|  |
| --- |
| ATMA RAM SANATAN DHARMA COLLEGE |
| ONLINE TEST MANAGEMENT TOOL (Documentation) |
| SOFTWARE ENGINEERING PROJECT |
|  |
|  |
| **SEMESTER IV (2016-17)** |

*Submitted to- Submitted by-*

Ms. Uma Ojha Kumar Prateek Viraj (38068)

Kuljeet Kumar (38076*)*

Certificate

This is to certify that

Kumar Prateek Viraj and Kuljeet Kumar

Students of course B.Sc.(H) Computer Science

Semester-IVth of the batch 2016-17

have successfully completed their

Software Engineering Project

on the topic

‘Online Test Management Tool’,

under the

supervision and guidance of the respective

subject teacher Ms. Uma Ojha’.

Signature Signature

(External Examiner) (Subject Teacher)

**ACKNOWLEDGEMENT**

We have taken efforts in this project as a team.

However, it would not have been possible without the kind support and help of many individuals.

We would like to extend my sincere thanks to all of them.

We are highly indebted to our Software Engineering teacher –

Ms. Uma Ojha ma’am for her guidance and constant supervision as well as for providing necessary information regarding the project & also for her support in completing the project.

We would like to express our gratitude towards our parents & friends for their kind co-operation and encouragement which helped us in completion of this project.

Our thanks and appreciations also go to the people who have willingly helped us out with their abilities.

PROBLEM STATEMENT

An online web based portal for the automation and management of examinations and evaluation with result generation functionalities.

A system where the administrator can make questions and design tests and assign it to students who in turn attempt the examination.

The system should automatically evaluate the answer scripts, calculate the score for each individual and display their performance statistics to them. The system should also provide the ability to store ones past results and exams attempted.

The administrator should be able to add as many questions to the question bank create tests and view any of the candidate’s result providing his/her details.

The student should be able to attempt as many tests, view his/her scores and review his/her performance over time.

Overall focus of the system should be in minimizing the human effort to conduct an examination and maintain it.

PROCESS MODEL

Since, the project is an online web-interface, it required a very high amount of interaction with users, and the product to be built must allow ease of use and need minimal training for the end users.

Hence, we chose the ‘EVOLUTIONARY PROTOTYPING’ model for the development of this project.

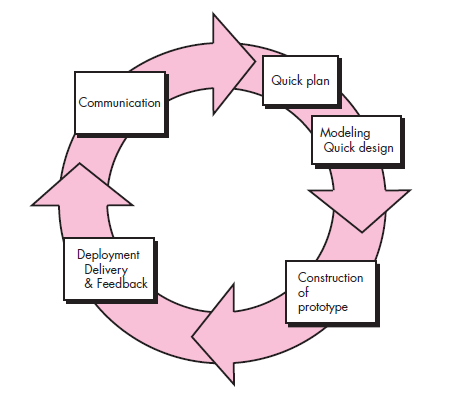
The selected model allows the project to be built, tested, and then reworked as necessary until an acceptable prototype is finally achieved from which the complete system can now be developed.

Since in this methodology,a working model of the system is provided, the users get a better understanding of the system being developed.

Errors can be detected much earlier.

Quicker user feedback is available leading to better solutions and also

Missing functionality can be identified easily.

The Prototyping Paradigm

REQUIREMENT ANALYSIS

1. Name of the Project Online Test Management Tool

2. Objective/ Vision

1. To provide a web based online system for candidates to attend tests.

2. To enable the application owner to effectively manage, creation and assignment of tests .

3. To enable the application owner to create customized tests, choosing questions depending on the skills being tested.

4. To completely automate the process right from creating tests to the point when the completed tests are evaluated.

3.Users of the System

A. Candidates who take tests

B. Admin/HR who create and assign tests and contribute questions to the question bank.

4. Functional Requirements

• System should enable admin to Create questions that maintain a bank of different types of questions chosen in various tests.

• System should enable admin to Setup Tests by choosing different questions from the question bank.

• System should enable Candidate to attend assigned tests and submit the same.

• System should provide an interface for admin to view Question Bank, Available Tests, and Assigned Tests.

• System should provide an interface for admin to view Completed Tests and their results.

•System should provide an interface for admin to view Question Bank, available Tests, and assigned Tests.

• System should provide the ability for the candidate to preview his/her result.

• System should provide the ability for the candidate to preview his/her result – Rightly answered, wrongly answered questions, unanswered ones.

Sorted Risk Table

|  |  |  |  |
| --- | --- | --- | --- |
| Risks | Category | Probability  (%) | Impact |
| Server cannot handle large no. of users | TE | 80 | 2 |
| Inexperienced team | ST | 70 | 1 |
| Technology will not meet expectations | TE | 40 | 2 |
| System Security will be hacked | TE | 50 | 1 |
| Less number of team members  Complexity of Application  Failure to meet high performance  Unclear requirements  Customer will change requirement  Lack of User training  Less reuse than planned | ST  PS  TE  PS  CU  BU  BU | 70  40  30  40  20  10  30 | 3  4  4  4  2  4  4 |
| Note: 1- Catastrophic 2-Critical 3-Marginal 4-negligible    Category Abbreviation : BU- Business Risk ; ST-Staff size and experience; TE-Technology Risk  PS-Process Defination ; CU- Customer Characteristic Risk | | | |

\*Risks above the dotted line will be considered further.

ESTIMATION

FP BASED:

The function point (FP) metric can be used effectively as a means for measuring the functionality delivered by a system.

Function points are derived using an empirical relationship based on countable(direct) measures of software’s information domain and qualitative assessments of software complexity.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Information domain value** | Optimistic | Likely | Pessimistic | Est.  Count. | Weight | FP Count |
| Number of external inputs | 4 | 5 | 7 | 5 | 3 | 15 |
| Number of external outputs | 3 | 4 | 7 | 4 | 5 | 20 |
| Number of external inquiries | 2 | 2 | 4 | 2 | 3 | 06 |
| Number of internal logical files | 3 | 5 | 5 | 5 | 7 | 35 |
| Number of external interface files | 0 | 0 | 1 | 0 | -- | 0 |

Count Total 76

VALUE ADJUSTMENT FACTORS

0 (Not Important) to 5 (Absolutely Essential)

**Factor Value**

Backup and recovery 2

Data communications 0

Distributed processing 2

Performance critical 3

Existing operating environment 3

Online data entry 5

Input transaction over multiple screens 2

Master files updated online 4

Information domain values complex 2

Internal processing complex 1

Code designed for reuse 1

Conversion/installation in design 0

Multiple installations 2

Application designed for change 4

**Value adjustment factor 0.96**

FPestimated = count total\* [0.65 + 0.01 \*∑(*Fi*)] = 73

EFFORT CALCULATION

COCOMO II MODEL

It uses ‘object points’ as an indirect software measure that is computed using counts of the number of -

(1) Screens (at the user interface),

(2) Reports, and

(3) Components likely to be required to build the application.

|  |  |  |  |
| --- | --- | --- | --- |
| **Object Type** | Complexity Weight | Count | Object Points |
| Screens | 2 (Medium) | 10 | 20 |
| Reports | 2 (Simple) | 3 | 6 |
| 3GL Component | - | 0 | 0 |
| Total Object Points | | | **26** |

NOP = (object points) \* [(100 - %reuse)/100]

= 26 (Resuability was Zero%)

Estimated Effort = NOP / PROD

(Taking Environment maturity = Nominal

Developer’s capability = Nominal

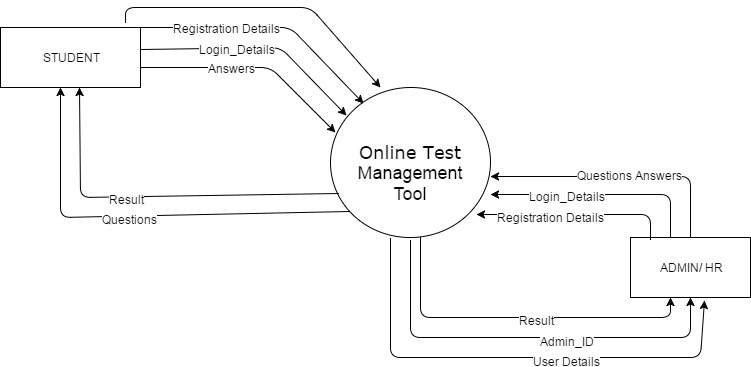
PROD = 13)

= 26 /13

= 2 person-month

DATA FLOW DIAGRAMS

CONTEXT DIAGRAM



DATA DICTIONARY

REG\_DETAILS = USERNAME \* EMAIL \* PASSWORD

LOGIN\_DETAILS = [USERNAME \* PASSWORD] + ADMIN\_ID

ADMIN\_ID = digit +digit +digit +digit

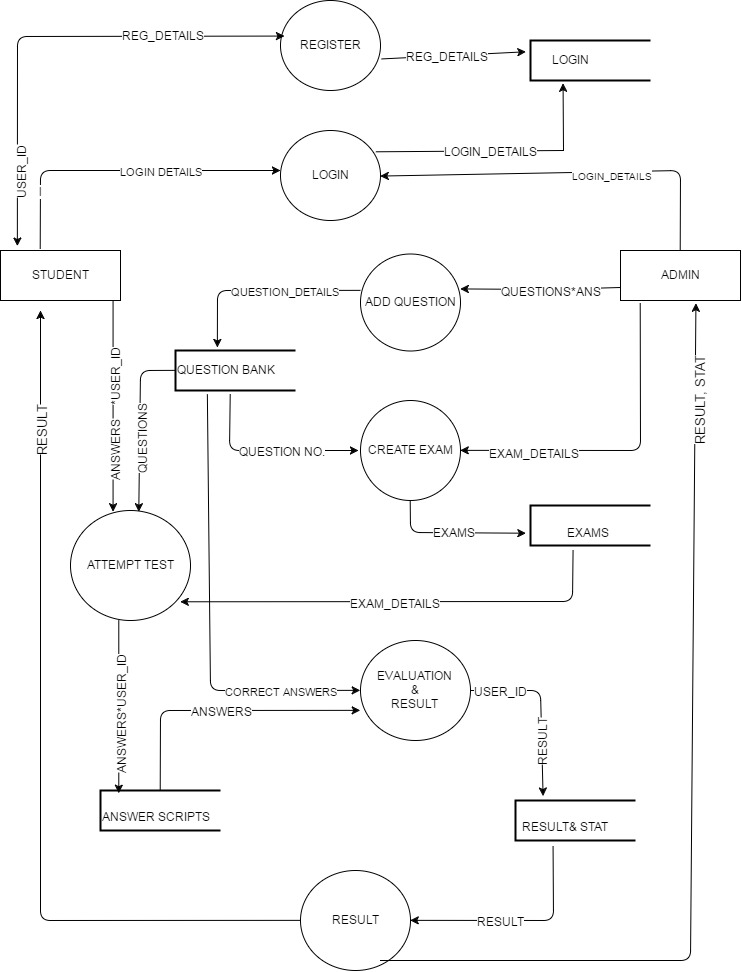
USER\_ID = digit +digit +digit (auto increment)

EXAM\_DETAILS = [EXAM\_ID \* SUBJECT\_NAME \* NO of QUES ] + DURATION

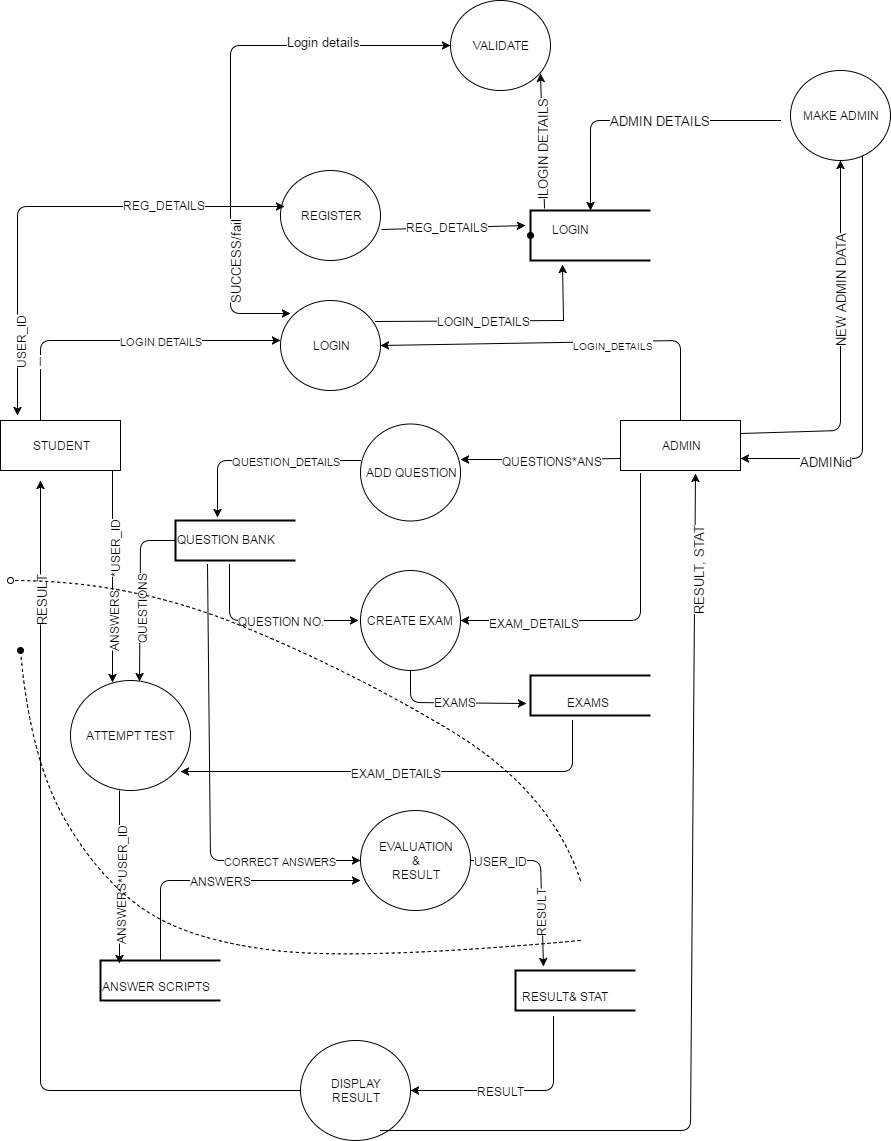
ATTEMPTED\_EXAMS = EXAM\_ID \* USER\_ID \* ANSWERS\*QNO.

RESULT = SCORE \* EXAM\_ID \* user\_ANSWER \* correct\_ANSWERS

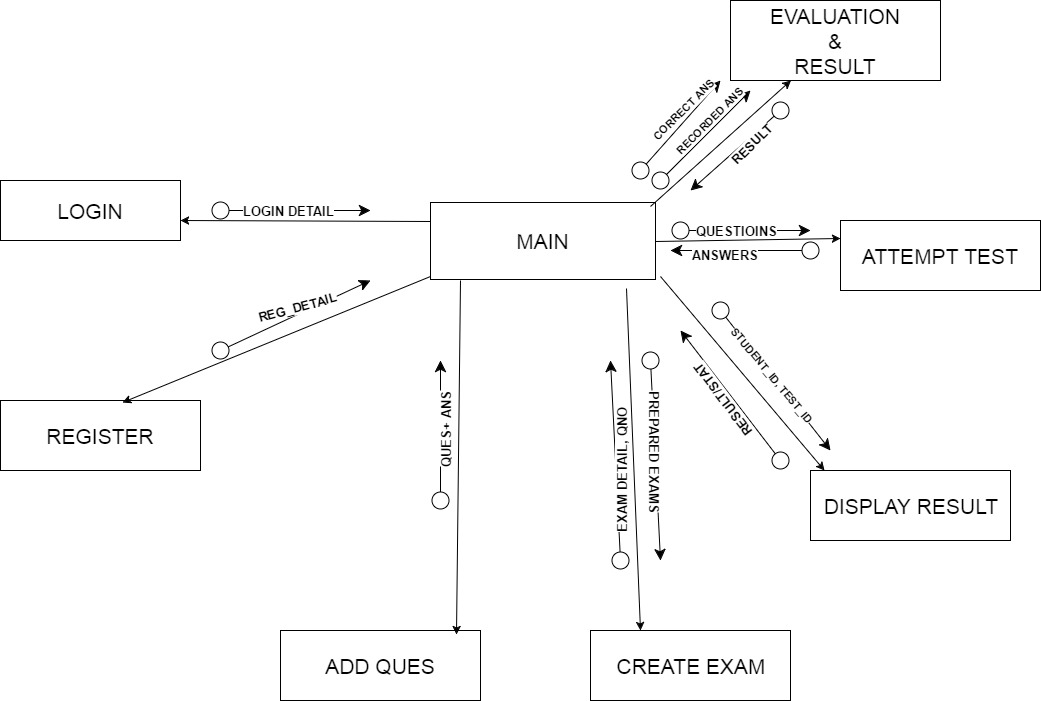
LEVEL 01



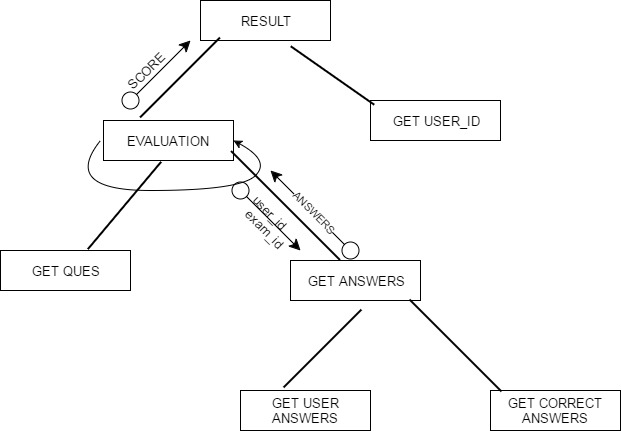
ARCHITECTURAL DESIGN

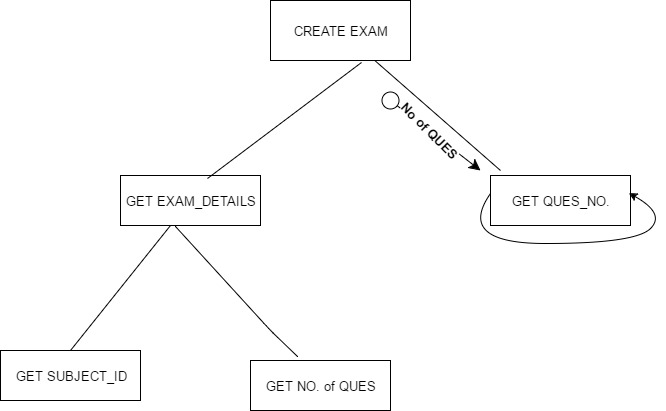
DESIGN DFD

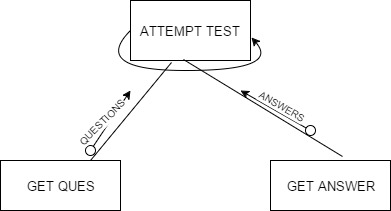
FIRST LEVEL FACTORING



Factoring Modules

 (continued..)





WHITE BOX TESTING

PSEUDOCODE

#Student\_registration.php

1.If (REQUEST\_METHOD==POST){

2. If (password==confirm\_password){

3. $variable ← extract( form\_data)

4. check\_duplication (email)

5. If (email already registered){

6. ERROR1;

7. }Else { Login\_database ← $variable

8. If (data inserted in db){

9. Goto student\_Homepage;

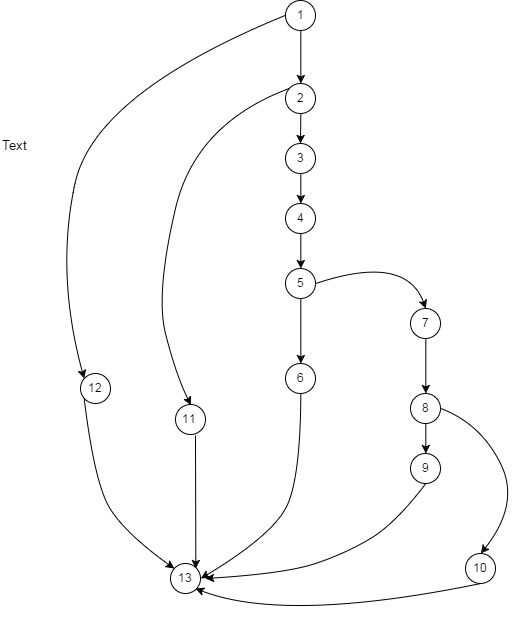
10. }else{ ERROR: In Registering }

11. }else { ERROR:PASSWORD DON’T MATCH}

12.}Else { ERROR:FORM NOT SUBMITTED}

13.END

FLOWGRAPH



CYCLOMATIC COMPLEXITY:

(i) V(G) = E - N + 2 =16edges -13 nodes + 2 = 5

(ii) V(G) = P+1 = 4 predicate nodes + 1 = 5

(iii) Number of regions in the graph = 5

INDEPENDENT PATHS:

Path1 : 1-2-3-4-5-6-13

Path2 : 1-12-13

Path3 : 1-2-11-13

Path4 : 1-2-3-4-5-7-8-9-13

Path5: 1-2-3-4-5-7-8-10-13

DATA DESIGN

Database name: onlineexamportal

Table name: login

|  |  |  |
| --- | --- | --- |
| Attribute name | Data Type(length) | Remarks |
| ID | Integer(11) | Primary key, auto-increment |
| Name | Varchar(255) | NULL |
| Username | Varchar(255) | unique |
| Password | Varchar(255) | Md5 hashed |
| usertype | Integer(3) | Binary value |
| adminid | Integer(4) | NULL |

Table name: questionbank\*

|  |  |  |
| --- | --- | --- |
| Attribute name | Data type(length) | Remarks |
| qno | Int(11) | Primary key |
| testid | Int(4) | multivalued |
| ques | Varchar(512) | Unique |
| a | Varchar(255) | Option a |
| b | Varchar(255) | Option b |
| c | Varchar(255) | Option c |
| d | Varchar(255) | Option d |
| ans | Varchar(255) | Correct answer |

Table name:exams

|  |  |  |
| --- | --- | --- |
| Qno | Int(11) | Partial key, refers to questionbank.qno |
| testid | Int (4) | Partial key |
| No. of ques | Int(3) | Total no of ques |

Table name: result

|  |  |  |
| --- | --- | --- |
| Userid | Varchar(255) | Partial key, refers to login.id |
| Testid | Integer(4) | Partial key, references exam.testid |
| score | Integer(3) | score |

Table name: answers\*

|  |  |  |
| --- | --- | --- |
| Userid | Varchar(255) | Partial key, refers to login.id |
| Testid | Integer(4) | Partial key, references exam.testid |
| qno | Integer(11) | Refers to questionbank.qno |
| answer | Integer(3) | Option no. |

\*tables haven’t been normalized

FUTURE SCOPE

Digitalization of exams is not a new thing to talk about.

This project, without altering the central theme of examinations, tries to make the whole process of taking and giving exams interesting, easier and simple.

The selection of Evolutionary Prototyping Model for the development of project development, provides an opportunity to make changes to the project and adding more functionalities to it.

It can further be enhanced to be a whole virtual learning system itself, or can be connected to an existing system as an examination portal also,since it is an independent (except for a few changes) portal for examinations and is highly customizable as per the need.

INDEX

1.Problem Statement

2. Process Model

3. Data Flow Diagrams-

3.1 Context Diagram & Data Dictionary

3.2 Level 01 DFD

4. Project Management –

4.1 Computing FP

4.2 Computing Effort

4.3 Timeline Chart

4.4 Risk Table

5. Design Engineering –

5.1Architectural Design

5.2 Component Level Design

5.3 Data Design

6.Testing-

6.1 PseudoCode

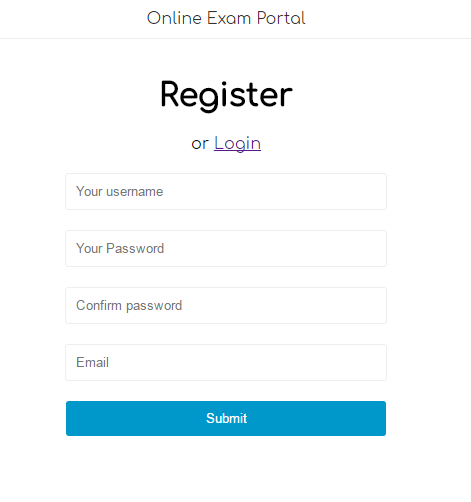
6.2 Cyclomatic Complexity

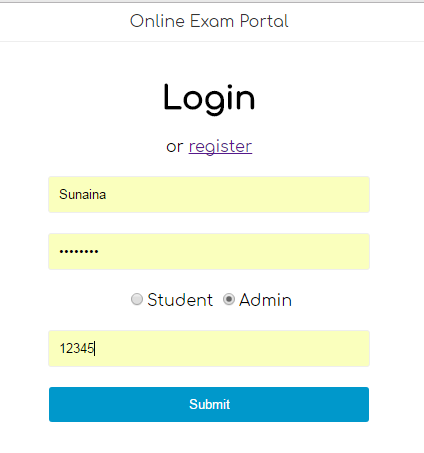
7. Future Scope

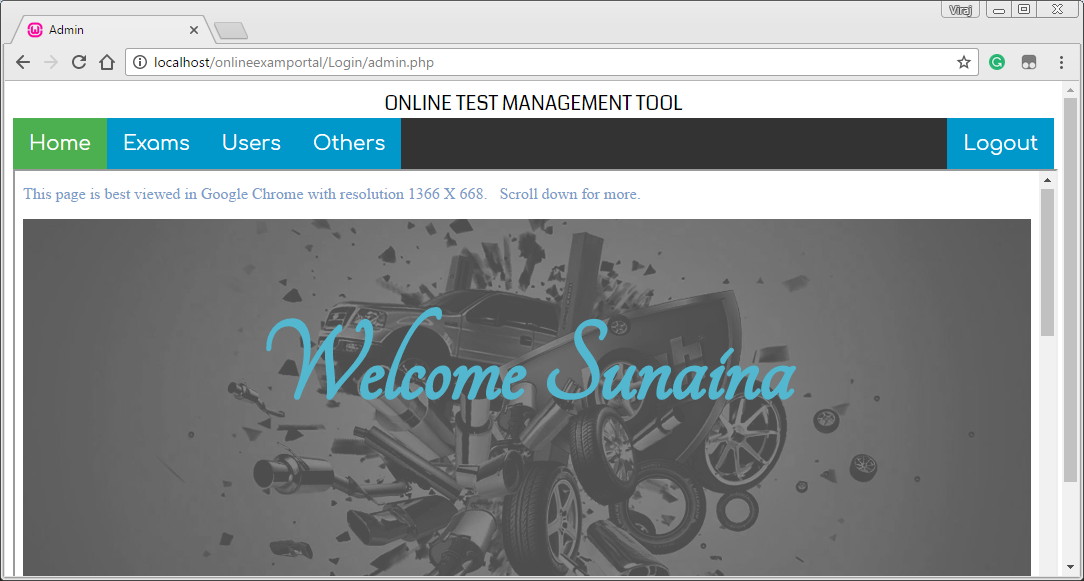
8.Screen Shots –

8.1 Register &Login

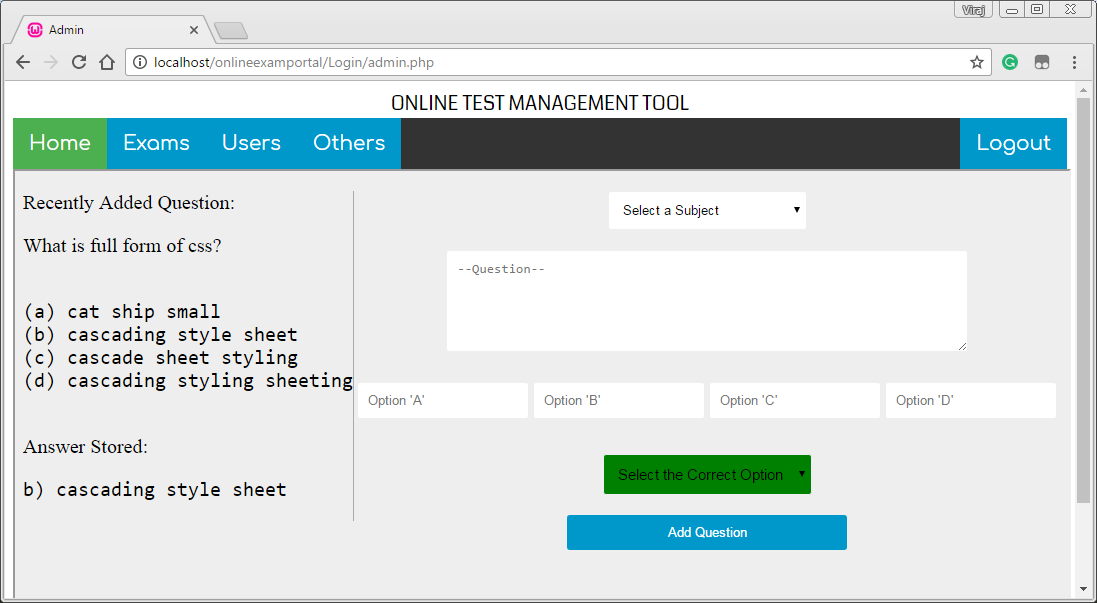
8.2 Admin Dashboard,Add question, Attempt Test



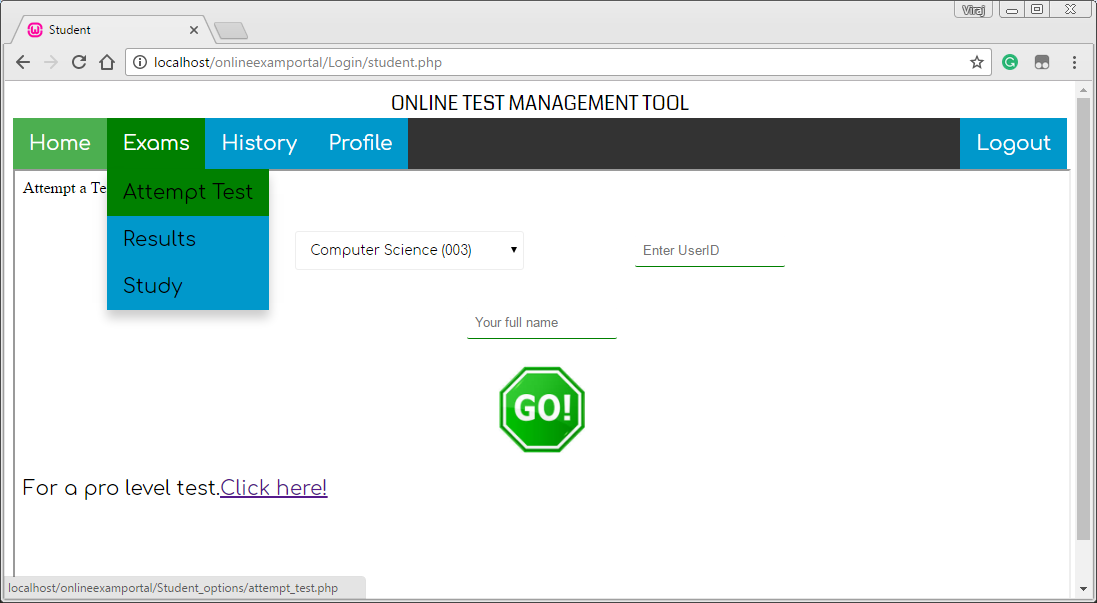




ADMIN DASHBOARD



ADD QUESTION

ATTEMPT TEST