Project Step 5: Displaying Data

Open Resource Learning Database

Marc Tibbs (tibbsm@oregonstate.edu)
CS340-Spring 2018

May 27, 2018

Updates

- Project website instance running on **http://flip3.engr.oregonstate.edu:3436/**
- Added an "Updates" section so that changes to the project outline are more apparent.
- Specified that the this will be on online service in the "Project Outline" section based on feedback.
- Changed email limits to 254 characters based on feedback.
- Updated Resource and Class URLs to be required.
- No feedback received from Project Step 2 & 3.
- Updated Schema arrows based on feedback.
- Updated relationship between resources and class to a 1-to-Many relationship in order to meet the assignment criteria.
- Updated relationship between group and class to a 1-to-Many relationship in order to meet the assignment criteria.
- Updated schema and ERD to reflect changes in relationships.

- Updated entity and relationship descriptions to reflect changes in relationships noted above.
- Updated "dataDefine.sql" to better reflect the project outline.

1 Project Outline

I plan on building a database to be used by students who are interested in learning new subjects with other people. The database will contain data from freely accessible educational resources online and keep track of students, groups of students, and the projects that they are working on. There is a rapidly growing amount of educational resources available on the internet to anyone interesting in learning about a new subject, so there is plenty of data to add to the database. This database will provide a service to a large amount of the public who are looking to learn a new skill or subject while meeting and socializing with new people. This resource will be available to online.

2 Database Outline

The entities in the database are:

- **Student** – This entity represents students within the database. Students are able to join classes and project groups. It has the following attributes:
  - **stu_id:** This number is automatically assigned to a student when they are recorded in the database. It is an auto-incrementing number which is the primary key.
  - **stu_fname:** This attribute represents the student's first name and is a string of at most 100 characters. It cannot be blank and there is no default.
  - **stu_lname:** This attribute represents the student's last name and is a string of at most 100 characters. It cannot be blank and there is no default.

  - **stu_email:** This attribute represents the student's email address and is a string of at most 254 characters. It cannot be blank and there is no default.

- **Class** – This entity represents classes that students can participate in. Each class has its own project group in which students can build something using what they learn in a specific class. It has the following attributes:

  - **cla_id:** This number is automatically assigned to a class when it is recorded in the database. It is an auto-incrementing number which is the primary key.
  - **cla_title:** This attribute represents the official title of a class. It is a string of at most 100 characters. It cannot be blank and there is no default.
  - **cla_url:** This attribute represents the specific web address of the class's homepage. It is a string of at most 200 characters. It cannot be blank and there is no default.

- **Group** – Each project group is composed of many students and belongs to one class. It has the following attributes:

  - **gro_id:** This number is automatically assigned to a group when it is recorded in the database. It is an auto-incrementing number which is the primary key.
  - **gro_name:** This attribute represents the name of a group. It is a string of no more than 100 characters. It cannot be blank and defaults to the "Class Name Group".
  - **cla_id:** This number is automatically assigned to a class when it is recorded in the database. It is an auto-incrementing number which is the primary key.

- **Resource** – Each class has additional resources which students can refer to. It has the following attributes:

  - **res_id:** This number is automatically assigned to a resource when it is recorded in the database. It is an auto-incrementing number which is the primary key.

- **res_title:** This attribute represents the title of the resource. It is a string of no more than 100 characters. It cannot be blank and there is no default.
- **res_author:** This attribute represents the author of the resource. It is a string of no more than 100 characters. It can be left blank and there is no default.
- **res_url:** This attribute represents the specifich web address where the resource is available. It is a string of no more than 200 characters. It cannot be left blank and there is no default.
- **cla_id:** This number is automatically assigned to a class when it is recorded in the database. It is an auto-incrementing number which is the primary key.

The relationships in the database are:

- **Students can take classes.** A student can participate in multiple classes and a class can have multiple students so the relationship is a many-to-many relationship. Students may exist in the system and not be taking any classes. This will allow students to sign up for the service first and then decide on the classes to take and groups to join later on. Classes will have to have at least one student signed up to exist.
- **Students can participate in project groups.** A student can participate in multiple project groups and a group can have multiple students. This represents a many-to-many relationship. Students do not have to partake in any groups. On the other hand a group must have at least one student in it.
- **Classes can have additional resources.** Each class may have multiple additional resources for students to learn from. Each resource can only have a relationship with one class. This represents a one-to-many relationship. Classes do not have to have any additional resources. While every resource needs to be related to at least one class.
- **Each class can have many groups.** Each class can have many groups. The groups will progress through the class together and work on a group project together. Each class can have many groups or no group at all. Each group can be related to only one class. This represents a one-to-many relationship.

3 Feedback

- Feedback from Project Step 2
  - **Feedback from the grader.** Added feedback section to the project outline to show feedback.
  - **Schema not according to Symbol Key. Arrows marked incorrectly.** Updated the project schema and its arrows.
- Feedback from Project Step 3
  - **Is the SQL file syntactically correct? This can be easily verified by importing/copy-pasting it in phpmyadmin. It doesnt give me any errors when running the SQL database.** Went back through the SQL files and reimported my files. The updated files have been uploaded to Canvas.
  - **Are the data types appropriate considering the description of the attribute in the database outline? Yes, it was perfect.** No changes.
  - **Are the foreign keys correctly defined when compared to the Schema? Yes, no found errors.** No changes.
  - **Are relationship tables present when compared to the ERD/Schema? They are all present when compared to Schema.** No changes.
  - **The SQL file is giving errors after importing it to PhpMyAdmin.** Updated the SQL files and there are no errors when importing it to PhpMyAdmin.
  - **The data types look different compared to outline All the foreign keys correctly defined The relationship tables are present.** Definitions and data definitions have been updated.

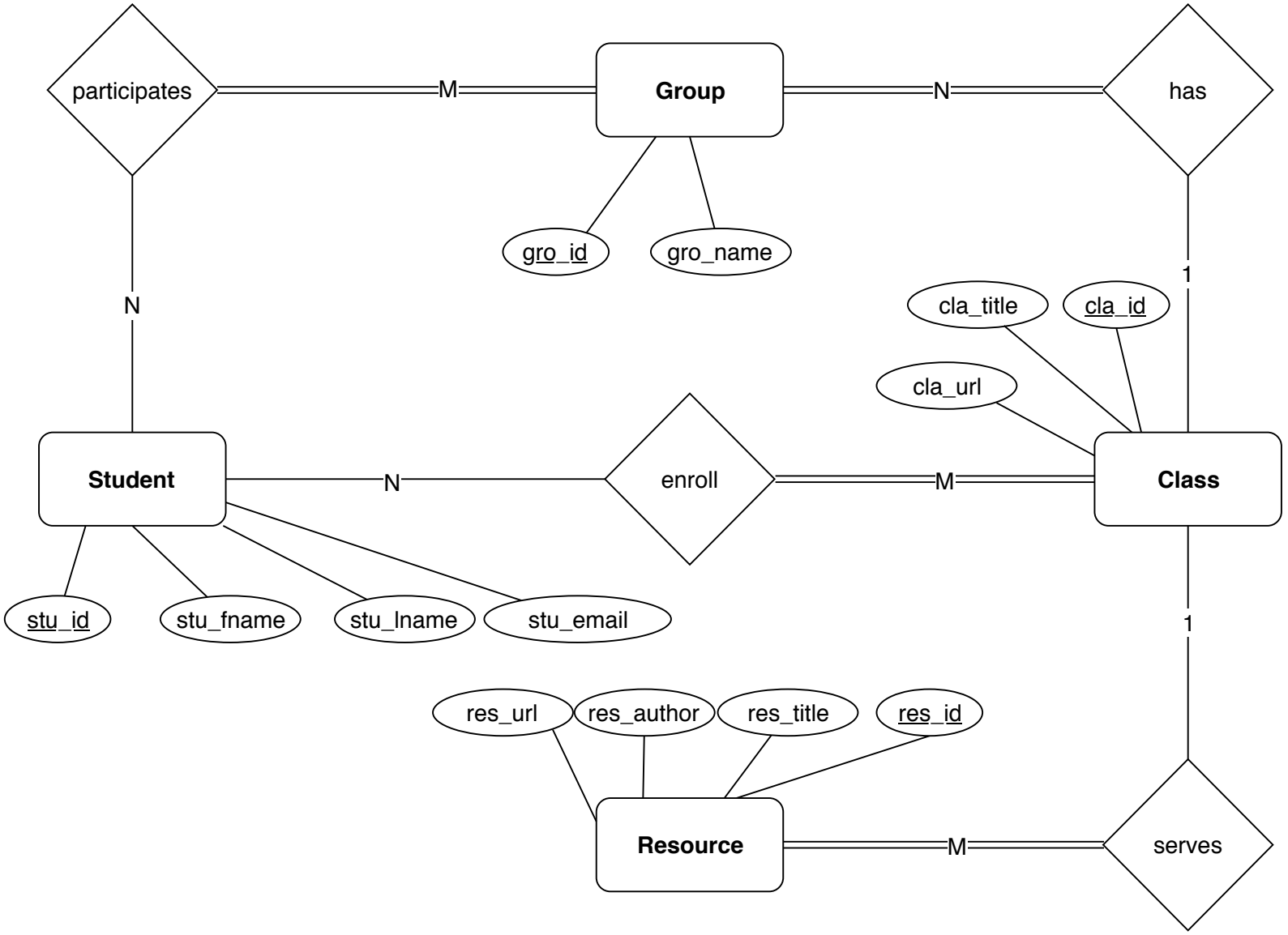
4 Website Features

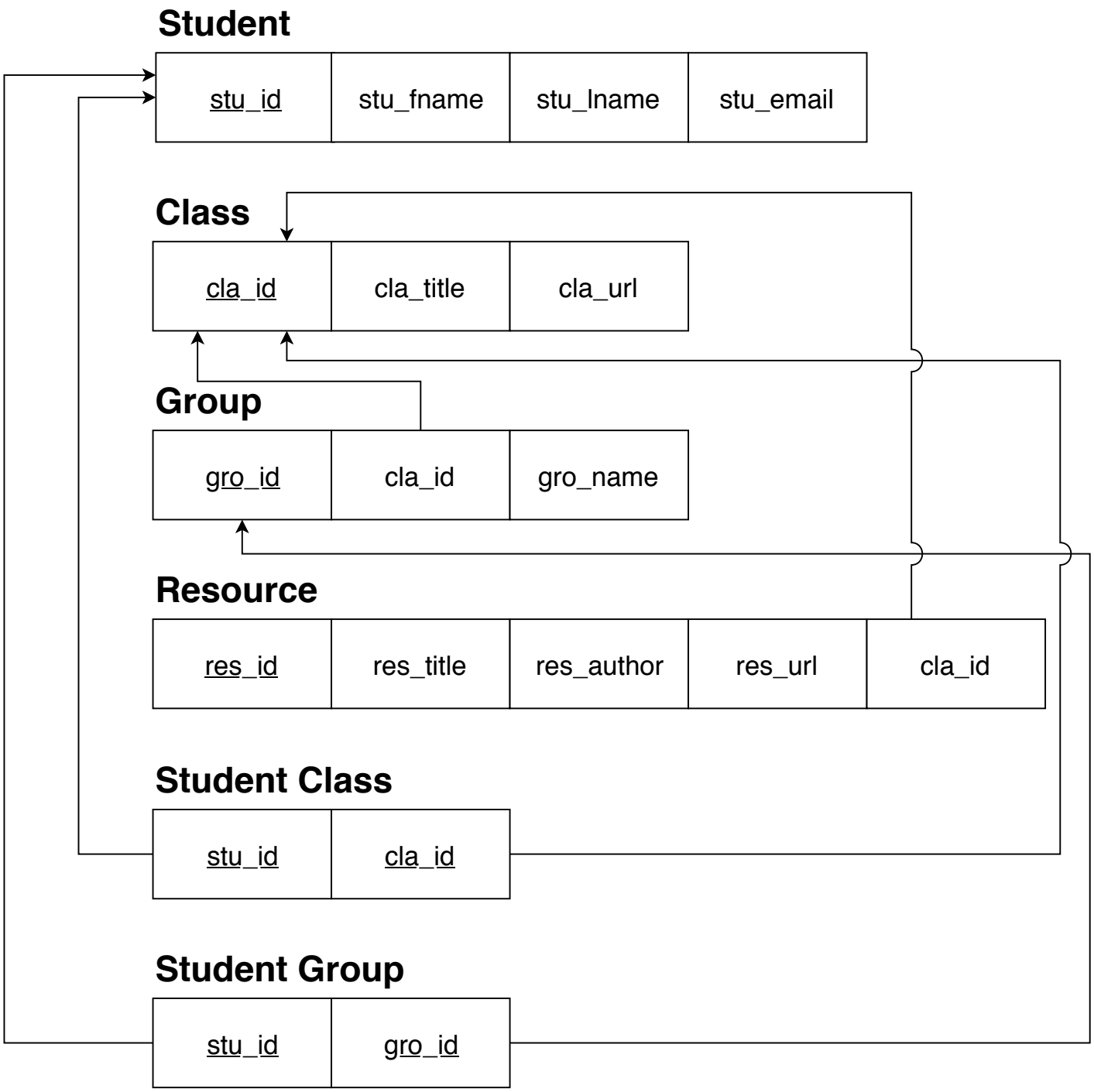
- **Has at least 4 entities - E1,E2,E3 and E4?**
  - Student: http://flip3.engr.oregonstate.edu:3436/student

  - Class: http://flip3.engr.oregonstate.edu:3436/class
  - Resource: http://flip3.engr.oregonstate.edu:3436/resource
  - Group: http://flip3.engr.oregonstate.edu:3436/group
- **Has at least two entities in a many-to-many relationship R1?**
  - Student/Class: http://flip3.engr.oregonstate.edu:3436/student$_{c}lass$
  - Student/Group: http://flip3.engr.oregonstate.edu:3436/student$_{g}roup$
- **Your project has at least two entities in a one-to-many relationship R2?**
  - Class/Group: http://flip3.engr.oregonstate.edu:3436/group
  - Class/Resource: http://flip3.engr.oregonstate.edu:3436/resource
- **Can show records from E1?** Student: http://flip3.engr.oregonstate.edu:3436/student
- **Can show records from E2?** Class: http://flip3.engr.oregonstate.edu:3436/class
- **Can show records from E3?** Resource: http://flip3.engr.oregonstate.edu:3436/resource
- **Can show records from E4?** Group: http://flip3.engr.oregonstate.edu:3436/group
- **Can show records from the many-to-many relationship R2?**
  - Student/Class: http://flip3.engr.oregonstate.edu:3436/student$_{c}lass$
  - Student/Group: http://flip3.engr.oregonstate.edu:3436/student$_{g}roup$
- **Can show records from the one-to-many relationship R1?**
  - Class/Group: http://flip3.engr.oregonstate.edu:3436/group
  - Class/Resource: http://flip3.engr.oregonstate.edu:3436/resource
- **Can add new records to E1?** Student: http://flip3.engr.oregonstate.edu:3436/student
- **Can add new records to E2?** Class: http://flip3.engr.oregonstate.edu:3436/class
- **Can add new records to E3?** Resource: http://flip3.engr.oregonstate.edu:3436/resource
- **Can add new records to E4?** Group: http://flip3.engr.oregonstate.edu:3436/group

- **Can remove records from E1?** Student: http://flip3.engr.oregonstate.edu:3436/student
- **Can remove records from E2?** Class: http://flip3.engr.oregonstate.edu:3436/class
- **Can remove records from E3?** Resource: http://flip3.engr.oregonstate.edu:3436/resource
- **Can remove records from E4?** Group: http://flip3.engr.oregonstate.edu:3436/group
- **Can associate any rows between E1 and E2 in the many-to-many relationship?**
  - Student/Class: http://flip3.engr.oregonstate.edu:3436/student$_c lass$
  - Student/Group: http://flip3.engr.oregonstate.edu:3436/student$_g roup$
- **Can de-associate any rows between E1 and E2 in the many-to-many relationship?**
  - Student/Class: http://flip3.engr.oregonstate.edu:3436/student$_c lass$
  - Student/Group: http://flip3.engr.oregonstate.edu:3436/student$_g roup$
- **Can change association of rows between E2 and E2 in the many-to-many relationship?**
  - Student/Class: http://flip3.engr.oregonstate.edu:3436/student$_c lass$
  - Student/Group: http://flip3.engr.oregonstate.edu:3436/student$_g roup$
- **Can associate any rows between E1 and E3 in the one-to-many relationship?**
  - Class/Group: http://flip3.engr.oregonstate.edu:3436/group
  - Class/Resource: http://flip3.engr.oregonstate.edu:3436/resource
- **Can de-associate any rows between E1 and E3 in the one-to-many relationship?**
  - Class/Group: http://flip3.engr.oregonstate.edu:3436/group
  - Class/Resource: http://flip3.engr.oregonstate.edu:3436/resource
- **Can search on at least one Entity's one attribute ?**Student: http://flip3.engr.oregonstate.edu:3436/student

- **Can update details for E1?**Student: http://flip3.engr.oregonstate.edu:3436/student
- **Can update details for E2?**Class: http://flip3.engr.oregonstate.edu:3436/class
- **Can update details for E3?**Resource: http://flip3.engr.oregonstate.edu:3436/resource
- **Can update details for E4?**Group: http://flip3.engr.oregonstate.edu:3436/group


participates
M
Group
N
has
gro_id
gro_name
N
1
cla_title
cla_id
cla_url
Student
N
enroll
M
Class
stu_id
stu_fname
stu_lname
stu_email
1
res_url
res_author
res_title
res_id
Resource
M
serves


Student
stu_id
stu_fname
stu_lname
stu_email
Class
cla_id
cla_title
cla_url
Group
gro_id
cla_id
gro_name
Resource
res_id
res_title
res_author
res_url
cla_id
Student Class
stu_id
cla_id
Student Group
stu_id
gro_id