Contents

Page No.

|  |
| --- |
| **1** |
| 2 |
| 2 |
| **3** |
| 4 |
| 4 |
| 4 |
| 4 |
| 5 |
| 6 |
| 7 |
| 7 |
| 7 |
| 7 |
| **8** |
| 9 |
| 9 |
| 10 |
| 12 |
| 13 |
| 13 |
| **15** |
| 16 |
| 16 |
| **26** |
| 27 |
| 28 |
| 28 |
| **29** |
| **31** |
| 32 |
| **38** |

1. Introduction

1.1. Overview of the system

1.2. Problem definition and objective

2. System Analysis

2.1. Introduction

2.2. Existing system

2.3. Proposed system

2.3.1. Justification of the proposed system

2.3.2. Benefits of the proposed system

2.4. Feasibility study

2.5. System specification

2.5.1. Software specification

2.5.2. Tools/platform

2.5.3. Hardware specification

3. System design

3.1. Introduction

3.2. Input design

3.3. Output design

3.4. Database design

3.5. ER Diagram

3.6. Data Flow Diagram

4. System Development

4.1. Process Description

4.2. Source Code

5. System Implementation

5.1. Testing

5.2. System Implementation

5.3. Security

6. Conclusion

7. Appendix

7.1. Sample input and Sample output screens

8. Bibliography

1. INTRODUCTION

1.1. Overview of the system

Online examinations are an important method of evaluating the success potential of students. This research effort the individuals under consideration were students who would be enrolling in computer courses or Technologies Registrations. A prototype of a web-based placement examination system is described from the standpoint of the research effort, end user, and software development.

An on-line educational system including exam processing and electronic journal features. An instructor builds a course based questions which on-line contain in identification of assignments. Which are compiled into an on-line exam syllabus?

Users enrolled in the platform may access the electronic details they provided and perform various functions with the on-line educational system in order to participate in the on-line examinations. Users can receive an on-line exam, having multimedia content, for the course, and they can electronically provide answers for the exam. And after Completion of their duration of exam they are provided the grade or marks secured in their examinations.

1.2. Problem definition and objective

Problem definition

The problem here is to develop a web application which completely automize the examination process for an institution.

To perform this, the complete problem has been divided into five sub-problems so that they can be solved easily and after that can be integrated to make it an integrated working application. That is,

1. Online login
2. Online schedule
3. Question bank management
4. Conducting examination
5. Online result declaration

At the login module, username and password facility and credentials should be checked properly at the time of login for student, faculty and Examination Department administrator.

In the schedule Module, it sends the exam schedule to the student.

Question bank module for the examination could be changed dynamically.

Automatic checking of objective answers and manual checking of single-sentenced answers should be facilitated, if any. The Objective answers module will be checked automatically by the system from the database.

In the exam module, when the student starts examination the timer will starts automatically and shows the student how much time is left.

At result module, it displays the result of each student and send by the examination department.

**Objective**

* This can be used in educational institutions
* No restriction that examiner has to be present when the candidate takes the test.

1. SYSTEM ANALYSIS

2.1. Introduction

The problem here is to develop a web application which completely automize the examination process for an institution.

To perform this, the complete problem has been divided into five sub-problems so that they can be solved easily and after that can be integrated to make it an integrated working application. That is,

1. Online login
2. Online schedule
3. Question bank management
4. Conducting examination
5. Online result declaration

2.2. Existing system

The existing system of conducting examination process in our institution is manual. Existing system is a large ma power process and is difficult to implement it in different platform. It has so many problems. So, we introduced an online examination system, which is fully computerized. Existing system is a large man power and is difficult to implement.

Disadvantages

* Time consuming
* Difficult to analyze the exam manually
* Results are not precise as evaluations are done annually
* Result processing after summation of exam tales more time as it is done manually,

These problems can be solved by introducing a new system. To overcome these problems we are implementing a computerized system with combination of MVC, HTML, Java Script, J Query, CSS and bootstrap.

* 1. Proposed system

The present system has limitations. The proposed system is designed to eliminate all the disadvantages of the existing system. The proposed system is an interactive system, highly user friendly and designed exclusively for the Authority. The system is developed in MVC and SQL SERVER as backend.

The proposed system reduces manpower, paper wastage etc. it is very efficient and saves time. The system is having many added advantages which are having higher number of economic features for user interface. The system modular structure will allow the system to pair with other system for easy data transfer with high control levels in user hands. The proposed system will be more appropriate and genuine without many errors and by using this application question paper and result can be generated easily.

It is designed keeping in mind all the drawbacks of the present system. The proposed system overcomes all the hurdles faced by the existing system.

2.3.1. Justification of the proposed system

The modern computerized system is developed with the aim to overcome the drawbacks of existing manual system. The proposed system has got many advantages. People from different parts of the world can register very easily. The new system is more personalized. It is a maze in such a manner that all the new users can understand all the options in it very easily. It is made in a quick and easy referential manner. Access to all important matters is not always locked and can be opened easily at the time of urgency. The advantages of proposed system are that security is maintained in the new system. Securities for all important data are maintained confidentially. As it is easily understandable and user-friendly, quickly entries can be made in this system.

2.3.2. Benefit of the proposed system

* Provides complete online web-based solution, including student login, giving tests, storing of results.
* Complete web-based administration, administrator can manage examination, student registration and question bank from web interface.
* Student ca give examination without pen and paper
* 100% accuracy in result calculation
* Faculty can manage courses and question bank

2.4. Feasibility study

The purpose of the feasibility study is to investigate the present system, evaluate the cost and effectiveness of the proposed system, evaluate the

possible applications of computer-based methods, select a tentative system, evaluate the impact of the proposed system on existing personnel and ascertain the need for new personnel.

Economic Feasibility

Economic analysis is most frequently used for evaluation of the effectiveness of the system. More commonly known as cost/benefit analysis the procedure is to determine the benefit and savings that are expected from a system and compare them with costs, decisions are made to design and implement the

system. This part of the feasibility study gives the top management the economic justification for the new system. This is an important input to the management, because very often the top management does not like to get confused by the various technicalities that are bound to be associated with a project of this kind. A simple economic analysis that gives the actual comparison of costs and benefits is much more meaningful in such cases. In the system, the organization is most satisfied by economic feasibility. Because, if the organization implements this system, it need not require any additional hardware resources as well as it will be saving a lot of time.

Technical Feasibility

Technical feasibility centers on the existing manual system of the test management process and to what extent it can support the system. According to the feasibility analysis procedure the technical feasibility of the system is analyzed and the technical requirements such as software facilities, procedure, inputs are identified. It is also one of the important phases of the system development activities.

Operational Feasibility

The system offers greater levels of user friendliness combined with greater processing speed. Therefore, the cost of maintenance can be reduced. Since, processing speed is very high and the work is reduced in the maintenance point of view, management is convinced that the project is operationally feasible.

* 1. System specification

System specifications help to define the operational and performance guidelines for a system. It may outline how the system is expected to perform, and what that may include. Key specifications may include interface definitions, document design rules and functional areas.

2.5.1. Software specification

Operating system: Windows 10

Front-End: MVC

Back-End: SQL Server 2017

2.5.2. Tools/platform

Visual Studio 2019

Internet Explorer

2.5.3. Hardware specification

Processor: Intel® Pentium® CPU N3540 @2.16GHz 2.16GHz

RAM: 4.00 GB

HDD: 250 GB or Above

1. SYSTEM DESIGN

3.1. Introduction

The system is a solution “how to” approach to the creation of new system. The basis for the system design is a good system analysis. The important phase is composed of several steps. The first step is to determine how the output should be produced and in what format. Second the database, the input data and the master files have to be designed to meet the requirements of the proposed system. It provides for the understanding and procedural details necessary for implementing the candidate system o the user and the organization are documented and evaluated by the management as steps towards implementation. System design goes through phases development, logical and physical design.

DFD shows the logical flow of a system and defines the boundaries of the system. For a candidate system, it describes the inputs, outputs, database and procedures. The design covers the review of the current physical system, prepares output specifications, prepares input specifications and specifies the implementation plan, reviews benefits, costs, and target dates and constrains.

System design objectives include specify logical design elements, support business activities ensure that systems features meet user requirements, provides engineered for ease of use by people, provides detailed software development specification, confirm to design standards.

The logical design specifics output, input file and screen layout. The physical design procedures the working system by defining the design specification that tells the programmer exactly what the candidate system must do.

3.2. Output design

One of the important features of an information system for users is the output it produces. Output is the information delivered to users through the information system. Without quality output, the entire system appears to be unnecessary that users will avoid using it. Uses generally merit the system solely by its output. In order to create the most useful output possible. One works closely with the user order to create the most useful output possible. One works closely with the user through an interactive process, until the result is considered to be satisfactory.

Output design has been an ongoing activity almost from the beginning of the project. In the study phase, outputs were identified and described general in the project directive. A tentative output medium was then selected and sketches made for each output. In the feasibility analysis, a “best” new system was selected; its description identified the input and output media. In the design phase the system has included an evolution and selection of specific equipment for the system.

Outputs from computer system are required primarily to communicate the results of processing to the user.

Admin\_tbl

|  |  |  |
| --- | --- | --- |
| admin\_id | admin\_name | admin\_password |
| 1 | Noufiya | noufiya |
| 2 | Sameesha | sameesha |

Candidate\_tbl

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| candidate\_id | candidate\_name | candidate\_password | candidate\_email | candidate\_contact | candidate\_institution |
| 1 | Anu | anu@1234 | anuanu@gmail.com | 7896543210 | MES COLLEGE MARAMPALLY |
| 2 | Ajith | ajith123 | ajith@gmail.com | 9876054321 | MES COLLEGE MARAMPALLY |

Exam\_tbl

|  |  |  |
| --- | --- | --- |
| exam\_id | subject\_id | exam\_title |
| 1 | 2 | Test1 |

Question\_tbl

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| all\_qus\_id | qus\_id | admin\_id | exam\_id | qus\_title |
| 1 | 1 | 1 | 1 | HTML stands for |
| 2 | 2 | 1 | 1 | The correct sequence of HTML tags for starting a webpage is - |
| 3 | 3 | 1 | 1 | Which of the following element is responsible for making the text bold in HTML |
| 4 | 4 | 1 | 1 | Which of the following tag is used for inserting the largest heading in HTML |
| 5 | 5 | 1 | 1 | Which of the following tag is used to insert a line-break in HTML |
| 6 | 6 | 1 | 1 | How to create an unordered list (a list with the list items in bullets) in HTML |
| 7 | 7 | 1 | 1 | Which character is used to represent the closing of a tag in HTML |
| 8 | 8 | 1 | 1 | How to create an ordered list (a list with the list items in numbers) in HTML |
| 9 | 9 | 1 | 1 | How to insert an image in HTML |
| 10 | 10 | 1 | 1 | input is - |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ans1 | ans2 | ans3 | ans4 | correct\_ ans | Selectedvalue |
| HyperText Markup Language | HighText and links Markup Language | HighText Machine Language | None of these | 1 | NULL |
| Head, Title, HTML, body | HTML, Body, Title, Head | HTML, Title, Head, Body | HTML, Head, Title, Boday | 4 | NULL |
| Pre | a | b | br | 3 | NULL |
| h3 | h1 | h5 | h6 | 2 | NULL |
| Br | a | pre | b | 1 | NULL |
| Ul | Ol | li | i | 1 | NULL |
| \ | ! | / | . | 3 | NULL |
| Ul | Ol | li | i | 2 | NULL |
| img href="image.jpg" | img url="image.png" | img link="image.jpg" | img src="image.png" | 4 | NULL |
| a format tag | an empty tag | All of the above | None of the above | 2 | NULL |

Result\_tbl

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| result\_id | exam\_id | candidate\_id | total | grade | quality |
| 1 | 1 | 1 | 10 | A+ | Best |
| 2 | 1 | 1 | 7 | B | Good |

Subject\_tbl

|  |  |  |
| --- | --- | --- |
| subject\_id | subject\_title0 | subject\_code |
| 1 | HTML | S01 |
| 2 | CSS | S02 |

3.3. Input design

Input design, user oriented data are converted to a computer based format. It is the link between user and the information system. The input design involves determining what the inputs are, how the data should be performed, how to validate data, how to minimize data entry and how to provide a multiuser facility.

The main objectives of the input design are:

1. Produce cost effective method of input

2. Achieve highest possible level of accuracy

3. Ensure that the input is acceptable to and understood by user

The goal of designing input data is to make entry easy, logical and free from errors as possible. The entering users like manager need to know the allocated space for each field; field sequence and which must match with that in the source document.

3.4. Database design

**Table: Admin\_tbl**

|  |  |  |
| --- | --- | --- |
| admin\_id | Int | Primary key |
| admin\_name | varchar(50) | Null |
| admin\_password | varchar(50) | Null |

Table: Candidate\_tbl

|  |  |  |
| --- | --- | --- |
| candidate\_id | int | Primary key |
| candidate\_name | varchar(50) | Null |
| candidate\_password | varchar(50) | Null |
| candidate\_email | varchar(50) | Null |
| candidate\_contact | varchar(50) | Null |
| candidate\_institution | varchar(50) | Null |

Table: Exam\_tbl

|  |  |  |
| --- | --- | --- |
| exam\_id | int | Primary key |
| subject\_id | int | Foreign key from the table Subject\_tbl |
| exam\_title | varchar(50) | Null |

Table: Questions\_tbl

|  |  |  |
| --- | --- | --- |
| all\_qus\_id | int | Primary key |
| qus\_id | int | Null |
| admin\_id | int | Foreign key from the table Admin \_tbl |
| exam\_id | int | Foreign key from the table Exam\_tbl |
| qus\_title | varchar(300) | Null |
| ans1 | varchar(300) | Null |
| ans2 | varchar(300) | Null |
| ans3 | varchar(300) | Null |
| ans4 | varchar(300) | Null |
| correct\_ans | int | Null |
| selectedvalue | int | Null |

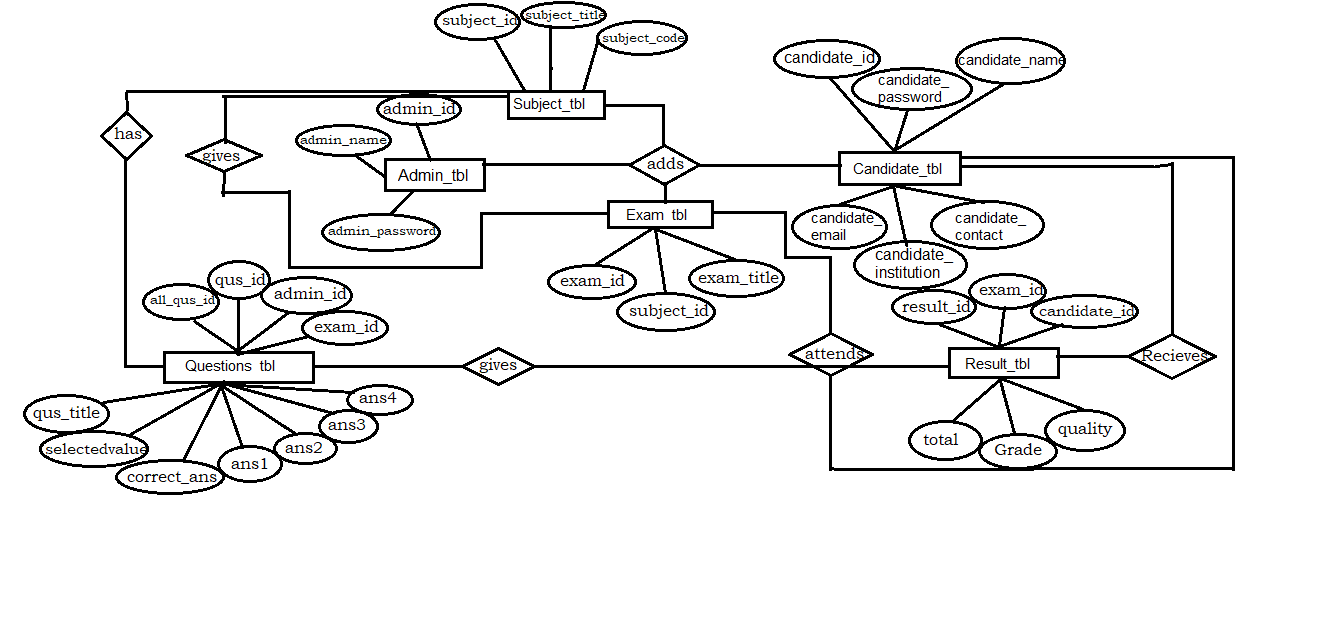
Table: Result\_tbl

|  |  |  |
| --- | --- | --- |
| result\_id | int | Primary key |
| exam\_id | int | Foreign key from the table Exam\_tbl |
| candidate\_id | int | Foreign key from the table Candidate\_tbl |
| total | int | Null |
| Grade | varchar(50) | Null |
| quality | varchar(50) | Null |

Table: Subject\_tbl

|  |  |  |
| --- | --- | --- |
| subject\_id | int | Primary key |
| subject\_title | varchar(50) | Null |
| subject\_code | varchar(50) | Null |

**3.5. ER Diagram**



**3.6. Data Flow Diagram**

**Level 0:**

Activate Exam

Online Examination System

CANDIDATE

Add, Drop & Edit

Exam, Candidate, Questions & Subjects

ADMIN

Attend Exam &

View Result

**Level 1:**

Check For Login

Request For Login

Response

Reply

ADMIN

admin

LOGIN

Insert Data

Create, Read, Update

and Delete Subject

subject

SUBJECT

CRUD

Reply

Response

Create, Read, Update

And Delete Question

Insert Data

Question

QUESTION

CRUD

Reply

Response

Fig 1.1: Level 1 ADMIN’S side Data Flow Diagram

Check For Login

Request For Login

candidate

LOGIN

CANDIDATE

Reply

Response

Attend

ATTEND

EXAM

Exam / Question

Reply

Response

Read

View

RESULT

Result

Fig 1.2: Level 1 CANDIDATE’S side Data Flow Diagram

**Level 2:**

Delete Data

Update Data

Read Data

Insert Data

Reply

Reply

Reply

Reply

Reply

Check For Login

Response

Request For Login

DELETE

CANDIDATE

UPDATE

CANDIDATE

READ

CANDIDATE

CREATE

CANDIDATE

LOGIN

candidate

candidate

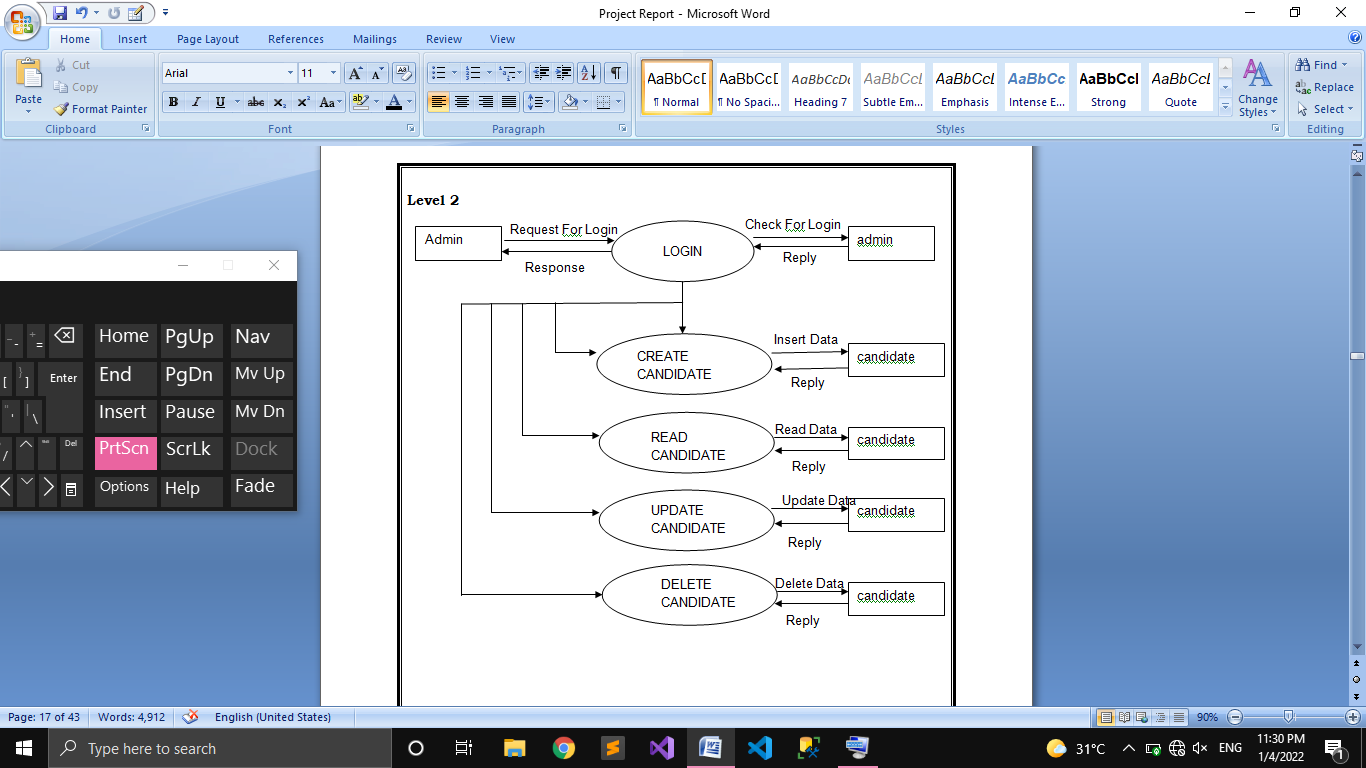
candidate

candidate

admin

Admin

Fig 2.1.1: Level 2 ADMIN’S side Data Flow Diagram



question

question

question

question

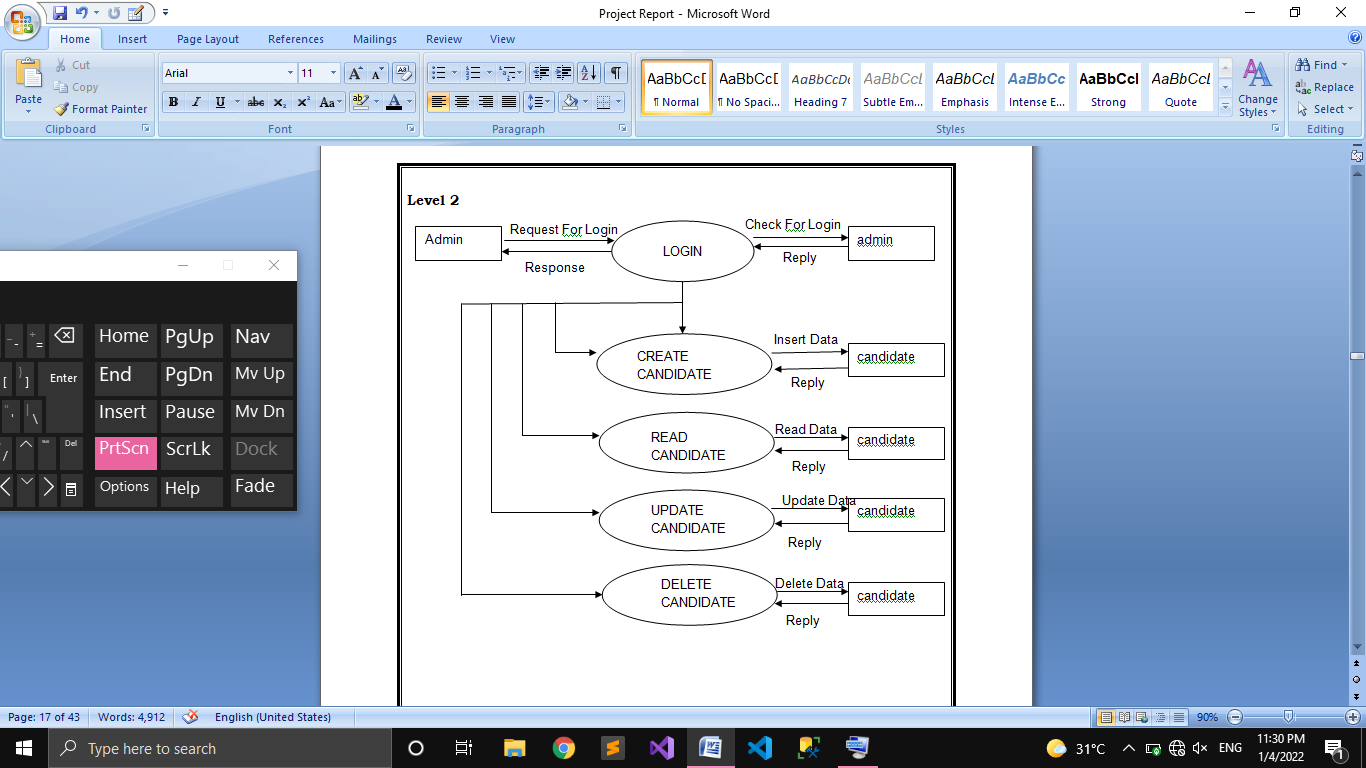
QUESTION

QUESTION

QUESTION

QUESTION

Fig 2.1.2: Level 2 ADMIN’S side Data Flow Diagram



EXAM

EXAM

EXAM

EXAM

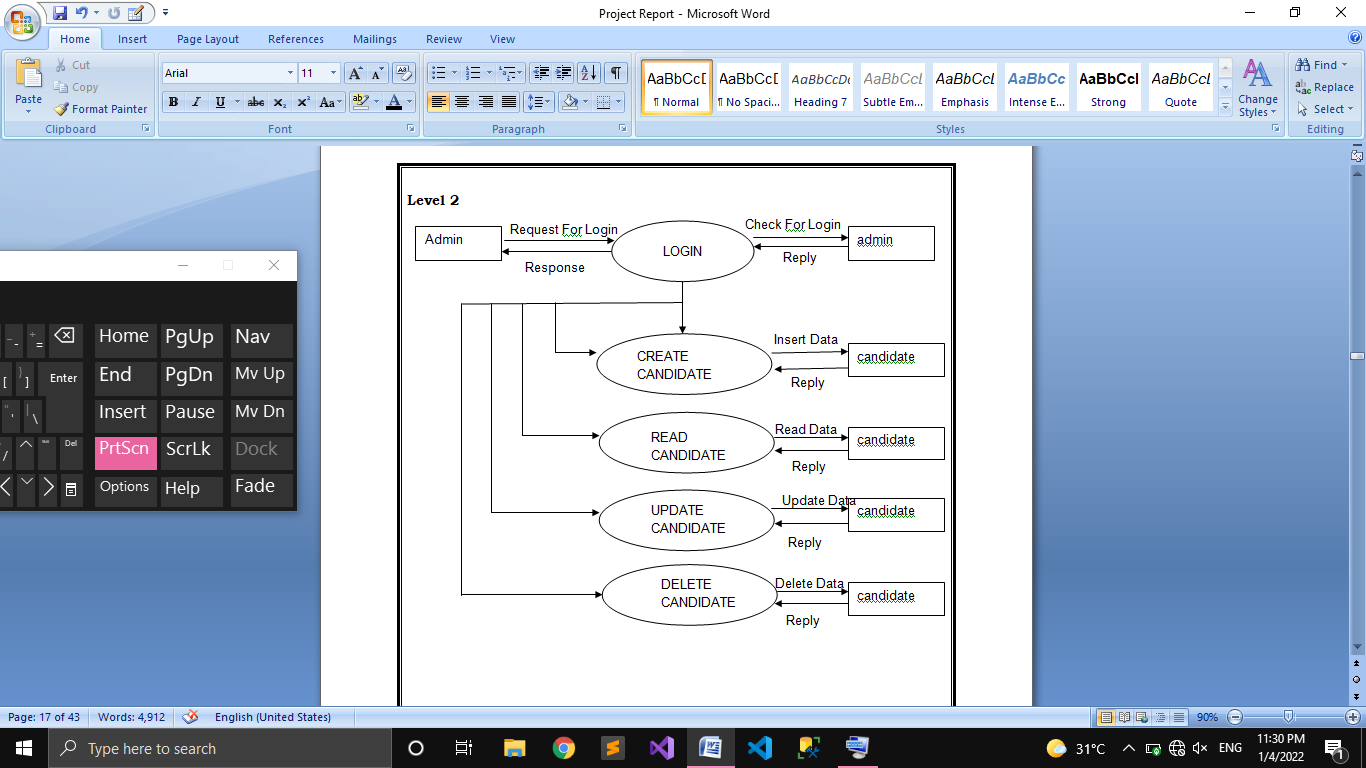
exam

exam

exam

exam

Fig 2.1.3: Level 2 ADMIN’S side Data Flow Diagram



SUBJECT

SUBJECT

SUBJECT

SUBJECT

subject

subject

subject

subject

Fig 2.1.4: Level 2 ADMIN’S side Data Flow Diagram

Check For Login

Request For Login

candidate

LOGIN

CANDIDATE

Reply

Response

Insert Data

Exam / Question

ATTEND

EXAM

Reply

view

result

RESULT

Fig 2.2.1: Level 2 ADMIN’S side Data Flow Diagram

1. SYSTEM DEVELOPMENT

4.1. Process description

The procedure of “Online Examination” begins with a selection among candidates. After login, the student is the directed to the page showing available exams and by selecting one among them he/she can attend the exam. The student is provided a set of questions and an amount of time to answer the questions. Different subjects hold different set of questions and each question have 4 choices with answer. After the exam is over the form will automatically submitted and the result is generated.

4.2. Source code

Timer

@model Online\_Exam\_Project\_Main.Models.Questions\_tbl

@{

ViewBag.Title = @ViewBag.score;

}

<br>

<h1></h1>

<div id="clockdiv">

<h1 style="color: red">

<div class="minutes" style="float: left"> </div>

<p style="float: left">:</p>

<div class="seconds"></div>

</h1>

</div>

<script>

function getTimeRemaining(endtime) {

var t = Date.parse(endtime) - Date.parse(new Date());

var seconds = Math.floor((t / 1000) % 60);

var minutes = Math.floor((t / 1000 / 60) % 1);

return {

'total': t,

'minutes': minutes,

'seconds': seconds

};

}

function initializeClock(id, endtime) {

var clock = document.getElementById(id);

var minutesSpan = clock.querySelector('.minutes');

var secondsSpan = clock.querySelector('.seconds');

function updateClock() {

var t = getTimeRemaining(endtime);

minutesSpan.innerHTML = ('0' + t.minutes).slice(-2);

secondsSpan.innerHTML = ('0' + t.seconds).slice(-2);

if (t.total <= 0) {

clearInterval(timeinterval);

}

}

updateClock();

var timeinterval = setInterval(updateClock, 1000);

}

var deadline = new Date(Date.parse(new Date()) + 60 \* 60 \* 1000);

initializeClock('clockdiv', deadline);

// ...................................

var \_second = 1000;

var \_minute = \_second \* 60;

var \_hour = \_minute \* 60;

var \_day = \_hour \* 24;

var timer;

var end = new Date().getTime() + (60 \* \_second);

//

function showRemaining() {

var now = new Date();

var distance = end - now;

if (distance < 0) {

clearInterval(timer);

document.getElementById('countdown') +.innerHTML = 'EXPIRED!';

return;

}

var days = Math.floor(distance / \_day);

var hours = Math.floor((distance % \_day) / \_hour);

var minutes = Math.floor((distance % \_hour) / \_minute);

var seconds = Math.floor((distance % \_minute) / \_second);

document.getElementById('countdown').innerHTML = days + 'days ';

document.getElementById('countdown').innerHTML += hours + 'hrs ';

document.getElementById('countdown').innerHTML += minutes + 'mins ';

document.getElementById('countdown').innerHTML += seconds + 'secs';

}

timer = setInterval(showRemaining, 1000);

</script>

<div>

@Html.Partial("\_QuestionPartial")

</div>

Admin Login

@model Online\_Exam\_Project\_Main.Models.Admin\_tbl

@{

ViewBag.Title = "Login";

Layout = "~/Views/Shared/\_Layout.cshtml";

}

<h2>Admin Login</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<hr />

@Html.ValidationSummary(true)

<div class="form-group">

<Label class = "control-label col-md-2">Admin Name:</Label>

<div class="col-md-10">

@Html.EditorFor(model => model.admin\_name)

@Html.ValidationMessageFor(model => model.admin\_name)

</div>

</div>

<div class="form-group">

<Label class="control-label col-md-2">Admin Password:</Label>

<div class="col-md-10">

<input type="password" name="admin\_password" autocomplete="off"/>

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<input type="submit" value="Login" class="btn btn-default" />

</div>

</div>

</div>

}

<div>

@Html.ActionLink("New Admin", "Create")

</div>

@section Scripts {

@Scripts.Render("~/bundles/jqueryval")

}

Candidate Register

@model Online\_Exam\_Project\_Main.Models.Candidate\_tbl

@{

ViewBag.Title = "Create";

}

<h2>Add New Candidate</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<hr />

@Html.ValidationSummary(true)

<div class="form-group">

<label class = "control-label col-md-2">Candidate Name:</label>

<div class="col-md-10">

@Html.EditorFor(model => model.candidate\_name)

@Html.ValidationMessageFor(model => model.candidate\_name)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Candidate Password:</label>

<div class="col-md-10">

@Html.EditorFor(model => model.candidate\_password)

@Html.ValidationMessageFor(model => model.candidate\_password)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Candidate Email:</label>

<div class="col-md-10">

@Html.EditorFor(model => model.candidate\_email)

@Html.ValidationMessageFor(model => model.candidate\_email)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Candidate Contact:</label>

<div class="col-md-10">

@Html.EditorFor(model => model.candidate\_contact)

@Html.ValidationMessageFor(model => model.candidate\_contact)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Candidate Institution:</label>

<div class="col-md-10">

@Html.EditorFor(model => model.candidate\_institution)

@Html.ValidationMessageFor(model => model.candidate\_institution)

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<input type="submit" value="Add" class="btn btn-default" />

</div>

</div>

</div>

}

<div>

@Html.ActionLink("Back to List", "Index")

</div>

@section Scripts {

@Scripts.Render("~/bundles/jqueryval")

}

Add Questions

@model Online\_Exam\_Project\_Main.Models.Questions\_tbl

@{

ViewBag.Title = "Create";

}

<h2>Create Questions</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<hr />

@Html.ValidationSummary(true)

<div class="form-group">

<label class = "control-label col-md-2">Question Id</label>

<div class="col-md-10">

@Html.EditorFor(model => model.qus\_id)

@Html.ValidationMessageFor(model => model.qus\_id)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Admin Id</label>

<div class="col-md-10">

@Html.DropDownList("admin\_id", String.Empty)

@Html.ValidationMessageFor(model => model.admin\_id)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Exam Id</label>

<div class="col-md-10">

@Html.DropDownList("exam\_id", String.Empty)

@Html.ValidationMessageFor(model => model.exam\_id)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Question Title</label>

<div class="col-md-10">

@Html.EditorFor(model => model.qus\_title)

@Html.ValidationMessageFor(model => model.qus\_title)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Answer 1</label>

<div class="col-md-10">

@Html.EditorFor(model => model.ans1)

@Html.ValidationMessageFor(model => model.ans1)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Answer 2</label>

<div class="col-md-10">

@Html.EditorFor(model => model.ans2)

@Html.ValidationMessageFor(model => model.ans2)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Answer 3</label>

<div class="col-md-10">

@Html.EditorFor(model => model.ans3)

@Html.ValidationMessageFor(model => model.ans3)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Answer 4</label>

<div class="col-md-10">

@Html.EditorFor(model => model.ans4)

@Html.ValidationMessageFor(model => model.ans4)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Correct Answer</label>

<div class="col-md-10">

@Html.EditorFor(model => model.correct\_ans)

@Html.ValidationMessageFor(model => model.correct\_ans)

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<input type="submit" value="Create" class="btn btn-default" />

</div>

</div>

</div>

}

<div>

@Html.ActionLink("Back to List", "Index")

</div>

@section Scripts {

@Scripts.Render("~/bundles/jqueryval")

}

Exam Details

@model Online\_Exam\_Project\_Main.Models.Exam\_tbl

@{

ViewBag.Title = "Details";

}

<h2>Details</h2>

<div>

<hr />

<dl class="dl-horizontal">

<dt>

<label>Exam Title</label>

</dt>

<dd>

@Html.DisplayFor(model => model.exam\_title)

</dd>

<dt>

<label>Subject Title</label>

</dt>

<dd>

@Html.DisplayFor(model => model.Subject\_tbl.subject\_title)

</dd>

</dl>

</div>

<p>

@Html.ActionLink("Edit", "Edit", new { id = Model.exam\_id }) |

@Html.ActionLink("Back to List", "Index")

</p>

Edit Result

@model Online\_Exam\_Project\_Main.Models.Result\_tbl

@{

ViewBag.Title = "Edit";

}

<h2>Edit</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<hr />

@Html.ValidationSummary(true)

@Html.HiddenFor(model => model.result\_id)

<div class="form-group">

<label class = "control-label col-md-2">Exam Id</label>

<div class="col-md-10">

@Html.DropDownList("exam\_id", String.Empty)

@Html.ValidationMessageFor(model => model.exam\_id)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Candidate Id</label>

<div class="col-md-10">

@Html.DropDownList("candidate\_id", String.Empty)

@Html.ValidationMessageFor(model => model.candidate\_id)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Total</label>

<div class="col-md-10">

@Html.EditorFor(model => model.total)

@Html.ValidationMessageFor(model => model.total)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Grade</label>

<div class="col-md-10">

@Html.EditorFor(model => model.grade)

@Html.ValidationMessageFor(model => model.grade)

</div>

</div>

<div class="form-group">

<label class="control-label col-md-2">Quality</label>

<div class="col-md-10">

@Html.EditorFor(model => model.quality)

@Html.ValidationMessageFor(model => model.quality)

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<input type="submit" value="Update" class="btn btn-default" />

</div>

</div>

</div>

}

<div>

@Html.ActionLink("Back to List", "Index")

</div>

1. SYSTEM IMPLEMENTATION

5.1 Testing

Testing is the process of examining the software to compare the actual behavior with that of the expected behavior. The major goal of the software is to demonstrate that faults are not present. In order to achieve this goal the tester executes the program with the intent of finding errors. Through testing cannot show absence of errors but by not showing their presence it is considered that these are not present.

System testing is the first stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operations commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct and the goal will be successfully achieved. A series of testing are performed for the proposed system before the proposed system is ready for user acceptance testing.

Levels of testing

1. Unit testing

2. Integration testing

3. Validation testing

4. Output testing

Unit testing

In this each module is tested individually before integrating it to final system. Unit test focuses verification in the smallest unit of the software design in each module. This is known as module testing as here each module is tested to check whether it is produced the desire output and to see if any error occurs.

Integration testing

Integration testing sometimes called integration and testing, abbreviation I&T) is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test to plan to those aggregates, and delivers as its output the integrated system ready for system testing. The purpose of integration testing is to verify functional, performance, and reliability requirements placed on major design items.

Validation testing

In software project management, software testing and software engineering, validation is the process of checking that a software system meets specifications and that it fulfills its intended purpose. The errors which are uncovered during integration testing are connected during this phase

Output testing

No system could be useful if it does not produce the required output in the specific format. Output testing is performed to ensure the correctness of the output and its format. The output generated or displayed by the system is tested asking the users about the format required by them.

5.2. System implementation

Implementation is the stage in the project where theoretical design is turned into a working system and is giving confidence on the new system for the users which will work efficiently. It involves careful planning, investigation of the current system and it’s constrains on implementations, design of methods to achieve the changeover, an evolution, of change over methods. Apart from planning major tasks of preparing the implementation coordination committee based of the system. According to this are to be carried out discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system.

Implementation is the final and important phase. The most critical stage in achieving a successful new system and giving the users confidence that the new system will work and be efficient. The system can be implement only after thorough testing is done and if found to working according to the specification.

5.3. Security

In any organizations data is the most important element and the main issue related to it is the security of those valuable data. One of the major areas in development process of a system is providing security to all its data in an efficient way. In my work, as it is for an institution it is tightly protected by password system. Only the admin and user can use it with userid and password. The database server is equipped with efficient password security system. So the entire system is provided with tight security and I am sure that data should not be hacked by an unauthorized person

1. CONCLUSION

In this new world, Computers have made their presence in each field. Computer simplified procedures in almost all fields. In my project, I provide almost all facilities for entering the details of the employee, which is presently done manually. The facilities include adding of new user in the database, and other necessary details like finding salary, printing salary, removing employee, updating details etc. The system also provides a platform to perform fetching employee details and so on.

This system reduces the effort and saves time and has a user friendly platform for the user. Users can efficiently use the system by accepting the appropriate warning messages and they can ensure that the details are stored into the database by reading the messages provided by the system also provide by the system. I have tried my level best to make a perfect user friendly system and I’m sure that it is with less percentage of errors. I expect the implementation of this system will provide them an active working environment and can reduce the manpower. I expect implementing my system will improve the efficiency of the institutions for managing employees.

1. APPENDIX

7.1. Sample inputs and sample output screens

1. BIBLIOLOGY

**Electronic Materials**

C sharp

<http://www.w3schools.com/cs/index.php>

<http://www.tutorialspoint.com/csharp/index.htm>

<http://www.javatpoint.com/c-sharp-tutorial>

YouTube

**Books Referred**

“CLR via C#”, Jeffrey Richter, Microsoft Press; 4 edition

“C# 5.0 in a Nutshell”, Ben Albahari, Shroff; Fifth edition.