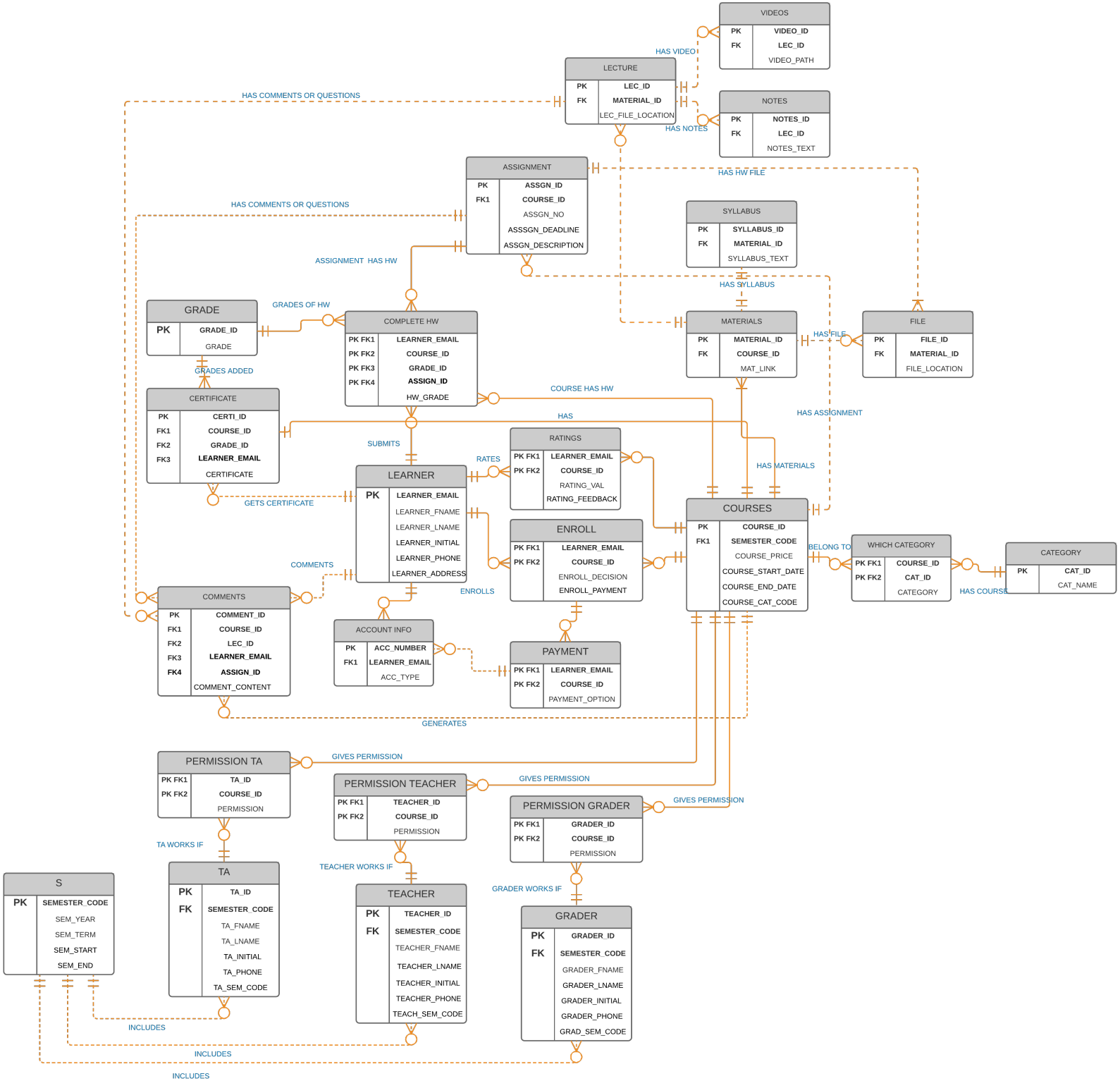
VIDEOS
PK VIDEO_ID
FK LEC_ID
VIDEO_PATH
HAS VIDEO
LECTURE
PK LEC_ID
FK MATERIAL_ID
LEC_FILE_LOCATION
HAS COMMENTS OR QUESTIONS
NOTES
PK NOTES_ID
FK LEC_ID
NOTES_TEXT
HAS NOTES
ASSIGNMENT
PK ASSGN_ID
FK1 COURSE_ID
ASSGN_NO
ASSSGN_DEADLINE
ASSGN_DESCRIPTION
HAS HW FILE
HAS COMMENTS OR QUESTIONS
SYLLABUS
PK SYLLABUS_ID
FK MATERIAL_ID
SYLLABUS_TEXT
ASSIGNMENT HAS HW
HAS SYLLABUS
GRADE
PK GRADE_ID
GRADE
GRADES OF HW
COMPLETE HW
PK FK1 LEARNER_EMAIL
PK FK2 COURSE_ID
PK FK3 GRADE_ID
PK FK4 ASSIGN_ID
HW_GRADE
MATERIALS
PK MATERIAL_ID
FK COURSE_ID
MAT_LINK
HAS FILE
FILE
PK FILE_ID
FK MATERIAL_ID
FILE_LOCATION
GRADES ADDED
CERTIFICATE
PK CERTI_ID
FK1 COURSE_ID
FK2 GRADE_ID
FK3 LEARNER_EMAIL
CERTIFICATE
COURSE HAS HW
HAS
HAS ASSIGNMENT
SUBMITS
RATINGS
PK FK1 LEARNER_EMAIL
PK FK2 COURSE_ID
RATING_VAL
RATING_FEEDBACK
RATES
HAS MATERIALS
LEARNER
PK LEARNER_EMAIL
LEARNER_FNAME
LEARNER_LNAME
LEARNER_INITIAL
LEARNER_PHONE
LEARNER_ADDRESS
GETS CERTIFICATE
ENROLL
PK FK1 LEARNER_EMAIL
PK FK2 COURSE_ID
ENROLL_DECISION
ENROLL_PAYMENT
COURSES
PK COURSE_ID
FK1 SEMESTER_CODE
COURSE_PRICE
COURSE_START_DATE
COURSE_END_DATE
COURSE_CAT_CODE
BELONG TO
WHICH CATEGORY
PK FK1 COURSE_ID
PK FK2 CAT_ID
CATEGORY
HAS COURSE
CATEGORY
PK CAT_ID
CAT_NAME
COMMENTS
COMMENTS
PK COMMENT_ID
FK1 COURSE_ID
FK2 LEC_ID
FK3 LEARNER_EMAIL
FK4 ASSIGN_ID
COMMENT_CONTENT
ENROLLS
ACCOUNT INFO
PK ACC_NUMBER
FK1 LEARNER_EMAIL
ACC_TYPE
PAYMENT
PK FK1 LEARNER_EMAIL
PK FK2 COURSE_ID
PAYMENT_OPTION
GENERATES
PERMISSION TA
PK FK1 TA_ID
PK FK2 COURSE_ID
PERMISSION
GIVES PERMISSION
PERMISSION TEACHER
PK FK1 TEACHER_ID
PK FK2 COURSE_ID
PERMISSION
GIVES PERMISSION
PERMISSION GRADER
PK FK1 GRADER_ID
PK FK2 COURSE_ID
PERMISSION
GIVES PERMISSION
TA WORKS IF
TEACHER WORKS IF
GRADER WORKS IF
S
PK SEMESTER_CODE
SEM_YEAR
SEM_TERM
SEM_START
SEM_END
TA
PK TA_ID
FK SEMESTER_CODE
TA_FNAME
TA_LNAME
TA_INITIAL
TA_PHONE
TA_SEM_CODE
TEACHER
PK TEACHER_ID
FK SEMESTER_CODE
TEACHER_FNAME
TEACHER_LNAME
TEACHER_INITIAL
TEACHER_PHONE
TEACH_SEM_CODE
GRADER
PK GRADER_ID
FK SEMESTER_CODE
GRADER_FNAME
GRADER_LNAME
GRADER_INITIAL
GRADER_PHONE
GRAD_SEM_CODE
INCLUDES
INCLUDES
INCLUDES

REPORT

METODS AND ASSUMPTIONS USED

Before going into the specifics I want to convey the fact that this report will take into consideration all the entities used as well as the assumptions used for them. The entities will be numbered from 1 to whatever I have used in the Homework. Also it is to be noted that every design is going to be different. This is because every person is different and he thinks differently and therby creates something that is totally according to his or her comprehension .

1 ***LEARNERS AND COURSES AND ENROLL:*** Learners include students or any other people willing to study.

- ***Relationship:*** Here one learner can take many courses and 1 course can be taken by many Learners. As a result of this the Learner to courses is a Many to many(M:N) relationship.

  However it is not wise to keep 2 primary keys in 2 different entities. As a result of this we will spilt this (M:N) into two 1:M relationships. For this we need to select a new dependent entity. We choose Enroll as that entity.
  Also the relationships are optional as learner may or may not take a course and course may or may not have a particular learner.

  Enroll will have primary keys of both LEARNER and Courses as its foreign key and that will be its composite primary key.

- Attributes:

  Learner(LEARNER_EMAIL,Learner_FNAME,LEARNER_LNAME,LEARNER_INITIAL,LEARNER_ADDRESS)

Courses(<u>COURSE_ID,</u>SEMESTER_CODE,COURSE_PRICE, COURSE_START_DATE, COURSE_END_DATE, COURSE_CAT_CODE)

ENROLL(<u>LEARNER_EMAIL,COURSE_ID,</u>PAYMENT_OPTION)

2 ***<u>COURSES AND ITS CATEGORIES:</u>*** As per the business rules, courses are grouped in multiple categories and a course may be present in a different category.

- ***<u>Relationship:</u>*** This means that a category can have multiple course and a course can have a multiple categories. Thus course and category have (M:N) relationship between them

  We choose WHICH CATEGORY as that entity which will become the bridge between COURSES and its CATEGORY.

- Attributes:

  Learner(<u>LEARNER_EMAIL,</u>Learner_FNAME,LEARNER_LNAME,LEARNER_INITIAL,LEARNER_ADDRESS)

  CATEGORY(<u>CAT_ID,</u>SEMESTER_CODE,CAT_NAME)

  WHICH CATEGORY(<u>COURSE_ID,CAT_ID,</u>PAYMENT_OPTION)

3 ***<u>COURSES AND MATERIALS:</u>*** If we think of it a course can have a lot of content, but all of that content is considered to be under materials. As a result of this we can say that course a course can have only 1 material and that particular material can include everything else.

- ***<u>Relationship:</u>*** This means that COURSE and MATERIALS have 1:1 relationship.

- Attributes:

MATERIALS(MATERIAL_ID,COURSE_ID,MAT_LINK)

The attribute material link has the path to where the material lies on the computer.

4 ***SYLLABUS AND MATERIALS:*** It is not possible that a course has more than 1 syllabus. It just does not make any sense.

- ***Relationship:*** This means that SYLLABUS and MATERIALS have 1:1 relationship.

- Attributes:

  MATERIALS(SYLLABUS_ID,COURSE_ID,SYLLABUS_TEXT)

  The attribute SYLLABUS_TEXT has the actual syllabus in text.

5 ***FILE AND MATERIALS:*** As mentioned earlier a course can have just 1 material, however there will be many files and textbooks below under materials. A file contains all the necessary things required for a course. Obviously a material of a course may have many files in it.

- ***Relationship:*** This means that FILES and MATERIALS have 1:M relationship.

- Attributes:

  FILE(FILE_ID,MATERIAL_ID,FILE_LOCATION)

  The attribute file_location gives us the location of the file in computer.

6 ***LECTURES AND MATERIALS AND COURSE:*** Business rule states that a materials could be general or come from a lecture.
This means that 1 lecture constitutes to many materials.
Also 1 course has many lectures and it is not optional.

- ***Relationship:***
  LECTURE AND MATERIALS :This means that LECTURES and MATERIALS have 1:M relationship.

LECTURE AND COURSES :This means that LECTURES and COURSES have 1:M relationship.

- Attributes:

  LECTURE(<u>LEC_ID</u>,MATERIAL_ID,LEC_FILE_LOCATION)

  The attribute LEC_FILE_LOCATION gives us the location where the lecture video is stored.

7 ***<u>LECTURES AND VIDEOS AND NOTES:</u>*** Business rule states that each lecture may have videos and notes. It is possible that a lecture may have a normal video and some other documentary associated with it, this is also possible in the case of notes.

- ***<u>Relationship:</u>***
  LECTURE AND VIDOES :This means that LECTURES and VIDEOS have 1:M relationship.
  LECTURE AND NOTES :This means that LECTURES and NOTES have 1:M relationship.

- Attributes:

  VIDEOS(<u>VIDEOS_ID</u>,LEC_ID,VIDEO_PATH)
  The attribute VIDEO_PATH gives us the location where the lecture video is stored.

  NOTES(<u>NOTES_ID</u>,LEC_ID,NOTES_TEXT)
  The attribute NOTES_TEXT gives us the actual text of the notes.

8 ***<u>COURSES AND ASSIGNMENTS:</u>*** Each course can have many assignments throughout the semester. Also many assignments have 1 course only.

- ***<u>Relationship:</u>*** This means that COURSE and Assignments have 1:1 relationship.

- Attributes:

  ASSIGNMENT(<u>ASSGN_ID</u>,COURSE_ID,ASSGN_NO,ASSGN_DEADLINE< ASSGN_DESCRIPTION)

  The attribute material link has the path to where the material lies on the computer.

9 ***<u>(LEARNERS,LECTURE,ASSIGNMENT,COURSES) AND COMMENTS:</u>*** For posting comments an entity should have:

1:Comment number
2: Learner_id
3: Course id:
4: lec and assignment

One Learner can post multiple comments regarding multiple lectures.
One lecture can have many comments posted by the students seeing those lectures.
One assignment can have many questions posted about it.
One course itself can have many comments posted in its different lectures and assignments.

- ***<u>Relationship:</u>***

  LECTURE and COMMENTS: As a result of this the Lecture to comments is a one to many(1:M) relationship.

  LEARNER and COMMENTS: As a result of this the Learner to comments is a one to many(1:M) relationship.

  ASSIGNMENTS and COMMENTS: As a result of this the assignments to comments is a one to many(1:N) relationship.

  COURSE_ID and COMMENTS: As a result of this the Learner to comments is a one to many(1:N) relationship.

- Attributes:

COMMENTS(COMMENT_ID,LEARNER_EMAIL,COURSE_ID,LEC_ID,ASSIGN_ID))

Courses(COURSE_ID,SEMESTER_CODE,COURSE_PRICE, COURSE_START_DATE, COURSE_END_DATE, COURSE_CAT_CODE)

ENROLL(LEARNER_EMAIL,COURSE_ID,PAYMENT_OPTION)

10 ***LEARNER AND ACCOUNT INFO:*** If we consider practical approaches a LEARNER can have many bank accounts in many banks and he would want to pay for the course from any of the bank accounts. Thus one learner can have many bank accounts while 1 bank account can have only one owner.

- ***Relationship:*** This means that LEARNER and ACCOUNT INFO have 1:M relationship.

- Attributes:

  ACCOUNTINFO(ACC_NUMBER,LEARNER_EMAIL,ACC_TYPE)

  The attribute ACC_TYPE tells us what kind of account are we using.

- ***ENROLL,COURSE_ID,ACC_INFO) AND PAYMENT:*** Account info has many accounts from 1 student however payment should choose account specified by LEARNER. Also payments has different methods like, visa , mastercard etc.
  Relationship:
  ENROLL and PAYMENT: This means that ENROLL and PAYMENT have 1:M relationship.
  COURSE_ID and PAYMENT: This means that COURSE_ID and PAYMENT have 1:M relationship.
  ACCINFO and PAYMENT: Payment is related to ACCINFO with M:1.

- Attributes:

PAYMENY(COURSE_ID,LEARNER_EMAIL,PAYMENT_OPTION)

The attribute PAYMENT_OPTION chooses the right kind of option specified by enroll.

11 ***(LEARNERS,GRADES,ASSIGNMENT,COURSES) AND GRADE:*** One assignment can have many grades and one grade can have many assignment. One learner can have many grades and vice versa, same goes for the other 2 entities as well.
Thus we make a new entity COMPLETE ENTITY to handle this M:N relationship.

- ***Relationship:***
  COURSES and COMPLETE HW: As a result of this the COURSES to comments is a one to many(1:M) relationship.

  LEARNER and COMPLETE HW: As a result of this the Learner to comments is a one to many(1:M) relationship.

  ASSIGNMENTS and COMPLETE HW: As a result of this the assignments to comments is a one to many(1:M) relationship.

  GRADES and COMPLETE HW: As a result of this the GRADES to comments is a one to many(1:M) relationship.

- Attributes:

  GRADE(GRADE_ID,GRADE**)**

12 ***LEARNER AND (CERTIFICATE ,RATINGS):*** Both CERTIFICATE and RATING are entity created to handle the M:N relation between (LEARNER and GRADE) and (LEARNER AND COURSE_ID0 respectively.

- ***Relationship:*** This means that LEARNER and CERTIFICATE and LEARNER and RATING have 1:M relationship.

13 ***(TA,TEACHERS,GRADERS) and COURSE_ID:*** Now one TA can be a TA of many courses and similarly a single course can have many TA , same is true for for graders and Teachers too. Thus all of them have M:N relationship with course_id.

14 ***SEMESTER and (TA,TEACHERS,GRADERS):*** a Semester can have many TAs, teachers and graders and as a result of this semester has 1:M relationship to all of them.