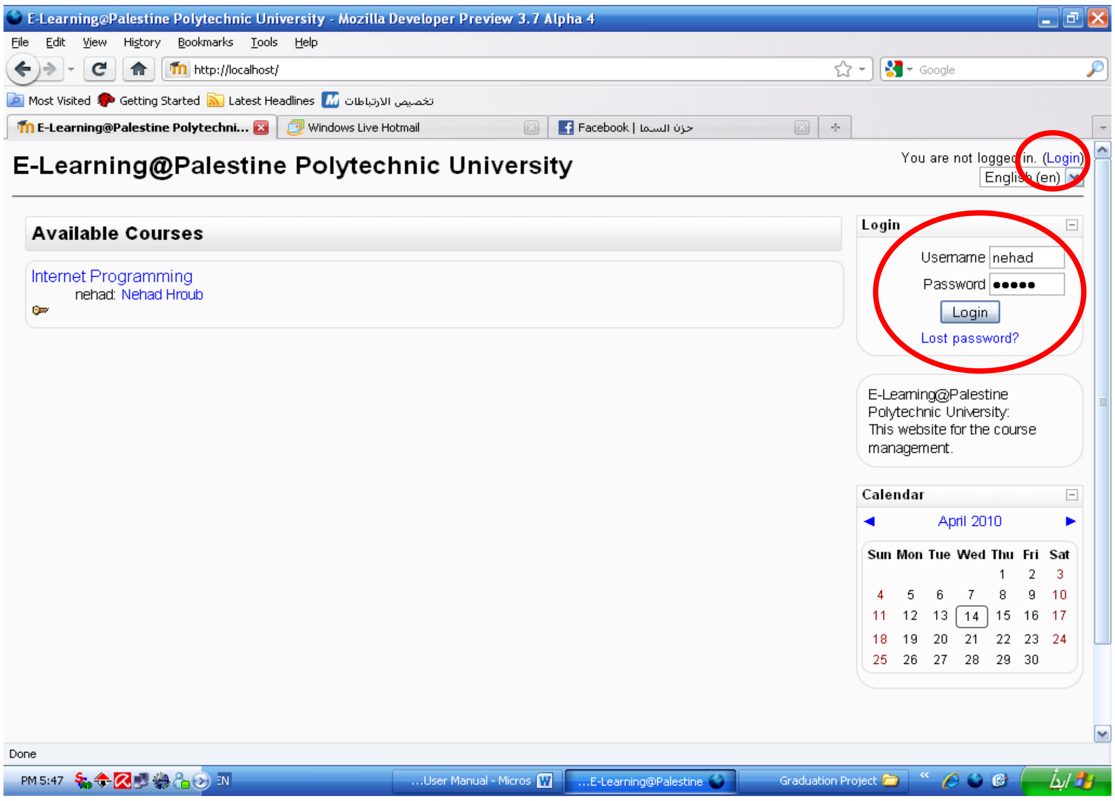
The software has the following features:

1. It is open-source and free.
2. It has a user-friendly interface by using interactive objects, easy to reading and understanding report, easy to navigate, and speed in viewing results.
3. It provides the plagiarism detection functionality.

1.3 Get Start

For using the Plagiarism Detection System, follow the instructions below:

1. When you request the Moodle Website (http://elearning.ppu.edu/) you will see the main page of the Moodle, it is containing interactive objects such as the organization name that using the Moodle, the courses names, and the website description. To enter the system please click on the button, or please enter your own username and password in the login form directly as shown in Figure (1).


E-Learning@Palestine Polytechnic University - Mozilla Developer Preview 3.7 Alpha 4
File Edit View History Bookmarks Tools Help
http://localhost/
Google
Most Visited Getting Started Latest Headlines تخصيص الارتباطات
E-Learning@Palestine Polytechni...
Windows Live Hotmail
Facebook | حزن السما
Done
PM 5:47
...User Manual - Micros
...E-Learning@Palestine
Graduation Project

E-Learning@Palestine Polytechnic University

You are not logged in. (Login)

English (en)

Available Courses

Internet Programming

nehad: Nehad Hroub

Login

Username	nehad
Password	●●●●●

Login

Lost password?

E-Learning@Palestine Polytechnic University: This website for the course management.

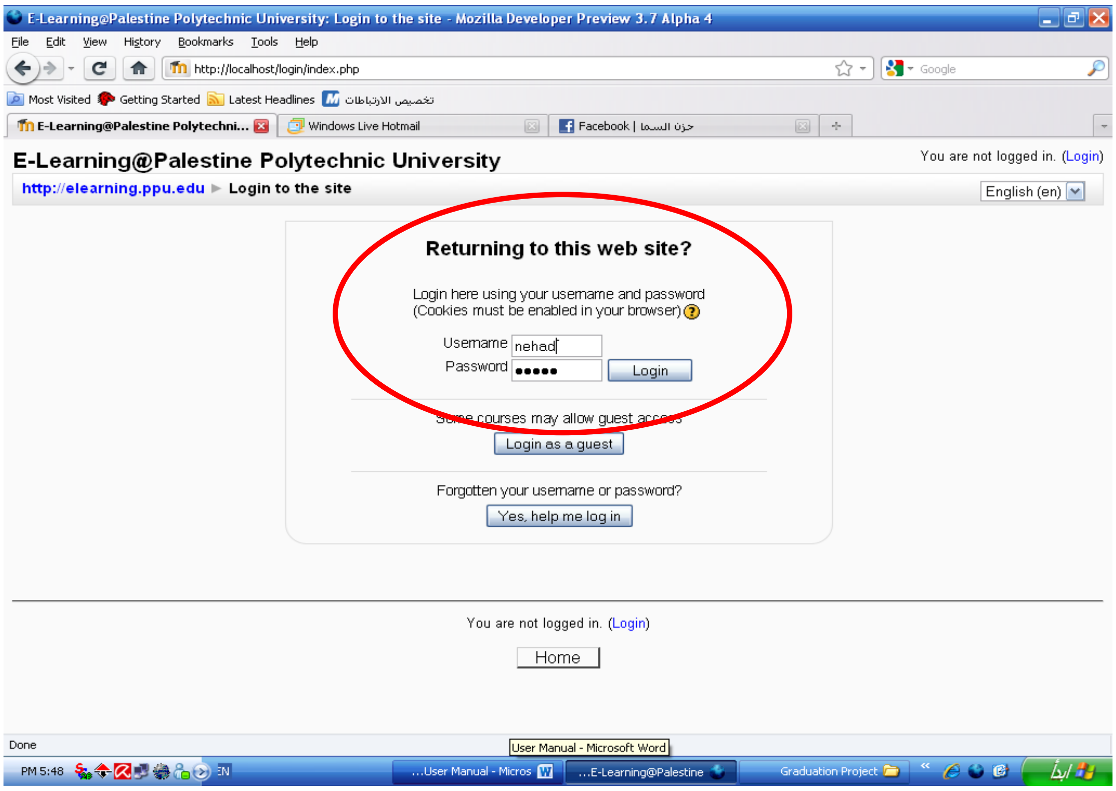
Calendar

April 2010

Sun	Mon	Tue	Wed	Thu	Fri	Sat
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

Figure (1): Login Form

2. If you click on the button, you show the login page, please enter your own username and password correctly in spatial place as shown in the Figure (2).


E-Learning@Palestine Polytechnic University: Login to the site - Mozilla Developer Preview 3.7 Alpha 4
File Edit View History Bookmarks Tools Help
http://localhost/login/index.php
Google
Most Visited Getting Started Latest Headlines تخصيص الارتباطات
E-Learning@Palestine Polytechni...
Windows Live Hotmail
Facebook | حزن السما
Done
User Manual - Microsoft Word
PM 5:48
...User Manual - Micros
...E-Learning@Palestine
Graduation Project

E-Learning@Palestine Polytechnic University

You are not logged in. (Login)

http://elearning.ppu.edu ▶ Login to the site

English (en)

Returning to this web site?

Login here using your username and password
(Cookies must be enabled in your browser)

Username	nehad
Password	●●●●●

Login

Some courses may allow guest access

Login as a guest

Forgotten your username or password?

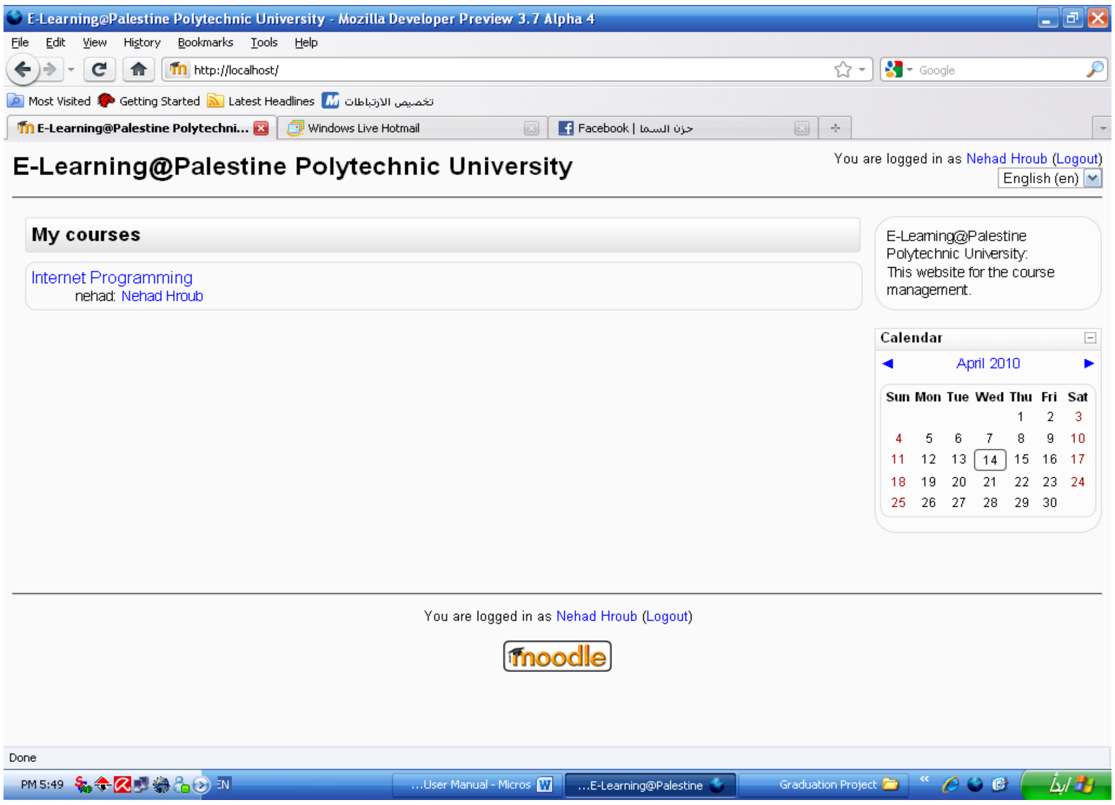
Yes, help me log in

You are not logged in. (Login)

Home

Figure (2): Login to the Site

3. After you have entered your own username and password correctly, the Moodle will redirect you to the main page, which contains the names of courses that you are teaching, course description, and the name of teacher (teachers) that he/she (they) teaching each course as shown in the Figure (3).



E-Learning@Palestine Polytechnic University

You are logged in as Nehad Hroub (Logout)

English (en)

My courses

Internet Programming
nehad: Nehad Hroub

E-Learning@Palestine Polytechnic University:
This website for the course management.

Calendar

April 2010

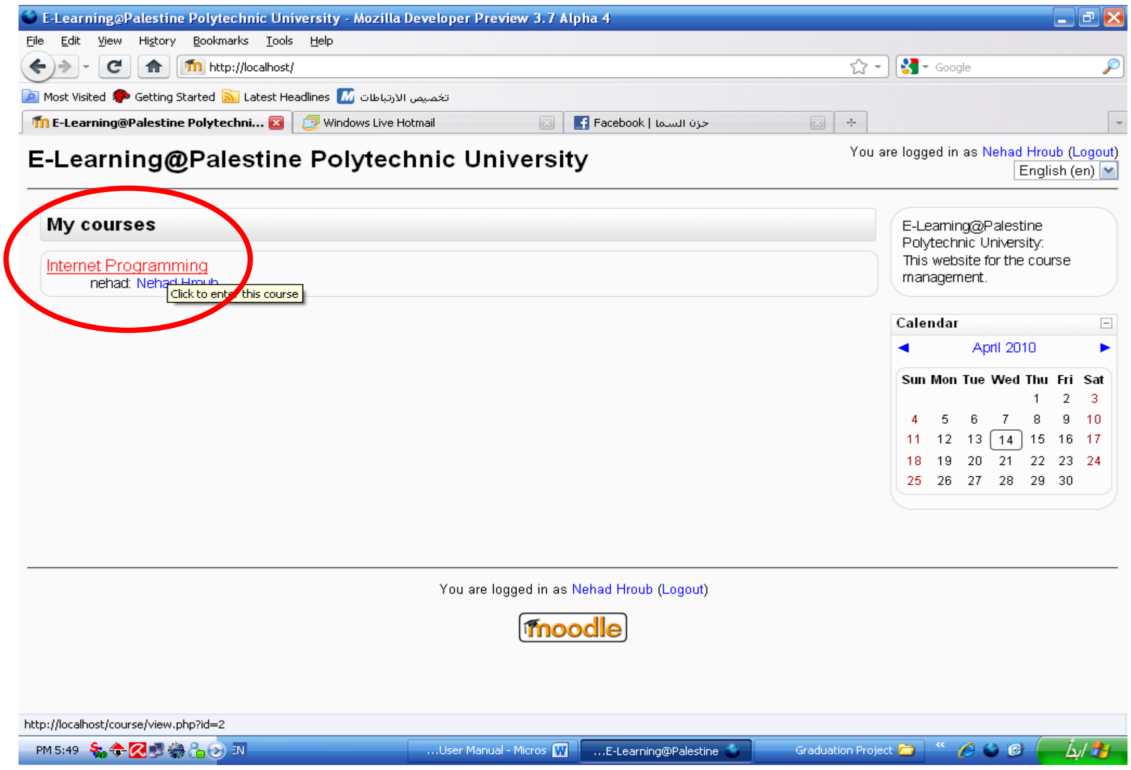
Sun	Mon	Tue	Wed	Thu	Fri	Sat
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

You are logged in as Nehad Hroub (Logout)

moodle

Figure (3): Main Page

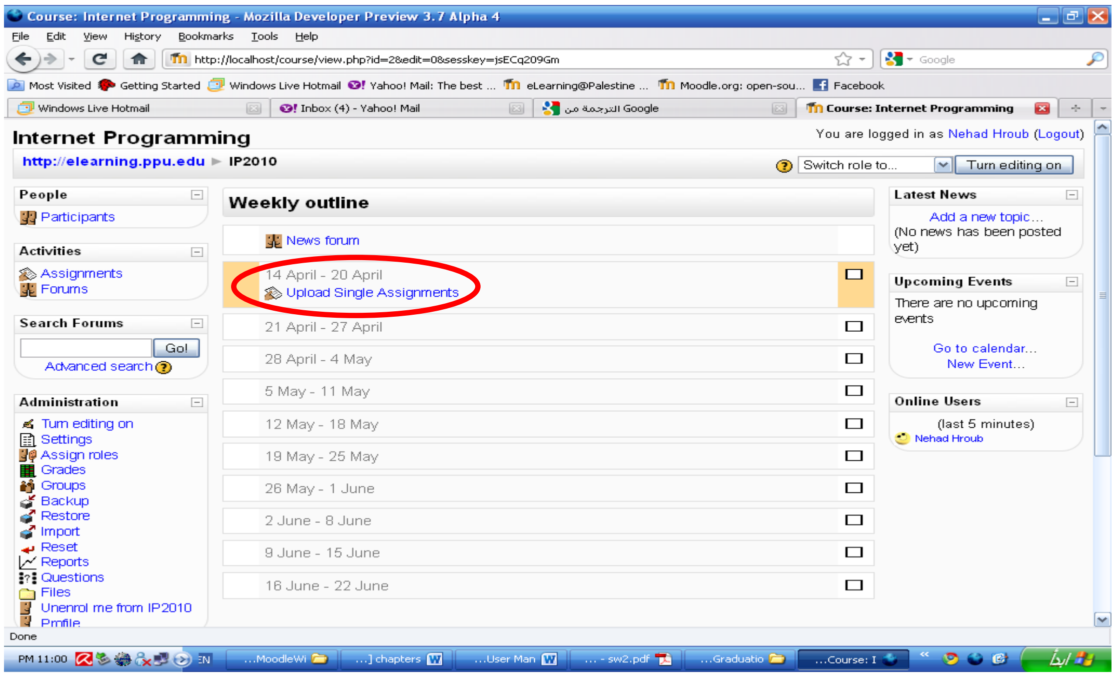
4. Please click on the name of the course that you want, this will redirect you to the main page for this course as shown in the Figure (4).


E-Learning@Palestine Polytechnic University - Mozilla Developer Preview 3.7 Alpha 4
http://localhost/
E-Learning@Palestine Polytechnic University
You are logged in as Nehad Hroub (Logout)
English (en)
My courses
Internet Programming
nehad: Nehad Hroub
Click to enter this course
Calendar
April 2010
http://localhost/course/view.php?id=2

E-Learning@Palestine Polytechnic University: This website for the course management.

Figure (4): Select the Course

5. After clicking on the name of the selected-course, you will see the main page of the selected-course. It contains course name, participants in the course, course assignments and other activities. Please click on the name of the wanted-assignment as shown in the Figure (5).

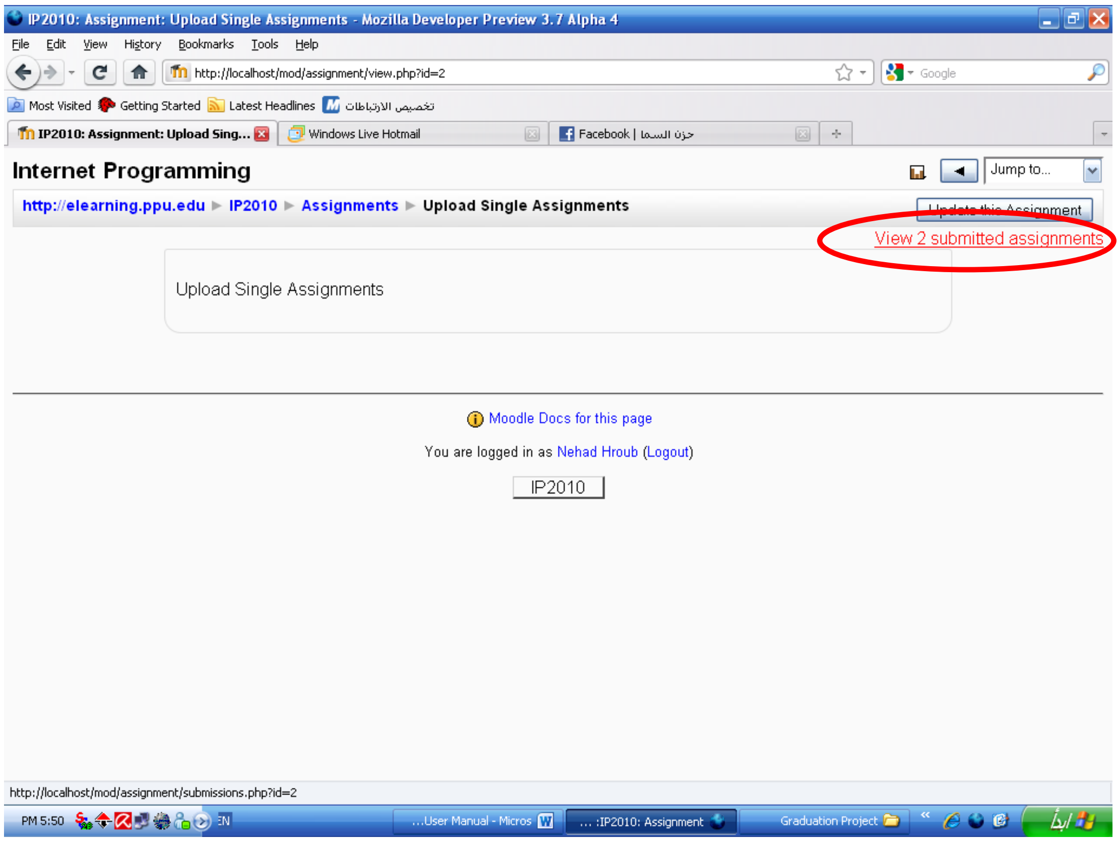


(No news has been posted yet)

There are no upcoming events

Figure (5): Select the Assignment

6. View the main page of the assignment, this page contains assignment description, number of students whose submitted-assignments, and button, to view the submitted assignments please click on as you see in Figure (6).


IP2010: Assignment: Upload Single Assignments - Mozilla Developer Preview 3.7 Alpha 4
File Edit View History Bookmarks Tools Help
http://localhost/mod/assignment/view.php?id=2
Most Visited Getting Started Latest Headlines
IP2010: Assignment: Upload Sing...
Windows Live Hotmail
Facebook
Jump to...
Update this Assignment
http://localhost/mod/assignment/submissions.php?id=2

Internet Programming

http://elearning.ppu.edu ▶ IP2010 ▶ Assignments ▶ Upload Single Assignments

View 2 submitted assignments

Upload Single Assignments

Moodle Docs for this page

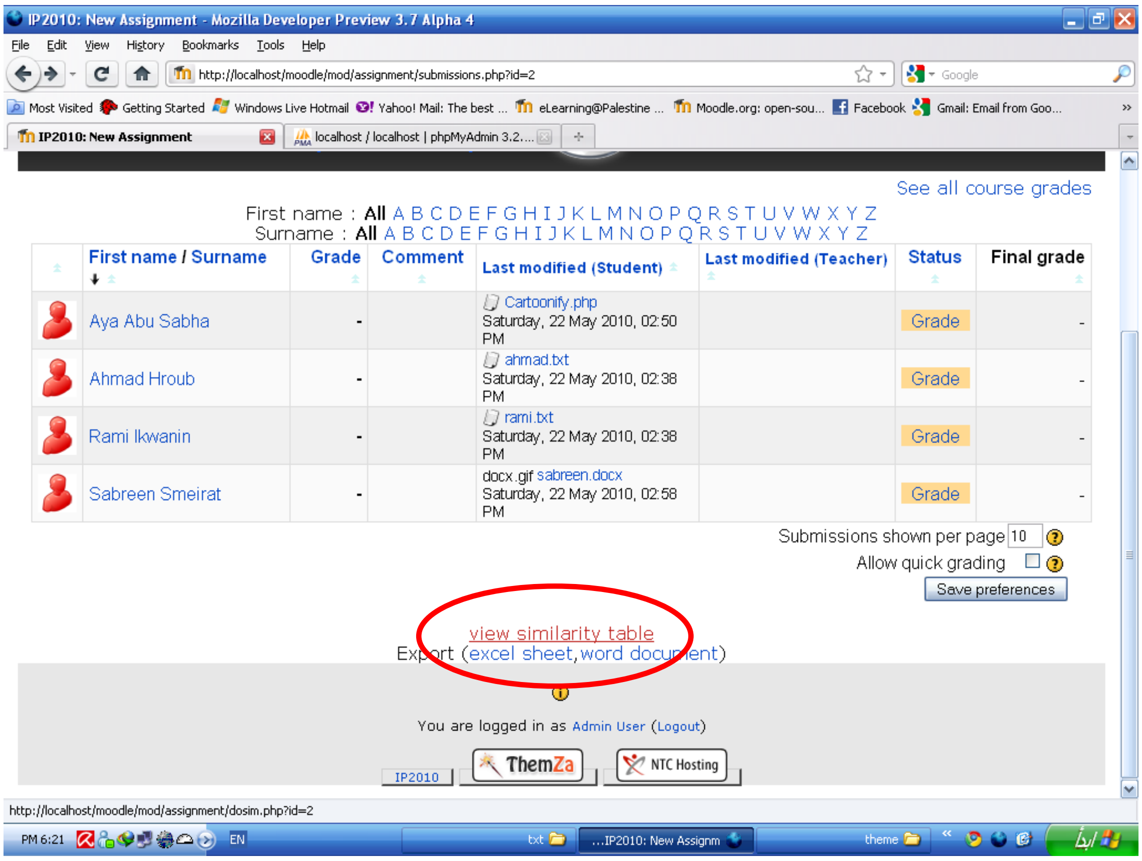
You are logged in as Nehad Hroub (Logout)

IP2010

Figure (6): The Main Page of the Assignment

7. After clicking on the link, the teacher will be redirected to the submission page that contains submitted assignments; this page contains the names of students who submitted the assignments and some other information.

The teacher can view these assignments, add comments, add the assignment grade, and update this assignment. When click on the link, the teacher can view the plagiarism report that compares the assignments with each others.



IP2010: New Assignment - Mozilla Developer Preview 3.7 Alpha 4

http://localhost/moodle/mod/assignment/submissions.php?id=2

See all course grades

First name : All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Surname : All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

	First name / Surname	Grade	Comment	Last modified (Student)	Last modified (Teacher)	Status	Final grade
	Aya Abu Sabha	-		Cartoonify.php Saturday, 22 May 2010, 02:50 PM		Grade	-
	Ahmad Hroub	-		ahmad.txt Saturday, 22 May 2010, 02:38 PM		Grade	-
	Rami Ikwanin	-		rami.txt Saturday, 22 May 2010, 02:38 PM		Grade	-
	Sabreen Smeirat	-		docx.gif sabreen.docx Saturday, 22 May 2010, 02:58 PM		Grade	-

Submissions shown per page 10
Allow quick grading
Save preferences

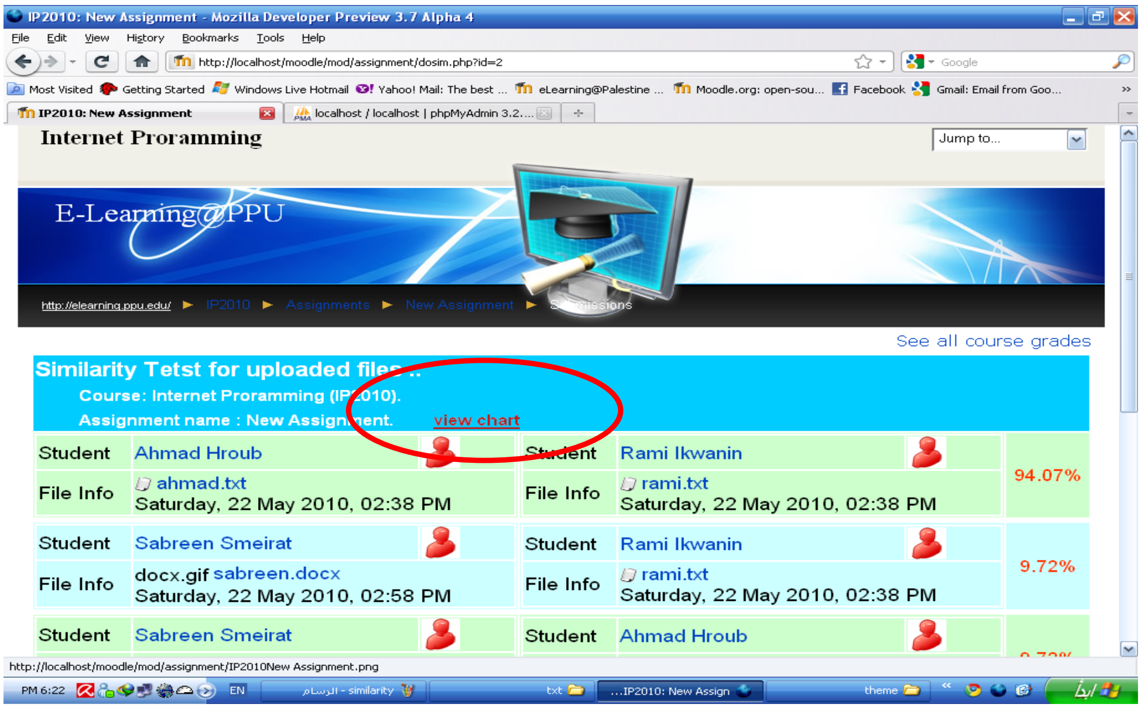
view similarity table
Export (excel sheet, word document)

You are logged in as Admin User (Logout)

http://localhost/moodle/mod/assignment/dosim.php?id=2

Figure (7): Submission Page

8. When the teacher clicks on the view similarity table, he / she will be redirected to the plagiarism form that shows the result of comparing between the submitted assignments as shown on the Figure (8).



IP2010: New Assignment - Mozilla Developer Preview 3.7 Alpha 4

http://localhost/moodle/mod/assignment/dosim.php?id=2

Internet Proramming

E-Learning@PPU

http://elearning.ppu.edu/ ► IP2010 ► Assignments ► New Assignment ► Submissions

See all course grades

Similarity Tetst for uploaded files ..

Course: Internet Proramming (IP2010).

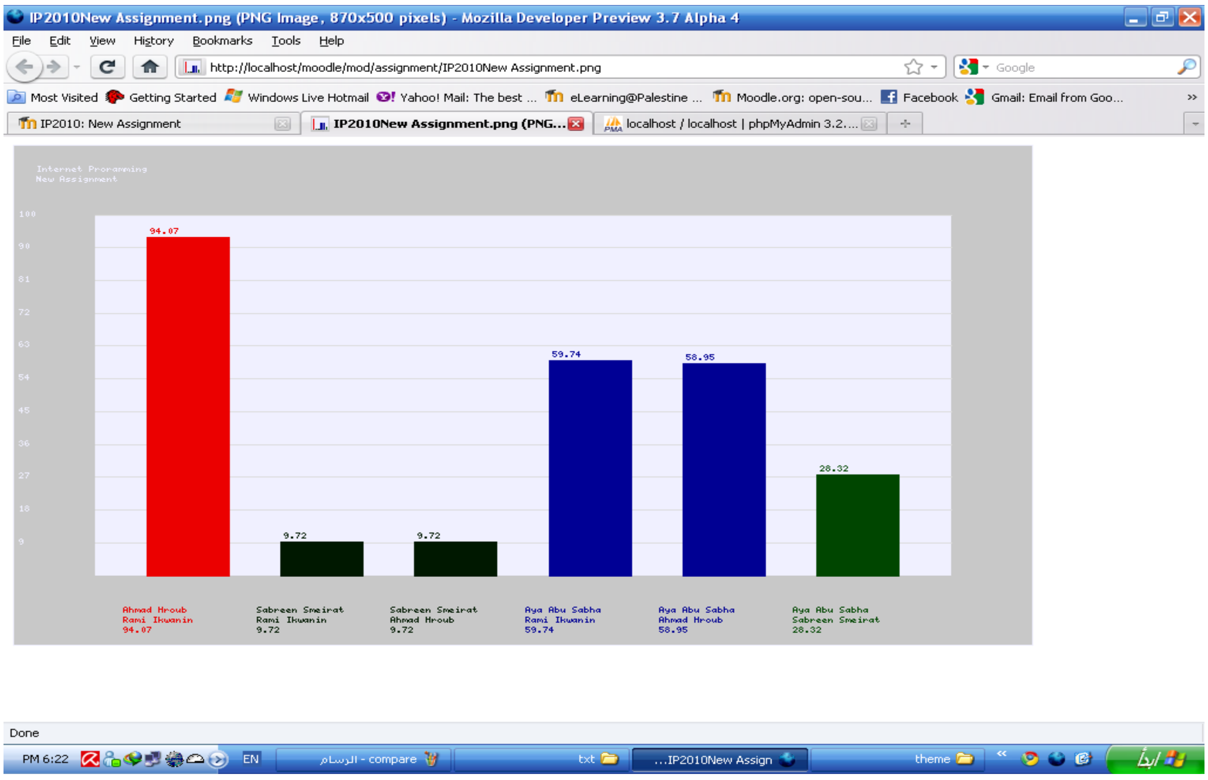
Assignment name : New Assignment. view chart

Student	Ahmad Hroub	Student	Rami Ikwanin	94.07%
File Info	ahmad.txt Saturday, 22 May 2010, 02:38 PM	File Info	rami.txt Saturday, 22 May 2010, 02:38 PM	
Student	Sabreen Smeirat	Student	Rami Ikwanin	9.72%
File Info	docx.gif sabreen.docx Saturday, 22 May 2010, 02:58 PM	File Info	rami.txt Saturday, 22 May 2010, 02:38 PM	
Student	Sabreen Smeirat	Student	Ahmad Hroub	

http://localhost/moodle/mod/assignment/IP2010New Assignment.png

Figure (8): Similarity Page

9. The teacher can view the visual report when he/she click on the view chart link as shown on the previous Figure (9).



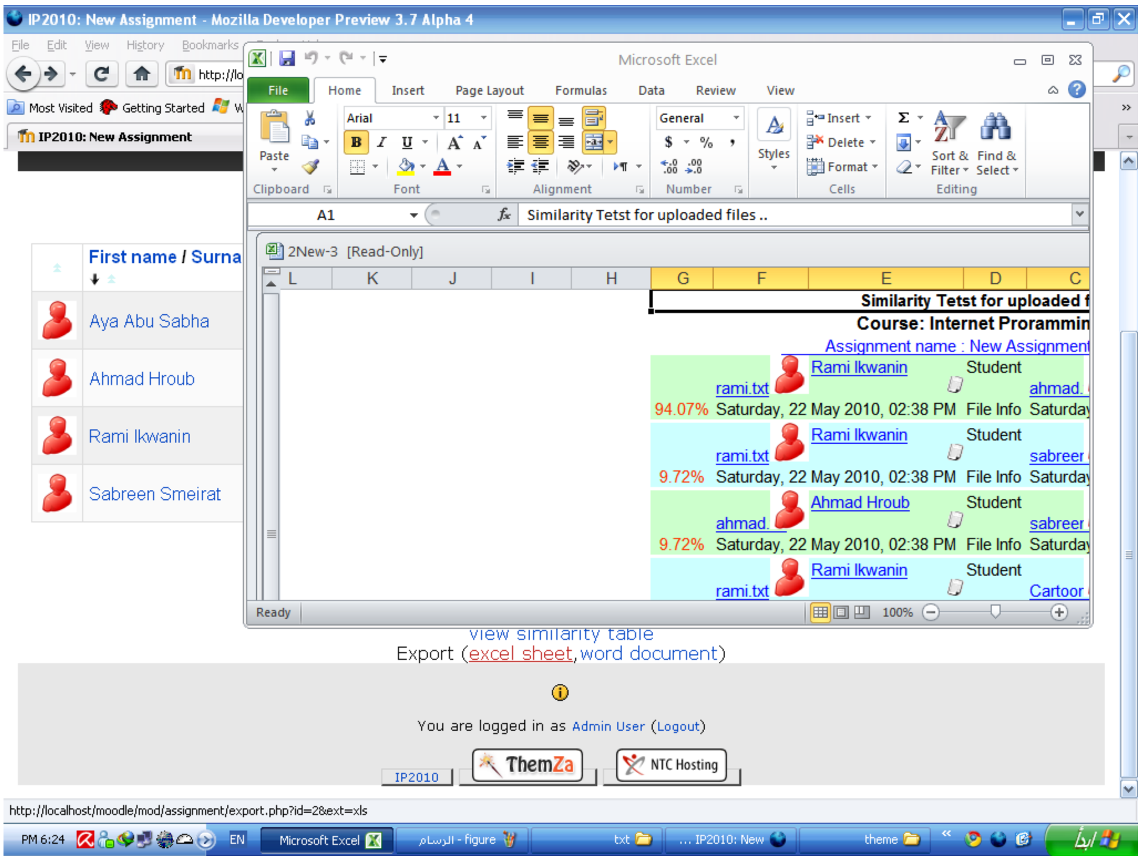
IP2010New Assignment.png (PNG Image, 870x500 pixels) - Mozilla Developer Preview 3.7 Alpha 4

http://localhost/moodle/mod/assignment/IP2010New Assignment.png

Internet Proramming
New Assignment
100
90
81
72
63
54
45
36
27
18
9
94.07
9.72
9.72
59.74
58.95
28.32
Ahmad Hroub Rami Ikwanin 94.07
Sabreen Smeirat Rami Ikwanin 9.72
Sabreen Smeirat Ahmad Hroub 9.72
Aya Abu Sabha Rami Ikwanin 59.74
Aya Abu Sabha Ahmad Hroub 58.95
Aya Abu Sabha Sabreen Smeirat 28.32
Done

Figure (9): Visual Cheating Report

10. The teacher can export the results of the cheating report to an Excel sheet or Word document by clicking on the Excel sheet, Word document link as shown in Figure (10).


IP2010: New Assignment - Mozilla Developer Preview 3.7 Alpha 4
File Edit View History Bookmarks
Most Visited Getting Started
IP2010: New Assignment
Microsoft Excel
File Home Insert Page Layout Formulas Data Review View
Clipboard Font Alignment Number Cells Editing
Paste Styles Insert Delete Format Sort & Filter Find & Select
A1
2New-3 [Read-Only]
Ready 100%
IP2010
ThemZa
NTC Hosting
Microsoft Excel
IP2010: New
theme

Similarity Tetst for uploaded files ..

First name / Surna

Aya Abu Sabha

Ahmad Hroub

Rami Ikwanin

Sabreen Smeirat

```
Similarity Tetst for uploaded f
Course: Internet Prorammin
Assignment name : New Assignment
        rami.txt   Rami Ikwanin   Student   ahmad.
94.07%  Saturday, 22 May 2010, 02:38 PM  File Info  Saturday
        rami.txt   Rami Ikwanin   Student   sabreer
9.72%   Saturday, 22 May 2010, 02:38 PM  File Info  Saturday
        ahmad.     Ahmad Hroub    Student   sabreer
9.72%   Saturday, 22 May 2010, 02:38 PM  File Info  Saturday
        rami.txt   Rami Ikwanin   Student   Cartoor
```

view similarity table

Export (excel sheet,word document)

You are logged in as Admin User (Logout)

http://localhost/moodle/mod/assignment/export.php?id=2&ext=xls

Figure (10): Save as Excel Sheet or Word Document

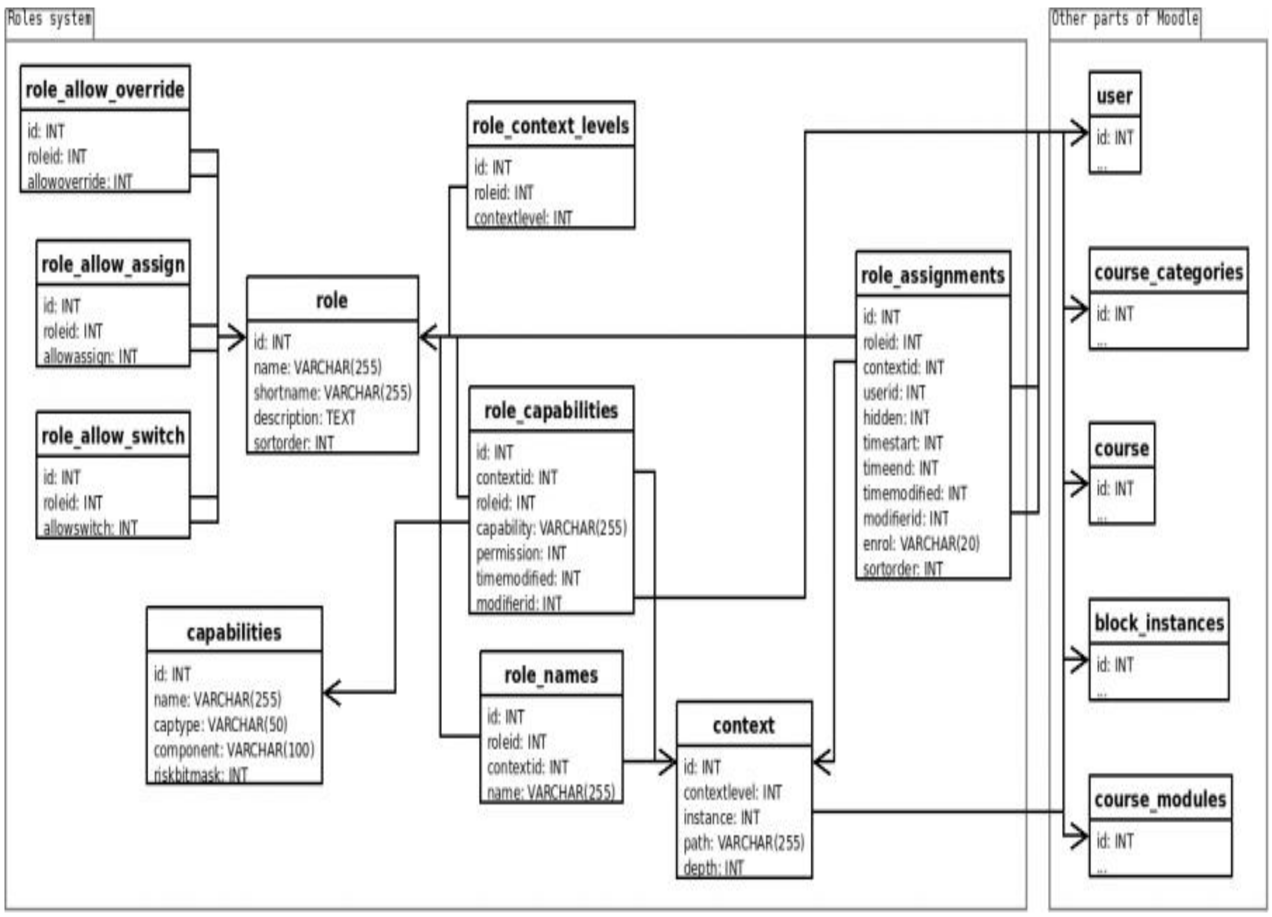
1.4 Glossary

1. **Plagiarism Detection System:** is open-source software that using to locating instance of plagiarism within the documents withers its text or code and reporting the plagiarism cases, and upgrade with the Moodle.
2. **Course Management System (CMS):** is a tool that allow instructors to post information on the web, this information related courses as course chapters, videos, audios, assignments, and so on, this is a tool facilitates course management by the instructors.
3. **Learning Management System (LMS):** is a software application for the administration, documentation, tracking, and reporting of training programs, classroom and online events, e-learning programs, and training content.

4. **Virtual Learning Environment (VLE):** is a software system designed to support teaching and learning in an educational setting, where the focus is on management.
5. **Open-Source:** describes practices in production and development that promote access to the product's source materials. Before the term open source became widely adopted, developers and producers used a variety of phrases to describe the concept; open source gained hold with the rise of the Internet, and the attendant need for massive retooling of the computing source code.
6. **Modular Object-Oriented Dynamic Learning Environment (Moodle):** is an Open Source CMS, also known as a LMS or a VLE. It has become as popular e-learning platform in higher education to creating online dynamic web sites for their students (http://www.moodle.org/about/).

Appendix (B)

Moodle Database Schema


Roles system
Other parts of Moodle
role_allow_override
id: INT
roleid: INT
allowoverride: INT
role_allow_assign
id: INT
roleid: INT
allowassign: INT
role_allow_switch
id: INT
roleid: INT
allowswitch: INT
role
id: INT
name: VARCHAR(255)
shortname: VARCHAR(255)
description: TEXT
sortorder: INT
role_context_levels
id: INT
roleid: INT
contextlevel: INT
role_capabilities
id: INT
contextid: INT
roleid: INT
capability: VARCHAR(255)
permission: INT
timemodified: INT
modifierid: INT
capabilities
id: INT
name: VARCHAR(255)
captype: VARCHAR(50)
component: VARCHAR(100)
riskbitmask: INT
role_names
id: INT
roleid: INT
contextid: INT
name: VARCHAR(255)
context
id: INT
contextlevel: INT
instance: INT
path: VARCHAR(255)
depth: INT
role_assignments
id: INT
roleid: INT
contextid: INT
userid: INT
hidden: INT
timestart: INT
timeend: INT
timemodified: INT
modifierid: INT
enrol: VARCHAR(20)
sortorder: INT
user
id: INT
...
course_categories
id: INT
...
course
id: INT
...
block_instances
id: INT
...
course_modules
id: INT
...