Assignment 2: Individual Design

Xin Tao (GTID: xtao41), Date: 05/29/2017

CLASSES, ATTRIBUTES and OPERATIONS:

Class #1: Course <Online <u>courses</u> can be offered in a variety of formats…>

- Course ID attr: <Each course must have a distinct <u>Course ID</u> >

- Title attr: <Each course must have…, along with the Course <u>Title</u> >

- Description attr: <Each course must have …, along with … <u>Description</u>>

- Time Offered attr: <… courses might only be offered during certain <u>times</u>>

- Prerequisite attr: <…have one or more other <u>prerequisite</u> courses, must pass the

  prerequisite before can take the course>

Class #2: Course Catalog< …select courses from the most current Course Catalog>

- Update() op: <…be able to <u>update</u> our courses>

- Create() op: <…<u>creation</u> of new courses, …>

- Remove() op: <… occasional <u>removal</u> of existing courses>

Class #3: Member

- UUID attr: < …will also have a single <u>UUID</u> to distinguish…>

- Names attr: <… <u>names</u>, addresses and phone numbers at a minimum>

- Address attr: <… names, <u>addresses</u> and phone numbers at a minimum>

- Phone Number attr: <… names, addresses and <u>phone numbers</u> at a minimum>

Class #4: Instructor <These courses must be …managed by qualified <u>instructors</u>.>

- Skills attr: <…where each instructor has the <u>skills and competencies</u> to…>

Class #5: Hired Instructor < ⋯a list of <u>instructors</u> who have been <u>hired</u>>

- Term attr: <For each <u>term</u>, a list of instructors who have been hired>

- Year attr:

- Create course() op: < These courses must be <u>created</u>, taught and managed by …>

- Teach course() op: < These courses must be created, <u>taught</u> and managed by …>

- Mange course() op: < These courses must be created, taught and <u>managed</u> by …>

- Select course() op: <Each instructor will <u>select</u> one of their eligible courses to teach>

- Enter records() op: <…be able to <u>enter</u> new academic <u>records</u> and update previously created records>

- Update records() op: <…be able to enter new academic records and <u>update</u> previously created records>

Class #6: Student < Each <u>student</u> will have a "UUID" >

- Request enrollment() op: < …allow students to <u>request enrollment</u> in certain courses>

- Request disenrollment() op: < … and to <u>request disenrollment</u> as needed >

- Take courses() op: < …<u>take</u> a wide variety of courses >

- Retake course() op:< ˙˙˙students might want to <u>retake</u> a course >

- Select courses() op: < …<u>select</u> courses from the most current Course Catalog>

Class #7: Working Student < …will have a significant number of working students>

- Job & Position attr: <…most appropriate for their current <u>job/position</u>…>

- Goals attr: <…most appropriate for their … and <u>future goals</u>>

- Interests attr: <…best fit their <u>interests</u> and skill requirements…>

- Work Schedule attr: <… students can take courses as their <u>work schedule</u>>

Class #8 Record < For each course that is taken by a student, an academic <u>record</u> …>

- Student UUID attr: < For each course that is taken by a student…>

- Term attr

- Year attr

- CourseID & Session attr: < … performance during that <u>session of the course</u>>

- Comments attr: < Each record with comments …>

- Link attr: < ⋯should also include a link to the instructor >

Class #9 Historical Record <…system must be able to maintain <u>historical records</u> >

- Student UUID attr: < ⋯maintain the historical records for all of these <u>students</u> >

Class #10 Session <… the student's performance during that session of the course >

- Session ID attr

- Term attr

- Year attr

- Course ID

- Instructor ID: <the UUID of the Instructor>

- Seats Available attr: < ⋯determine the total number of available seats…>

- Seats Assigned attr: <… students who have already been assigned a seat>

- Update Seats() op: <This operation can be used to change seats available and/or

  seats assigned>

RELATIONSHIPS:

Relationships #1 (Association class): Take: association between Course and Student

<Their students will be given the opportunity to <u>take</u> a wide variety of courses>

- Term attr: < … the <u>term</u> and year must also be maintained >

- Year attr: < … the term and <u>year</u> must also be maintained >

- Letter grade attr: < The student's performance … by a final <u>letter grade</u>>

- Session attr: < ···student's performance during that <u>session</u> of the course >

Relationships #2 (Association class): Teach: association between Course and

Instructor < Each instructor will select one of their eligible courses to <u>teach</u> during the

term>

- Term attr:

- Year attr:

Relationships #3 (Association class): Course request: association between Student and

Course <… …keep track of student <u>course requests</u> …>

- Type attr: < request <u>enrollment</u> …and to request <u>disenrollment</u> as needed>

- Approved enrollment attr: <Working student is permitted to take 1, 2 or 3

  courses>

- Grant() op: < …keep track of student course requests that are <u>granted</u> or refused>

- Refuse() op: < …keep track of student course requests …<u>refused</u>>

- Track() op: < …keep <u>track</u> of student course requests>

Relationships #4: Manage: association between Hired Instructor and Record <

<u>Instructors</u> must be able to <u>enter</u> new academic <u>records</u> and <u>update</u>>

Relationships #5 Generalization between Instructor and Member < Instructor is a type of Member>

Relationships #6 Generalization between Student and Member <Student is a type of Member>

Relationships #7 Generalization between Working Student and Student <Working Student is a special case of Student>

Relationships #8 Generalization between Hired Instructor and Instructor < Hired Instructor is a special case of Instructor>

Relationships #9 Aggregation between Student and Historical Record <A student has a historical record>

Relationships #10 Aggregation between Instructor and Course <A class has one or multiple instructors>

Relationships #11 Aggregation between Student and Course <A class has many Students>

Relationships #12 Composition between Record and Historical Record <Historical Record is a collection of Records>

Relationships #13 Composition between Course and Course Catalog < Course Catalog is a collection of Course >

Relationships #14 Composition between Session and Course < A Course has one or more sessions>

Relationships#15 Generate: association between Association Class "Take" and Record <A student taking a course can generate a Record >

Relationships#16 Generate: association between Association Class "Teach" and

Session <An Instructor teaching a course can generate a Session >