# E-Learn

Guided by,

Mr. Gireesh T K

Assistant Professor
Dept. of CSE

Presented by,

Group No: 6

| SI no: | Name of the member | Reg. No: |
|---|---|---|
| 1 | Anshida Sherin M | AWH18CS007 |
| 2 | Arathi P P | AWH18CS010 |
| 3 | Nabeel Ahmed | AWH18CS033 |
| 4 | Muhammed Swalih M P | LAWH18CS050 |

DEPT. OF COMPUTER SCIENCE AND ENGINEERING
AWH ENGINEERING COLLEGE

# About the Group Members….

TEAM LEADER: MUHAMMED SWALIH MP

| ACTIVITY | ASSIGNED TO (NAME OF THE MEMBER) |
|---|---|
| REQUIREMENT ANALYSIS | ARATHI P P |
| DESIGNING | NABEEL AHMED |
| IMPLEMENTATION | MUHAMMED SWALIH MP |
| TESTING | ANSHIDA SHERIN M |
| DOCUMENTATION | ANSHIDA SHERIN M |

# CONTENTS

# 1. OBJECTIVE

# OBJECTIVES OF THE PROJECT

- E-Learn web app provides complete academic activities under a single platform.
- It consist of an e-learning system that facilitates learning in a college environment.
- It enhances the quality of learning & teaching and improves efficiency & effectiveness.
- It improve user accessibility and time flexibility.

# 2. INTRODUCTION

# INTRODUCTION

- E-Learning is an emerging methodology of modern education .
- It can deliver more value at very less cost than any other traditional mode of education.
- It makes the process of education more student-centered , creative , and flexible.
- E-Learning web app is an interactive webpage that allows learners to input their data and get expected results through interactions.

# 3. LITERATURE SURVEY

# Byju’s App

Reference:

- https://www.comparably.com/companies/byju-s/mission
- https://aws.amazon.com/solutions/case-studies/byjus

Aim:

- Their aim is to become one of the most preferred education technology platforms across the globe.
- They implement one-to-one learning that addresses every child's learning needs.
- And allows students to be holistically involved in their education and active life long leaners.

Methodology :

1) Visualization with 3D animations and in air projections

2) Byju's uses amazon web services (aws) as the cloud platform for it's website and mobile app

3) PostgreSQL uses as primary database

4) Technologies used

- Amazon s3
- Amazon EC2
- Amazon RDS
- Amazon RedShift

Advantages :

interactive online classes

Easy registration using mobile number , email id or

social media account

Monthly or weakly tests to sharpen students

Disadvantages :

High Expensive

No provision for college based classes

No live classes

# Google classroom

Reference:

- https://support.google.com/edu/classroom/answer/6020279?hl=en
- Aim:
- Users can use Classroom in their schools to streamline assignments, boost collaboration, and foster communication
- Classroom is available on the web or by mobile app

- Users can use Classroom with many tools that they already use, such as Gmail, Google Docs, and Google Calendar.

Methodology :

- It uses a similar model as Dr. Ruben Puentedura’s SAMR model.

Advantages :

- Free of cost
- Ease of use
- High privacy
- Unlimited Class size

Disadvantages :

- No provision for live classes
- It lacks Attendance management system

# Google Meet

Reference:

- https://en.m.wikipedia.org/wiki/Google_Meet

Aim:

- Google describes Meet as "a video meeting experience with one goal: make joining meetings effortless".
- The company wanted to improve Hangouts to make it easier and faster for people start and join video conferences

Methodology :

- Technologies used
- WebRTC
- It stands for Web Real-Time Communication,
- It is a networking technology introduced in 2011 by Google
- It enables real-time audio, video, and data transmission across the web and native browsers

Advantages :

- Two-way and multi-way audio and video calls with a resolution up to 720p
- An accompanying chat
- Call encryption between all users[8]
- Noise-cancelling audio filter
- Low-light mode for video

Disadvantages :

- No Academic activities
- only for video conferencing

# Mobile application for students attendance and mark management system

Reference:


- Somasundaram, V., Kannan. M, Sriram, V., 2016,“Mobile based Attendance Management System”, Indian Journal of Science and Technology, 9 (35),pp. 1-4.
- Avinaash Ram, S.P., Albert Mayan, J., 2015,“Mobile attendance management and employee registration”

Aim:

- To design a mobile application for student attendance and mark management system
- Allows the users to mark attendance through mobile devices.
- Allows the teachers to mark and edit the attendance.

Methodology :

- Mobile application is developed using Sun Java Wireless Toolkit 2.5.2_01

WORK FLOW :

STAFF

- Enter, edit and update the student mark attendance details
- View the attendance
- View mark details

STUDENT

- View the attendance
- View mark details

PARENTS

- View the attendance
- View mark details

Advantages :

- Accuracy
- Efficiency
- Visibility
- Real-time tracking
- Security

Disadvantages :

- Less features
- Absence of proper network connection make it difficult to access.

# SMART UNIVERSITY-STUDENT INFORMATI0N MANAGEMNT SYSTEM

Reference:


- S.R.Bharamagoudar, Geeta R.B., S.G.Totad “Web Based Student Information Management System” , International Journal of Advanced Research in Computer and Communication Engineering -June 2013, ISSN : 2319-5940
- Sandeep Kumar, Mohammed Abdul Qadeer, Archana Gupta, “Location Based Services using Android” , IEEE- 2009

Aim:

- To design Student Information Management System
- It is a cloud based system for university management

Methodology :

- SUSIMS is a cross platform solution supporting Android, IOS and Web.
- Data is stored is MySQL data base and php language is used to make backend scripting language .

- student information management system are
- partitioned into modules.
- Each module represents activities carried out by each department.
- Attendance module
- Placement module
- Result module
- E-notes module etc

Advantages :

- It overcomes the limitations of the desktop based system as our cloud can be accessed by Android, IOS and Web.
- Improved communication gap between university and students with help of instant notification

Disadvantages :

- No provisions for live classes
- Lack of assignment and grading options

# COMPARISON OF RELATED WORKS

| Ref. No: | Methodology Used | Advantages | Disadvantages |
|---|---|---|---|
| [1] | Byju’ s app | Interactive online classes<br>Easy registration | High expensive<br>No provision for online based classes |
| [2] | Google Classroom | Free of cost<br>Ease of use | No attendance system<br>No live classes |
| [3] | Mobile application for students attendance and mark management system | Accurate<br>Real time tracking<br>secure | No feature that support classes |
| [4] | Google meet | Two-way and multi-way audio and video calls with a resolution up to 720p | No other academic activities |
| [5] | Smart University-Student Information Management System | Huge amount of papers saved .hence eco-friendly | No live classes<br>No support for test |

# 4. PROBLEM STATEMENT

# PROBLEM STATEMENT

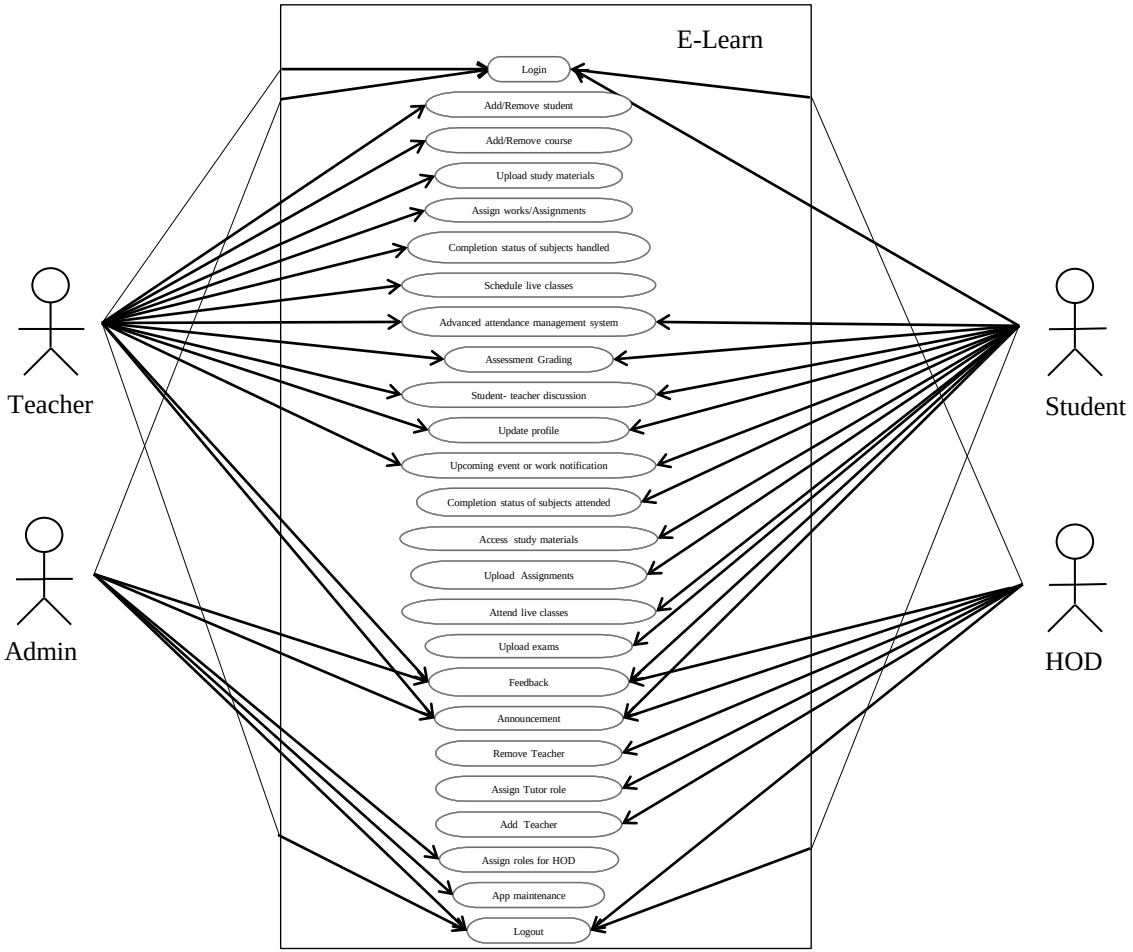
- The goal of building an e-learning web app with extra features along with the basic ones make online education more promising.
- It is a non-trivial task to effectively discover and evaluate the features of each system providing e-learn facilities.
- Each of the e-learn system may lack some of the features which the user mostly needed.
- An e-learning system which focuses on good quality teaching, periodic class works, attendance & security , all through a user friendly interface at free of cost will be mostly adopted.

# 5. E-Learn

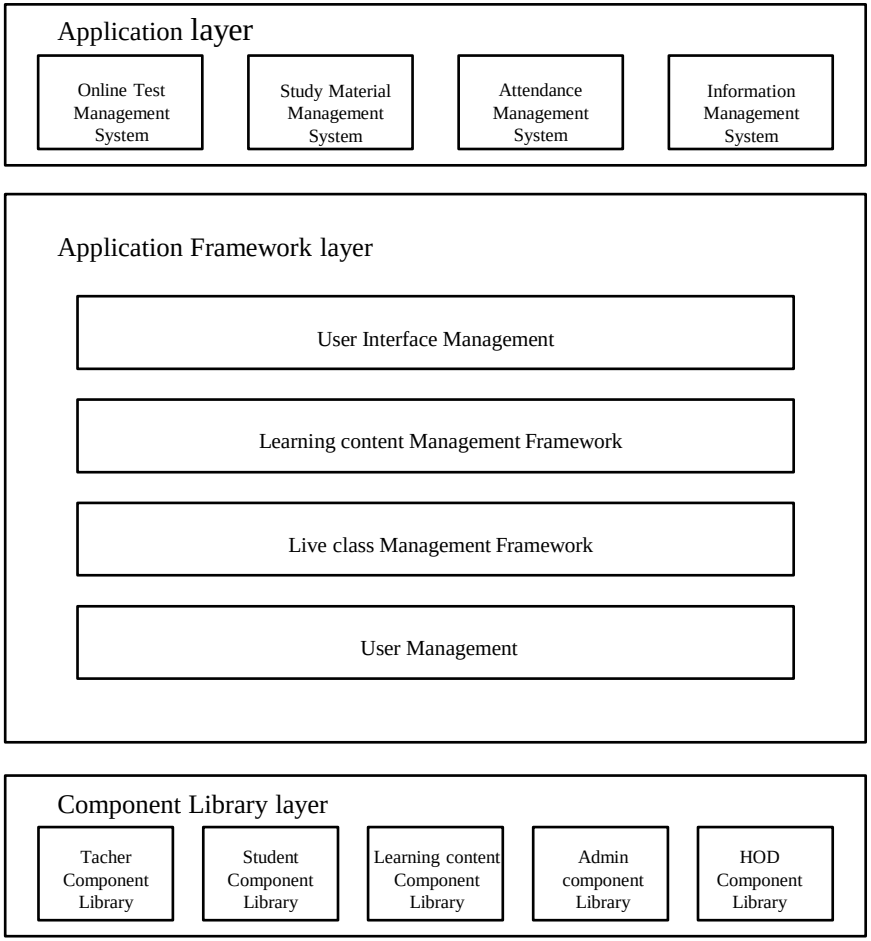
# 5. E-Learn

- Use-case Diagram
- Software Architecture
- Working Principle
- Data Flow Diagram
- ER Diagrams

# 5.1 USE CASE DIAGRAM



# 5.2 SOFTWARE ARCHITECTURE (Detailed)



# 5.3 WORKING PRINCIPLE

- E-learning domain software architecture is divided into three levels
- They are component library layer, application framework layer, and application layer, respectively, from bottom to top

Component library layer

- This layer is the basis of the whole E-learning software architecture
- It is a group of reusable software units that had been tested in other projects.

# 5.3 WORKING PRINCIPLE

- It mainly includes
- the student component library,
- the learning content component library,
- the instructor component library
- the admin component library
- the hod component library

# 5.3 WORKING PRINCIPLE

- Application framework layer
- The application framework layer is the gathering of all the frameworks
- From the perspective of E-learning, we could integrate every component that supported the instructional model/learning model
- The application framework layer also provides the management and maintenance of the system

# 5.3 WORKING PRINCIPLE

- This layer includes
- User interface management
- Learning content management framework
- Live class management framework
- User management framework
- The user interface management framework is responsible for all the UI element
- user experience of the web app while the user management framework manages the student and teachers
- learning content management framework manages all the learning content

# 5.3 WORKING PRINCIPLE

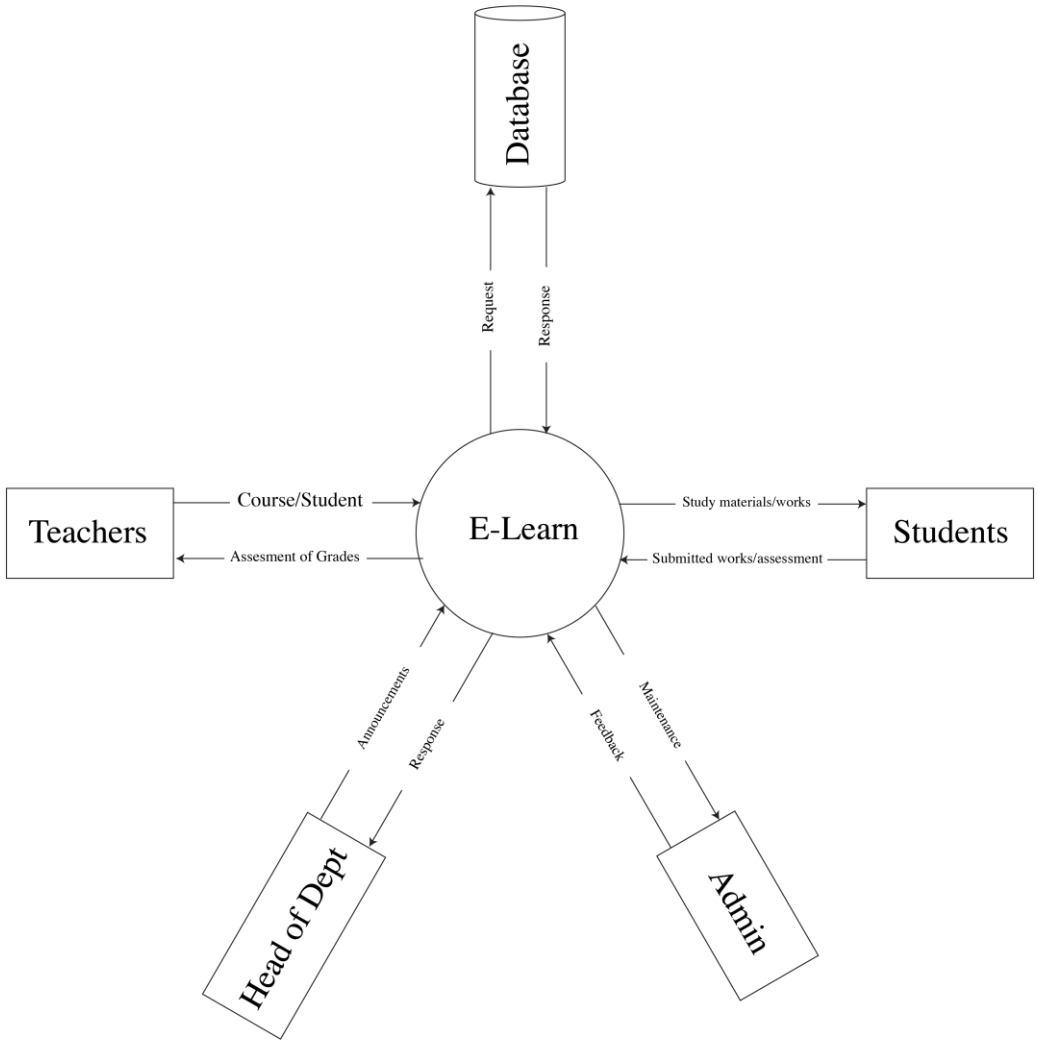
- Application layer
- This layer is a group of individualized and custom-made E-learning software architecture developed through choosing the framework and components
- It includes:
- An online testing management system
- A study material management system,
- An educational information management system
- Advanced attendance management system

# 5.3 WORKING PRINCIPLE

- These three layers combine to create The software architecture
- Ui elements are includes in public framework
- Request passed by public framework is passed to abstract framework
- Abstract framework passes a query to component library layer which respond to the query request.
- Application layer provides the necessary systems for the functioning

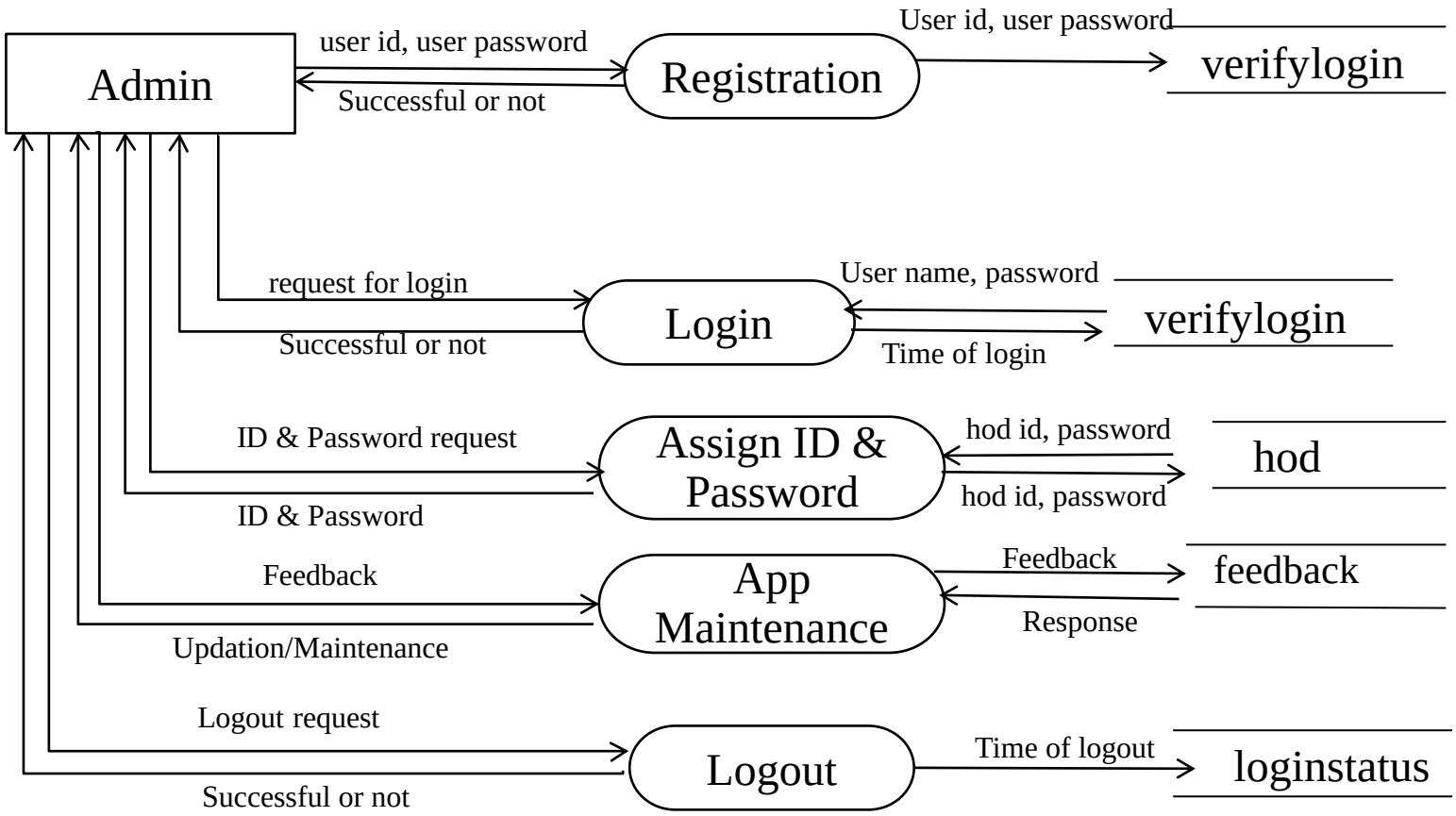
# 5.4 DATA FLOW DIAGRAM

LEVEL 0



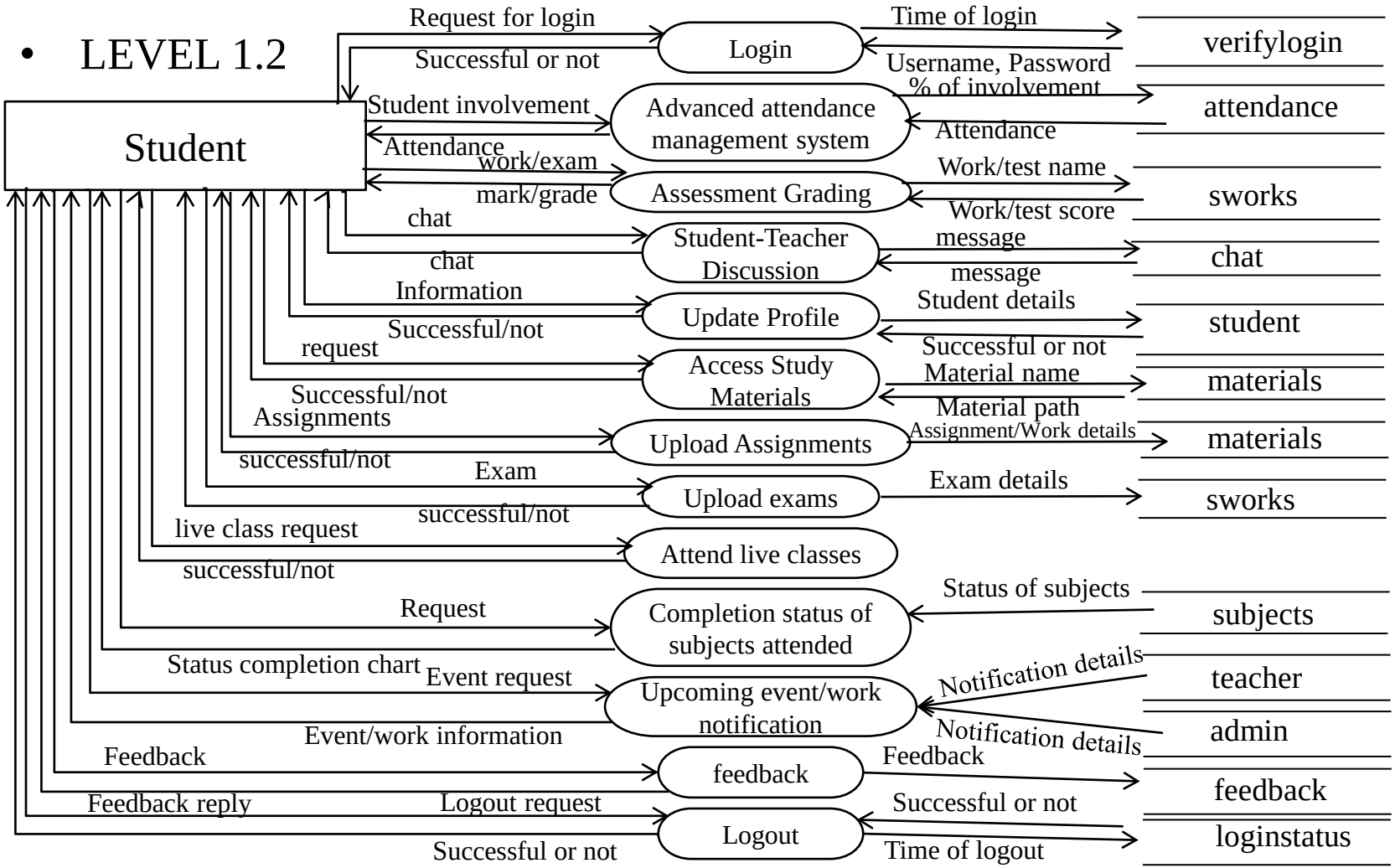
# 5.4 DATA FLOW DIAGRAM(contd.)

- LEVEL 1.1



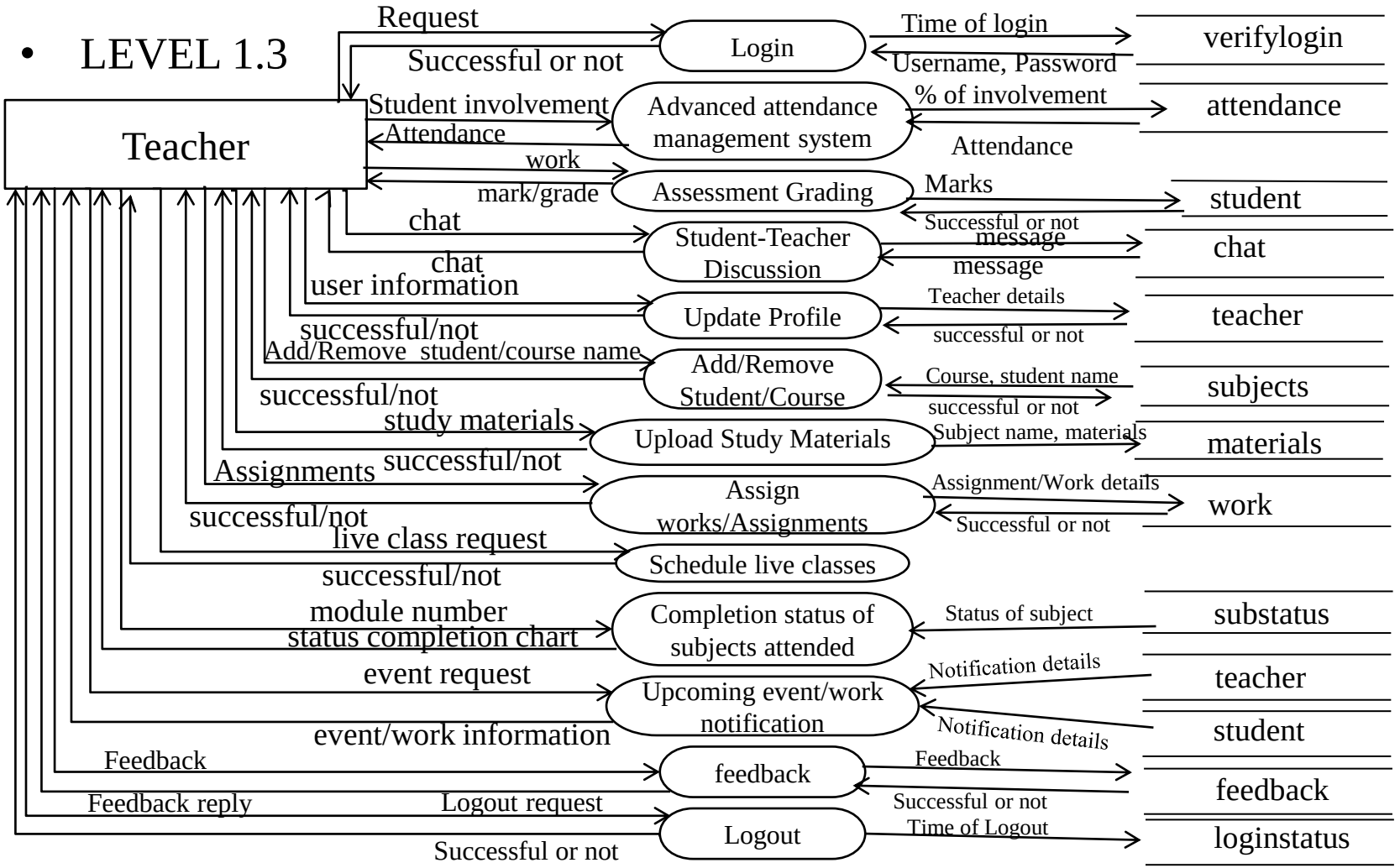
# 5.4 DATA FLOW DIAGRAM(contd.)

- LEVEL 1.2



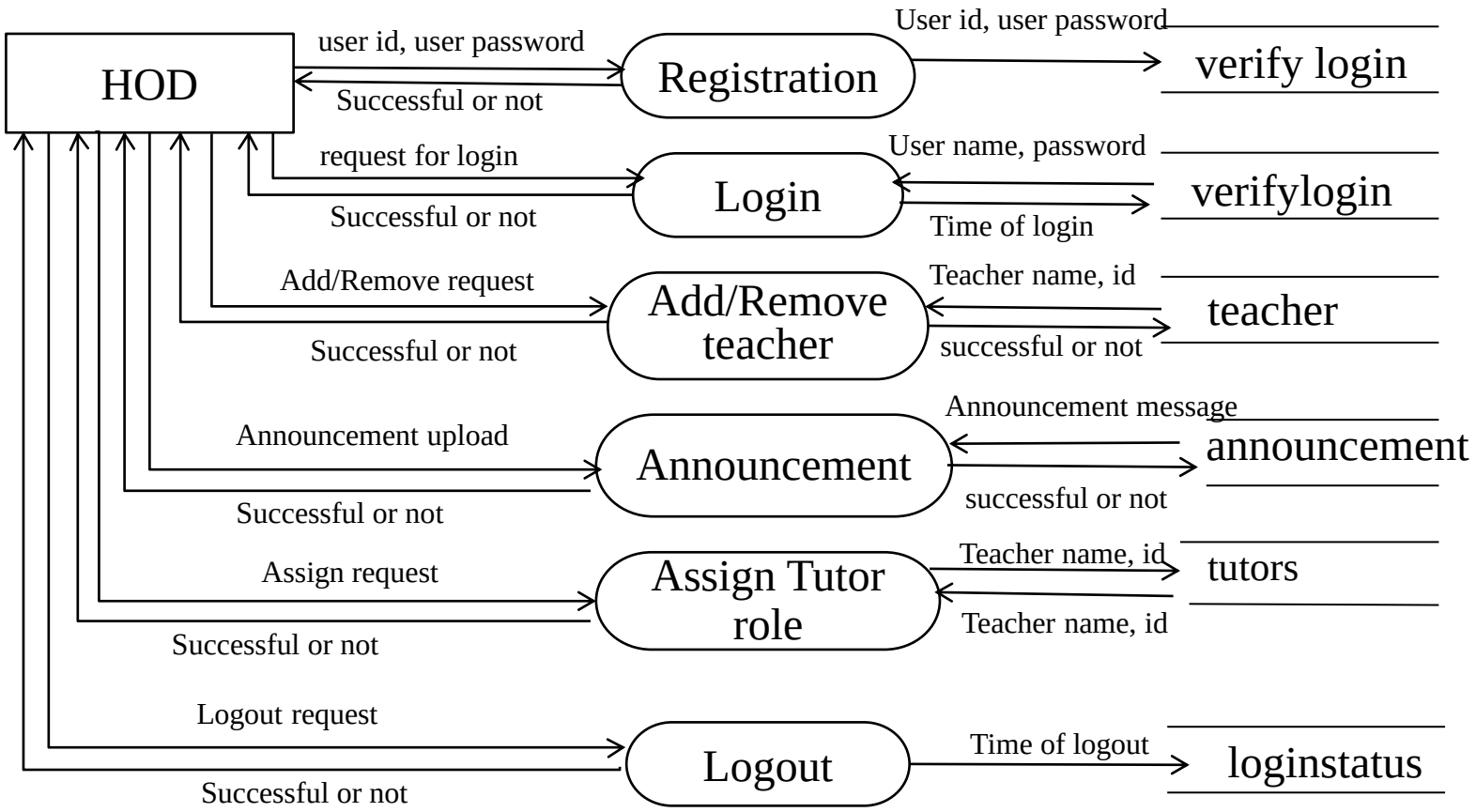
# 5.4 DATA FLOW DIAGRAM(contd.)

- LEVEL 1.3

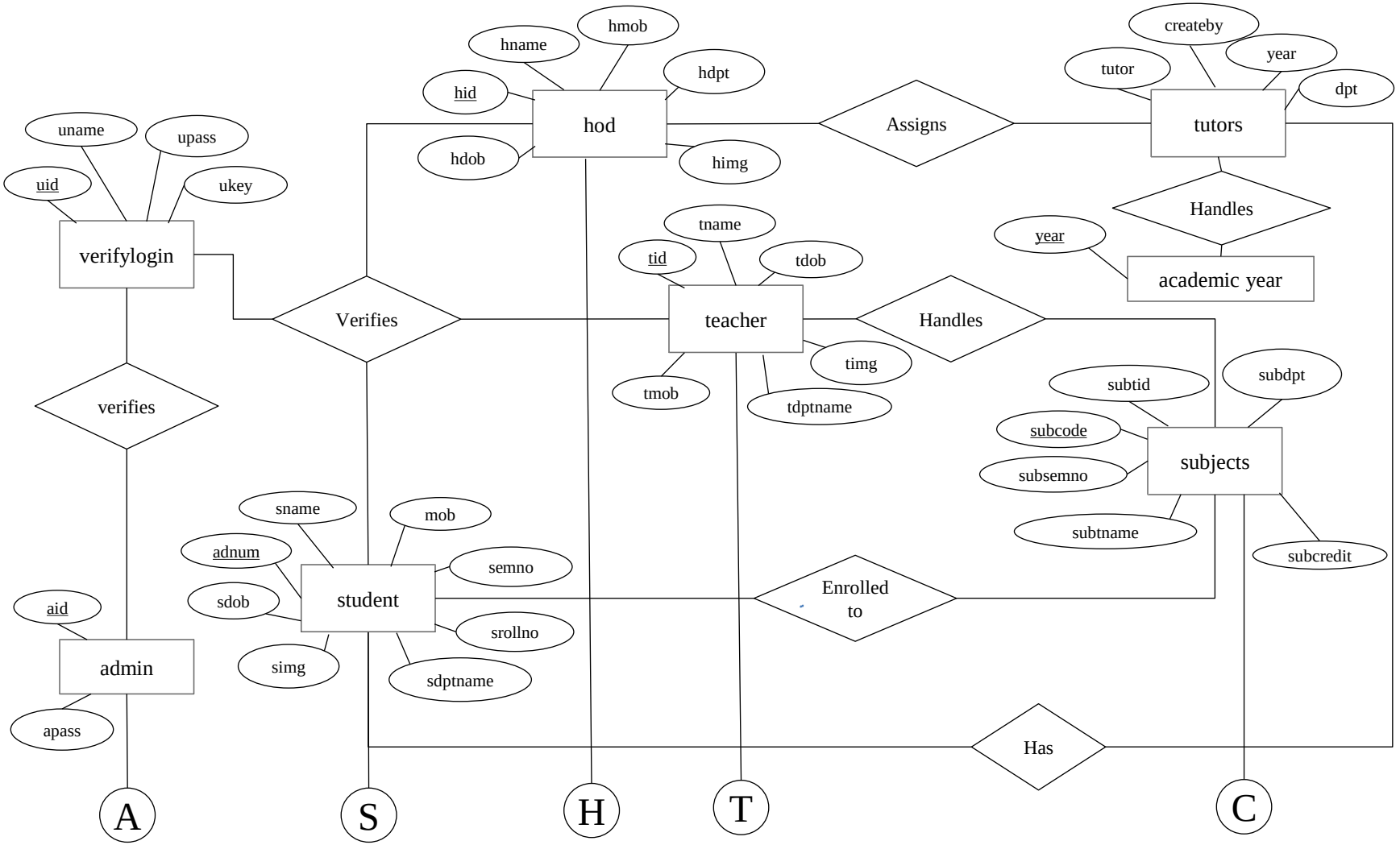


# 5.4 DATA FLOW DIAGRAM(contd.)

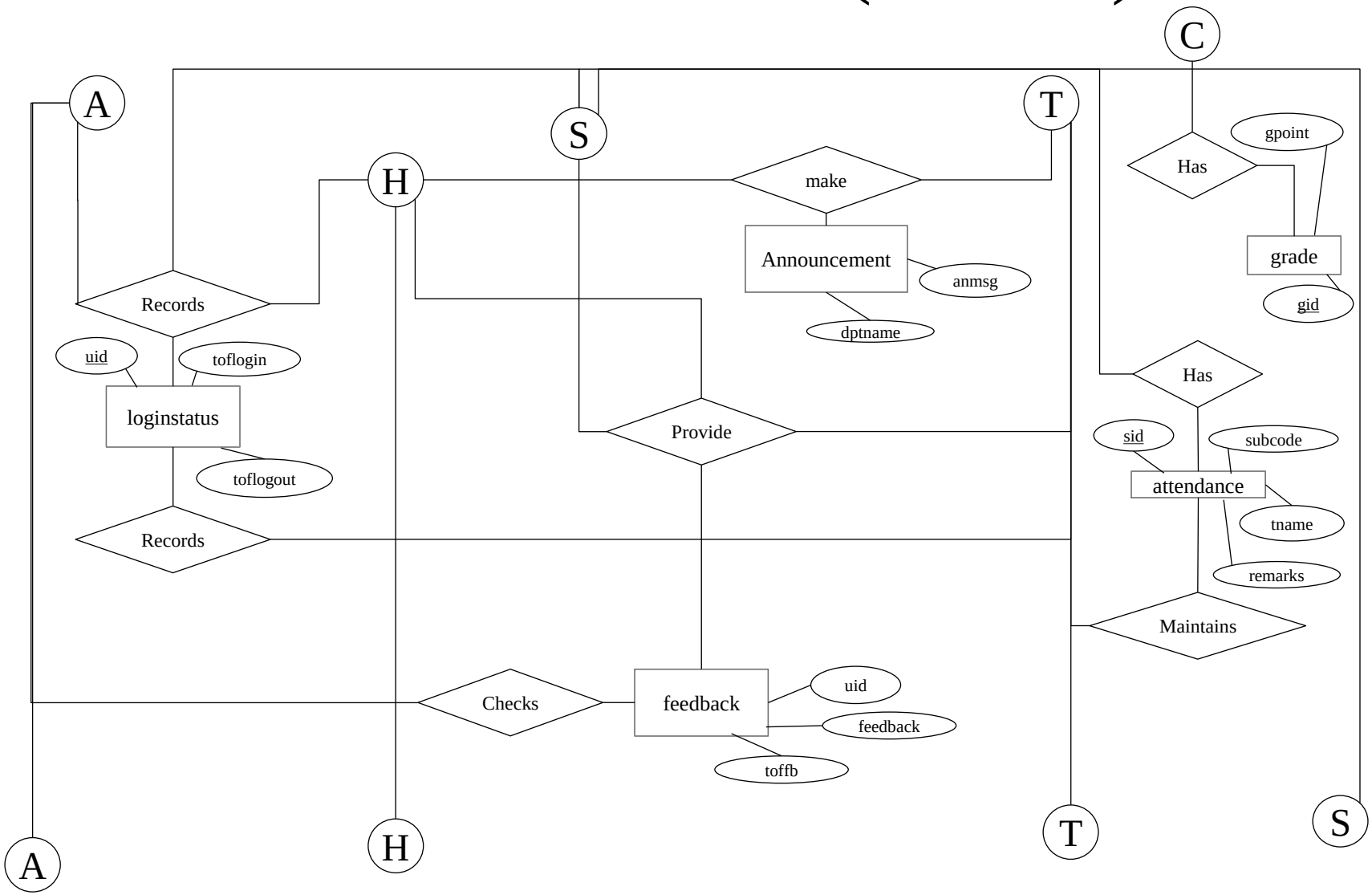
- LEVEL 1.4



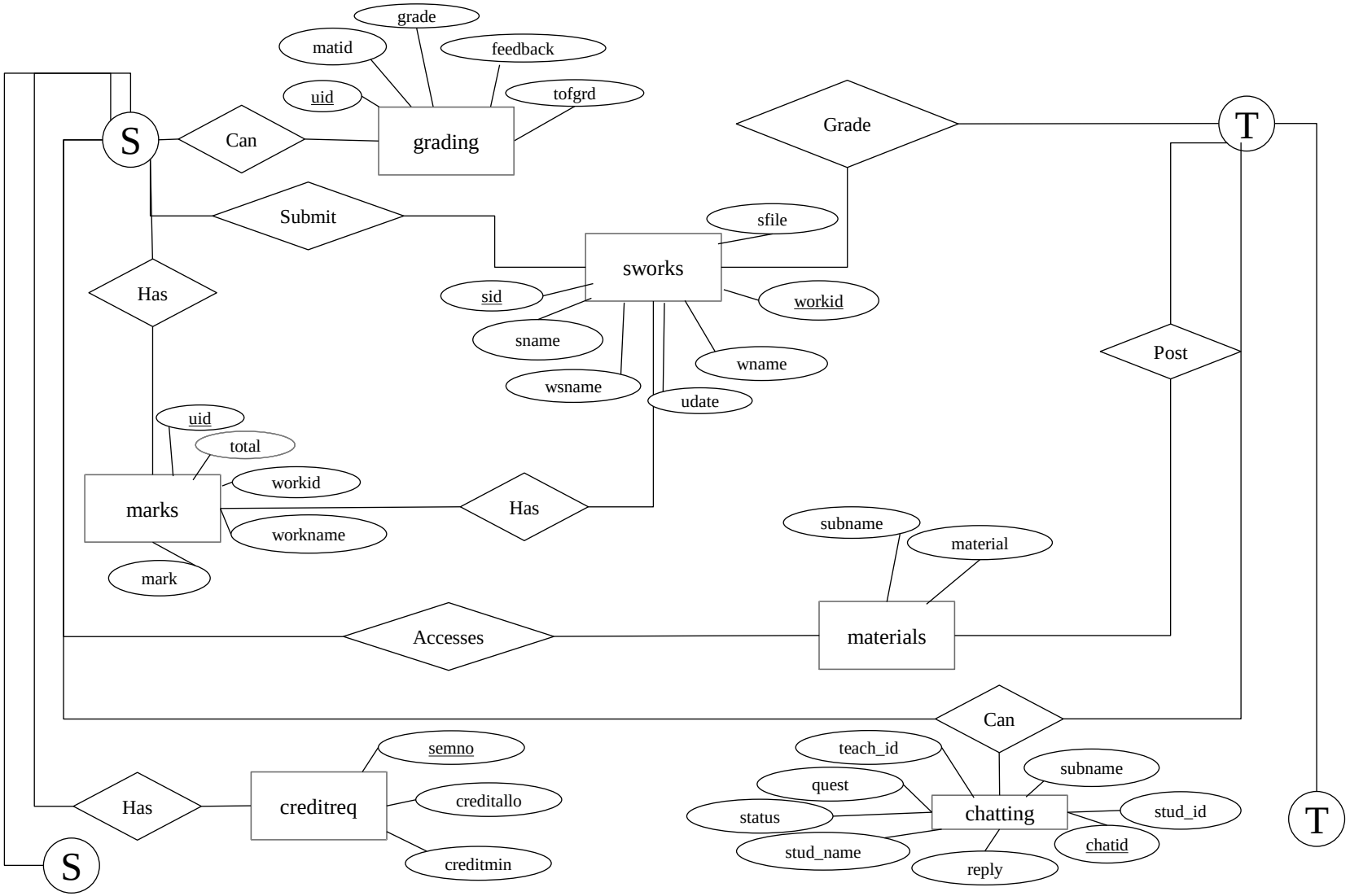
# 5.5 ER DIAGRAM



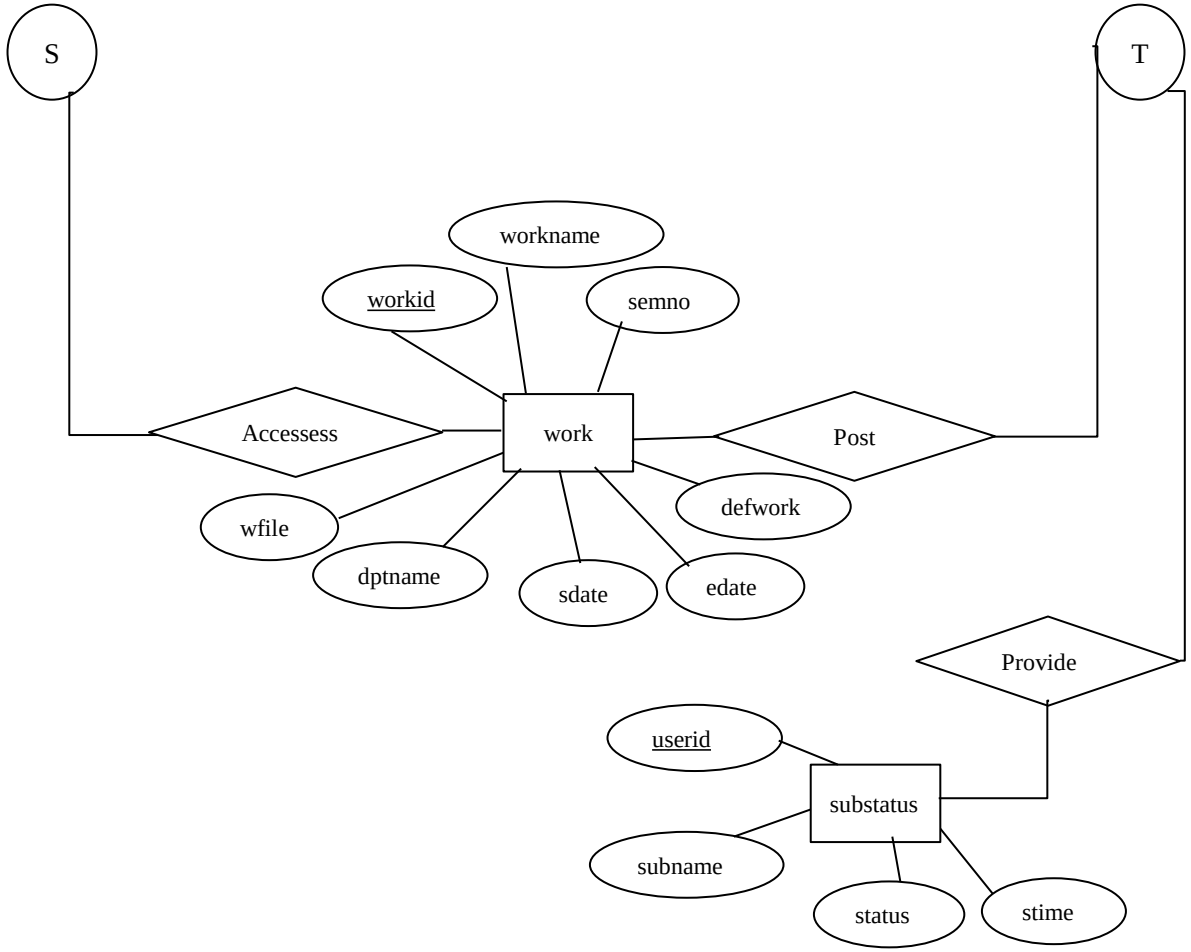
# 5.5 ER DIAGRAM(Contd..)



# 5.5 ER DIAGRAM(Contd..)



# 5.5 ER DIAGRAM(Contd..)



# 6. DESIGN

# 6. DESIGN

- Input Design
- Output Design
- Database Design

# 6.1 INPUT DESIGN

- **Login credentials :**
- The username and user password are provided as input to the system
- **Profile :**
- Details in the profile
- **Works / Exams :**
- These are provided by the teachers as  input to the system
- Students upload works/exams as the input to the system.

# 6.1 INPUT DESIGN(contd.)

- **Marks :**
- Marks are provide by the teachers
- **Search :**
- Users are able to search contents in the system by providing search content.
- **Chat :**
- The teacher-student chat are input to the system.

# 6.1 INPUT DESIGN(contd.)

- **Feedback :**
- The teacher, student can provide feedback, which are input to the system.
- **Event/News :**
- They are provided by the Admin, teachers as the input to the system.
- **Study Materials :**
- Teachers upload the study materials as the input to the system.

# 6.2 OUTPUT DESIGN

- **Attendance :**
- Attendance are calculated by system and shown as output.
- **Study Materials :**
- The study materials uploaded by the teachers are output from the system.
- **Marks :**
- Marks are shown as the output of the system.

# 6.2 OUTPUT DESIGN(contd.)

- **Work/Exams :**
- These are output of the system.
- **Chat :**
- Teacher-Student chat are output of the system
- **Feedback :**
- Feedback is output of the system
- **Status of completion :**
- Status of completion is output of the system

# 6.3 DATABASE DESIGN

- teacher

| Field Name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| tid | varchar | 10 | primary key | Id of teacher |
| tname | varchar | 100 | not null | Name of teacher |
| tdob | varchar | | | Date of Birth of teacher |
| tmob | int | 15 | not null | Mobile number of teacher |
| dptname | varchar | 10 | not null | Department name ‘CSE’, ‘CE’, ‘EEE’, ‘ECE’, ‘ME’ |
| timage | varchar | 30 | not null | Image of the teacher |

# 6.3 DATABASE DESIGN(contd.)

- hod

| Field Name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| hid | varchar | 10 | primary key | Id of hod |
| hname | varchar | 100 | not null | Name of hod |
| hdob | varchar | | | Date of Birth of hod |
| hmob | int | 15 | not null | Mobile number of hod |
| dhdpt | varchar | 10 | not null | Department name ‘CSE’, ‘CE’, ‘EEE’, ‘ECE’, ‘ME’ |
| himage | varchar | 30 | not null | Image of the hod |

# 6.3 DATABASE DESIGN(contd.)

- student

| Field name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| adnum | varchar | 10 | primary key | Admission number of student |
| name | varchar | 100 | not null | Student name |
| dob | date | | | Student date of birth |
| mob | integer | 15 | not null | Student mobile number |
| sdptname | varchar | 10 | not null | Department name ‘CSE’, ‘CE’, ‘EEE’, ‘ECE’, ‘ME’ |
| semno | varchar | 10 | not null | Student current semester number |
| simage | varchar | 10 | not null | Image of the student |
| year | varchar | 20 | not null | Academic year of student |

# 6.3 DATABASE DESIGN(contd.)

- verifylogin

| Field name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| uid | varchar | 50 | primary key , foreign key | Id of user |
| uname | varchar | 50 | Not null | Name of user |
| upass | varchar | 50 | Not null | Password of user |
| ukey | time | 5 | Not null | User key , ‘s’ : student , ‘t’ : teacher , ‘h’ : hod ,‘a’ : admin |

- loginstatus

| Field name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| uid | varchar | 10 | primary key , foreign key | Id of the user |
| toflogin | date | | | The time of login |
| toflogout | date | | | The time of logout |



# 6.3 DATABASE DESIGN(contd.)

- subjects

| Field name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| subdpt | varchar | 10 | | Subject Department |
| subcode | varchar | 10 | Primary key | Subject code |
| subname | varchar | 10 | Not null | Subject Name |
| subsemno | int | 10 | Not null | Semester Number |
| subteach | varchar | 100 | | Subject Teacher name |
| subteachid | varchar | 10 | Foreign key | Teacher id |
| subcredit | int | 5 | not null | Credit assigned for subject |

# 6.3 DATABASE DESIGN(contd.)

- materials

| Field name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| subname | varchar | 20 | Primary key | Subject name |
| material | File | | Not null | Material uploaded by teacher |

# 6.3 DATABASE DESIGN(contd.)

- work

| Field name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| workid | varchar | 10 | Primary key | Id of the work |
| workname | varchar | 20 | | Name of work |
| dptname | varchar | 10 | not null | Department name |
| semno | int | 10 | not null | Semester number |
| subname | int | 10 | not null | Name of the subject |
| defwork | varchar | 10 | | Description of the work |
| wfile | file | | | Work file |
| sdate | date | | | Date of assigning work |
| edate | datetime | | | Due date & time to submit |



# 6.3 DATABASE DESIGN(contd.)

- grade

| Field name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| gid | varchar | 5 | primary key | Grade |
| gpoint | int | 10 | not null | Point corresponding to grade |

- Admin

| Field name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| aid | varchar | 10 | primary key | id of admin |
| apass | varchar | 50 | not null | password of admin |



# 6.3 DATABASE DESIGN(contd.)

- attendance

| Field name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| sid | varchar | 10 | primary key | id of student |
| subcode | varchar | 10 | Foreign key,not null | code of subject |
| tname | varchar | 20 | not null | name of teacher |
| attendance | varchar | 10 | | attendance |
| remarks | varchar | 20 | not null | incase of applicable reason for leave |

# 6.3 DATABASE DESIGN(contd.)

- feedback

| Field name | Data type | size | constrain | Description |
|---|---|---|---|---|
| uid | varchar | 15 | | Id of the user |
| feedback | varchar | 1000 | | description of feedback |
| toffb | time | 10 | Not null | time of feedback was given |

- announcement

| Field name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| anmsg | varchar | 100 | Not null | The message contents |
| dptname | varchar | 10 | Not null | Department name. |



# 6.3 DATABASE DESIGN(contd.)

- tutors

| Field name | Data type | size | constrain | Description |
|---|---|---|---|---|
| Year | varchar | 10 | not null | Academic year |
| Dpt | varchar | 10 | not null | Department name |
| Tutor | varchar | 10 | Not null | Name of tutor |
| createby | varchar | 10 | Not null | Name of Hod that assigns tutor |



# 6.3 DATABASE DESIGN(contd.)

- creditreq

| Field name | Data type | size | constrain | Description |
|---|---|---|---|---|
| semno | int | 5 | primary key | Semester number |
| creditallo | int | 5 | not null | Credit allocated for a semester |
| creditmin | int | 5 | not null | Minimum credit required till now for next semester |

- academicyear

| Field Name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| Year | Varchar | 10 | Not null | Academic year |



# 6.3 DATABASE DESIGN(contd.)

- grading

| Field name | Data type | size | constrain | Description |
|---|---|---|---|---|
| uid | varchar | 10 | primary key , foreign key | Id of the student |
| matid | varchar | 20 | Primary key | Id of the material |
| rating | varchar | 20 | not null | Rating or score for the subject |
| tofgrd | datetime | | | Time of grading |
| feedback | varchar | 100 | | Feedback about review. |

# 6.3 DATABASE DESIGN(contd..)

- chatting

| Field Name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| chatid | Autofield | 10 | primary key | Id of the question |
| subname | varchar | 10 | not null | Subject name |
| Stud_id | varchar | 10 | Not null | Student id |
| stud_name | varchar | 10 | not null | Student name |
| teach_id | varchar | 10 | not null | Teacher id |
| status | varchar | 10 | Not null | Status of the replay by teacher |
| quest | varchar | 50 | Not null | Question by student |
| reply | varchar | 50 | | Answer by teacher |



# 6.3 DATABASE DESIGN(contd..)

- marks

| Field Name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| uid | varchar | 10 | primary key | Id if the student |
| workid | float | 15 | | Id of work |
| wname | float | 15 | | Name of work |
| Mark | float | 15 | | Mark |
| Total | float | 15 | | Total |



# 6.3 DATABASE DESIGN(contd..)

- sworks

| Field Name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| workid | varchar | 3 | Not null | Id of work |
| wname | varchar | 50 | Not null | Name of work |
| wsname | varchar | 20 | Not null | Name of subject |
| sid | varchar | 20 | Not null | Student id |
| sname | varchar | 20 | Not null | Student name |
| udate | datetime | | Not null | Date and Time of upload |
| Sfile | File | | | Work file uploaded by student |

# 6.3 DATABASE DESIGN(contd..)

- substatus

| Field Name | Data type | Size | Constrain | Description |
|---|---|---|---|---|
| Userid | Varchar | 10 | Not null | Id of teacher |
| Subname | Varchar | 50 | Not null | Subject name |
| Status | Varchar | 50 | Not null | Status of subject |
| stime | datetime | | Not null | Date and time of status upload |

# 7. IMPLEMENTATION

# 7. IMPLEMENTATION

- Minimum Hardware Requirement
- Minimum Software Requirement
- Sample Code- GUI
- Sample Code- Algorithm

# 7.1
# MINIMUM HARDWARE REQUIREMENTS

- Stable internet connection of at least 3 mb/s is needed.
- Device with minimum  of 375x812 resolution needed to be used.
- RAM of minimum 2 GB is needed.
- Storage of at least 500 mb is required.

# 7.2
# MINIMUM SOFTWARE REQUIREMENTS

- Web browsers like chrome and Mozilla are preferred.
- For mac, safari/chrome is preferred.
- Best viewed with Chrome 99 and above, Mozilla 91.0 and above , Safari 12.1.2 and above.

# 7.3 SAMPLE CODE- GUI

```
@charset "UTF-8";
/*
 * Container style
 */
.ps {
  overflow: hidden !important;
  overflow-anchor: none;
  -ms-overflow-style: none;
  touch-action: auto;
  -ms-touch-action: auto;
}

/*
 * Scrollbar rail styles
 */
.ps__rail-x {
  display: none;
  opacity: 0;
  transition: background-color 0.2s linear, opacity 0.2s linear;
  -webkit-transition: background-color 0.2s linear, opacity 0.2s linear;
  height: 15px;
  /* there must be 'bottom' or 'top' for ps__rail-x */
  bottom: 0px;
  /* please don't change 'position' */
  position: absolute;
}

.ps__rail-y {
  display: none;
  opacity: 0;
  transition: background-color 0.2s linear, opacity 0.2s linear;
  -webkit-transition: background-color 0.2s linear, opacity 0.2s linear;
  width: 15px;
  /* there must be 'right' or 'left' for ps__rail-y */
  right: 0;
  /* please don't change 'position' */
  position: absolute;
}
```



# 7.3 SAMPLE CODE- GUI

```
.ps__rail-y {
  display: none;
  opacity: 0;
  transition: background-color 0.2s linear, opacity 0.2s linear;
  -webkit-transition: background-color 0.2s linear, opacity 0.2s linear;
  width: 15px;
  /* there must be 'right' or 'left' for ps__rail-y */
  right: 0;
  /* please don't change 'position' */
  position: absolute;
}

.ps--active-x > .ps__rail-x,
.ps--active-y > .ps__rail-y {
  display: block;
  background-color: transparent;
}

.ps:hover > .ps__rail-x,
.ps:hover > .ps__rail-y,
.ps--focus > .ps__rail-x,
.ps--focus > .ps__rail-y,
.ps--scrolling-x > .ps__rail-x,
.ps--scrolling-y > .ps__rail-y {
  opacity: 0.6;
}

.ps .ps__rail-x:hover,
.ps .ps__rail-y:hover,
.ps .ps__rail-x:focus,
.ps .ps__rail-y:focus,
.ps .ps__rail-x.ps--clicking,
.ps .ps__rail-y.ps--clicking {
  background-color: #eee;
  opacity: 0.9;
}
```



# 7.4 SAMPLE CODE- ALGORITHM

```
from asyncio.constants import _SendfileMode
from copyreg import add_extension
from datetime import timedelta
from email.policy import default
from plistlib import UID
from threading import get_ident
from unittest.util import _MAX_LENGTH
from django.db import models
from django.contrib.auth.models import AbstractUser
from django.forms import CharField
# Create your models here.

class teacher(models.Model):
    tid = models.CharField(max_length=10, default=0)
    tname = models.CharField(max_length=100, null=False)
    tdob = models.DateField()
    tmob = models.IntegerField(default=0)
    dptname = models.CharField(max_length=100, null=False)
    timage = models.FileField(default=0)
```



# 7.4 SAMPLE CODE- ALGORITHM

```
class hod(models.Model):
    hid = models.CharField(max_length=10, default=0)
    hname = models.CharField(max_length=100, null=False)
    hdob = models.DateField()
    hmob = models.IntegerField(default=0)
    hdpt = models.CharField(max_length=100, null=False)
    himage = models.FileField(default=0)

class student(models.Model):
    adnum = models.CharField(max_length=10, default=0)
    name = models.CharField(max_length=100, null=False)
    dob = models.DateField()
    mob = models.IntegerField(default=0)
    sdptname = models.CharField(max_length=50, default=0, null=False)
    semno = models.CharField(max_length=10, default=0, null=False)
    year = models.CharField(max_length=20, default=0)
    simage = models.FileField(default=0)
```



# 7.4 SAMPLE CODE- ALGORITHM

```
from django.conf import settings
from django.urls import path, include
from django.conf.urls.static import static
from django.conf import settings
from . import views

urlpatterns = [
    path('addstud/', views.addstud, name='addstud'),
    path('add_stud/', views.add_stud, name='add_stud'),
    path('add_teach/',views.add_teach, name='add_teach'),
    path('add_hod',views.add_hod, name='add_hod'),
    path('addwork/',views.addwork, name='addwork'),
    path('add_work/',views.add_work, name='add_work'),
    path('teachhome/', views.teachhome, name='teachhome'),
    path('hodhome', views.hodhome, name='hodhome'),
    path('adminhome', views.adminhome, name='adminhome'),
    path('', views.login_view,name='login_view'),
```



# 7.4 SAMPLE CODE- ALGORITHM

```
{% extends 'tbase.html' %}
{% load static %}
{% block rightmenu %}
<form action="/view_works/" method="post">
    {% csrf_token %}
    <p>Select Subject  :
        <select id="subname" name="subname" value="{{ subname }}">
            {% for results1 in showdpt1 %}
            <option value="{{ results1.subname }}">{{ results1.subname }}</option>
            {% endfor %}
        </select><br>
    </p>
    <button type="submit" value="Submit">Submit</button>

    <table>
        <tr>
            <th>Work Name</th>
            <th>Student Name</th>
            <th>Student id</th>
            <th>Student Work</th>
            <th>Mark</th>
        </tr>
        {% for work in works %}
        <tr>
```



# 8. TEST CASES

# SAMPLE TEST CASES

| Test Type | Input | Expected Output | Actual Output | Success/ Fail |
|---|---|---|---|---|
| add_student | student details | save to database | save to database | success |
| add_teacher | teacher details | save to database | save to database | success |
| add_work | work details | save to database | save to database | success |
| upload materials | documents | files uploaded | files uploaded | success |
| login | user id & password | login to the home page | login to the home page | success |

# 9. CONCLUSION

# CONCLUSION

- In this work, we presented a single platform where all the learning activities can be done easily.
- Our work captures the need of students and teachers properly, and provide facilities accordingly.
- The same login page using different keys ensure ease of use , proper security and authority in this platform .
- Our solution provides an effective method to assess the need of education with in the colleges.

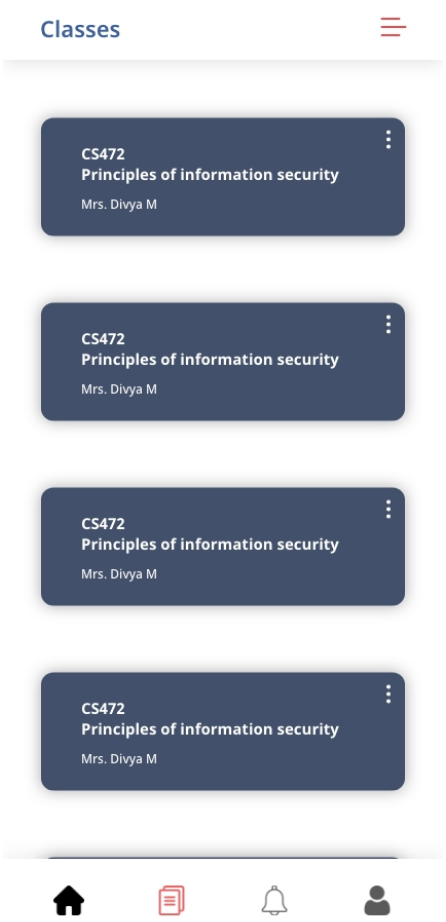
# FUTURE ENHANCEMENT

- To this system online courses apart from academic subjects can be integrated
- Online payment facilities within this system
- Live caption for videos

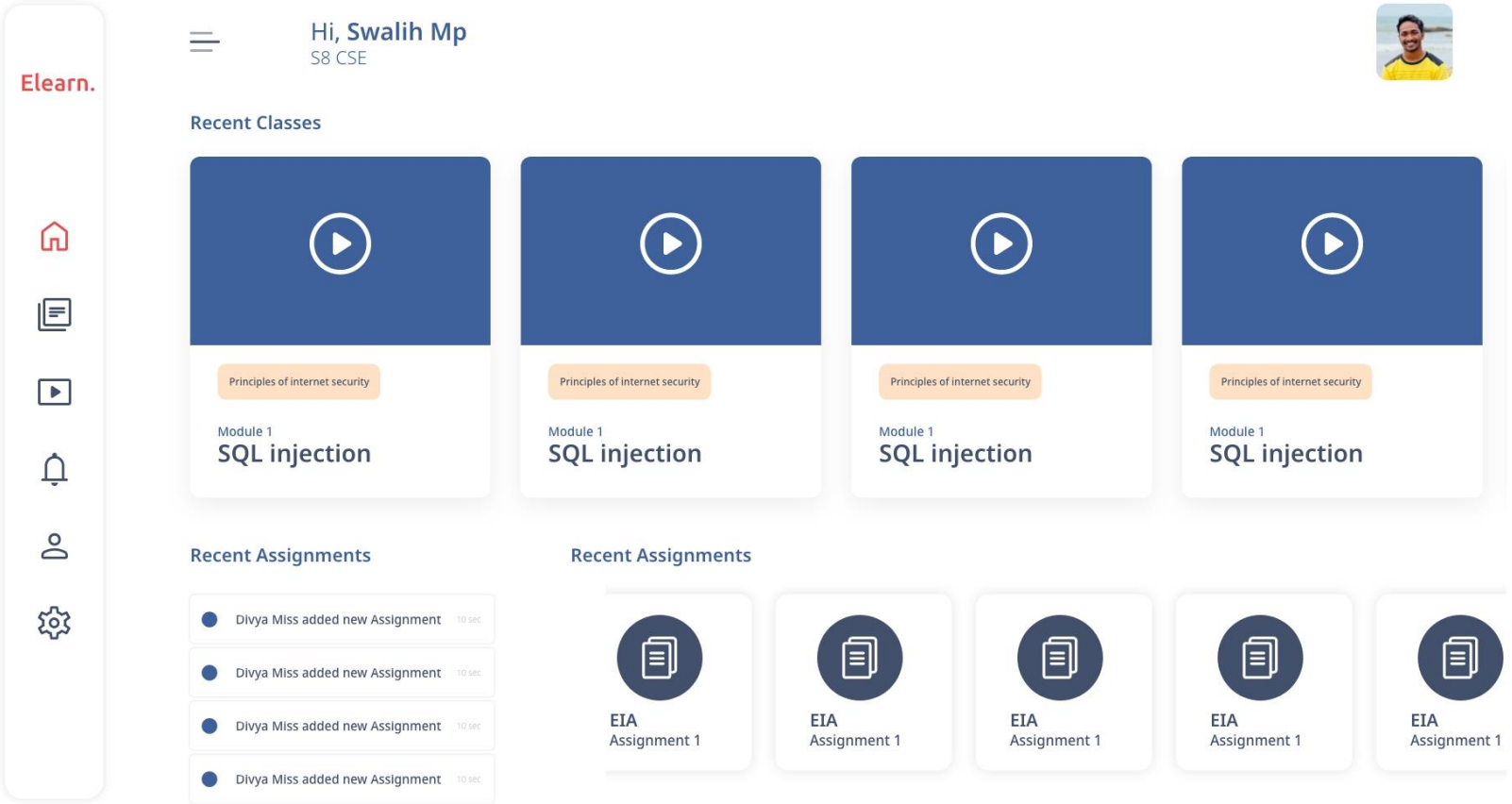
# SCREENSHOTS

# Notifications

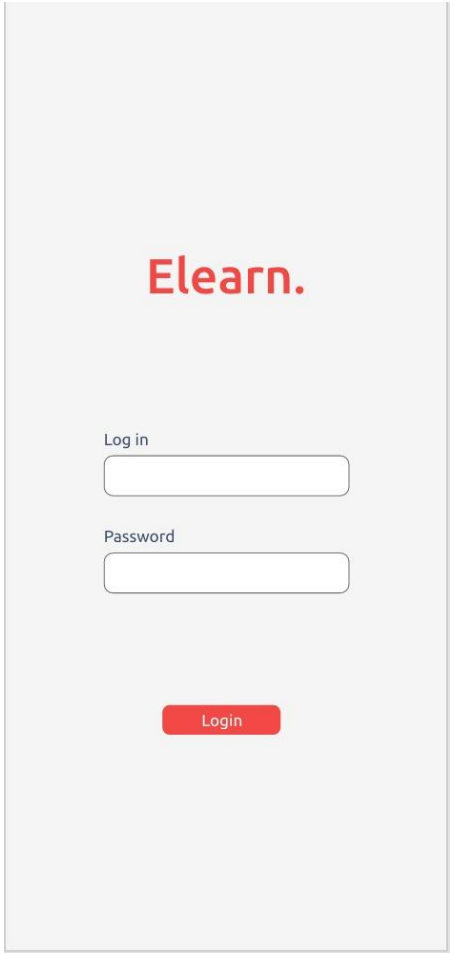
# Courses Of Student



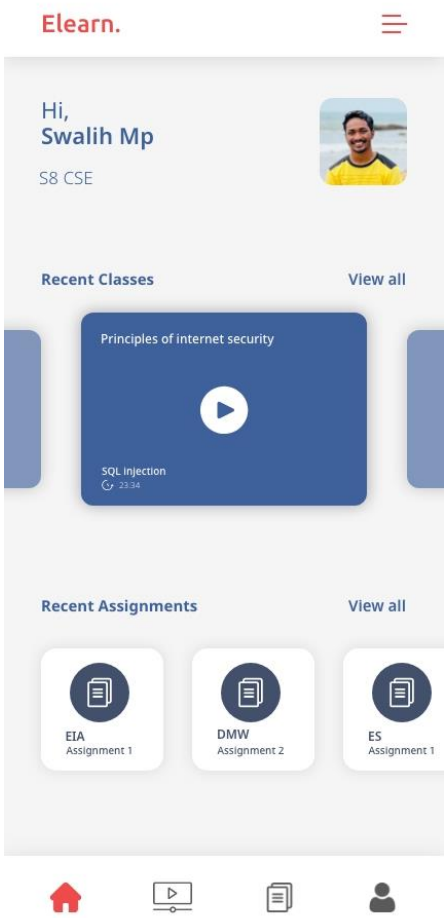
# Subject Page



# Login Page for All Users



# Home Page view of Student



# REFERENCES

| | |
|---|---|
| [1] | https://docs.djangoproject.com/en/3.2/ |
| [2] | https://dev.mysql.com/doc/ |
| [3] | https://dev.w3.org/html5/html-author/ |
| [4] | https://developer.mozilla.org/en-US/docs/Web/JavaScript |
| [5] | https://educationaltechnologyjournal.springeropen.com/articles/10.1186/s41239-020-00216-z |
| [6] | https://developer.mozilla.org/en-US/docs/Web/JavaScript |
| [7] | https://www.apachefriends.org/docs/ |




# THANK YOU