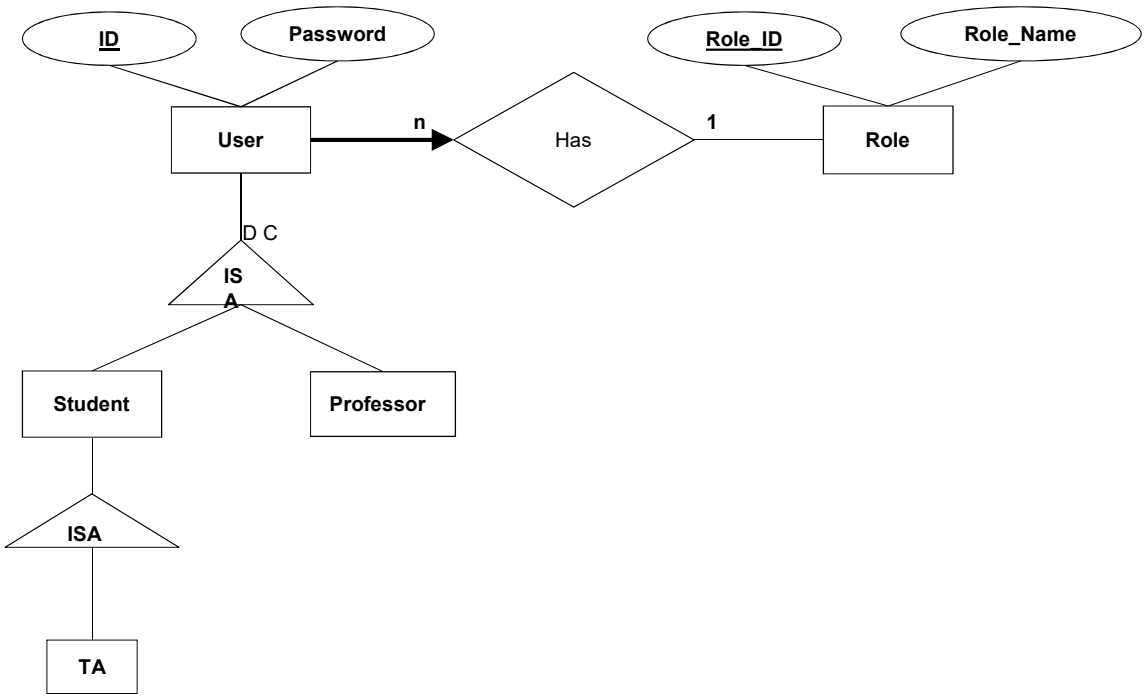
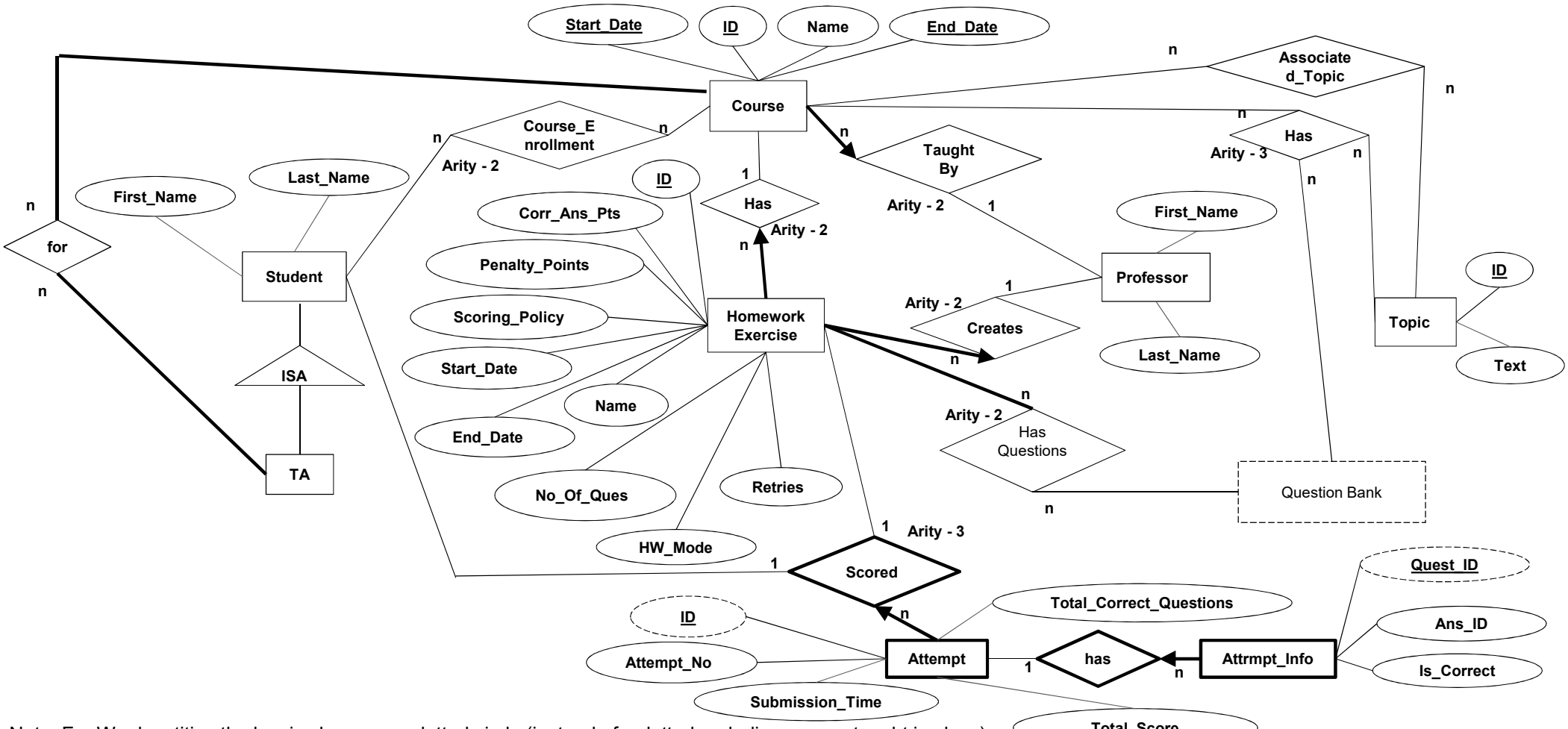
TEAM MEMBERS

Name	Unity Id
Abhisha Bhattacharyya	abhatt22
Bhavik Patel	bcpatel
Shrikanth Narayanaswamy Chandrasekaran	snaraya7
Arjun Sivagaminathan Mathan Kumar	asivaga
Rajan Alwan	ralwan

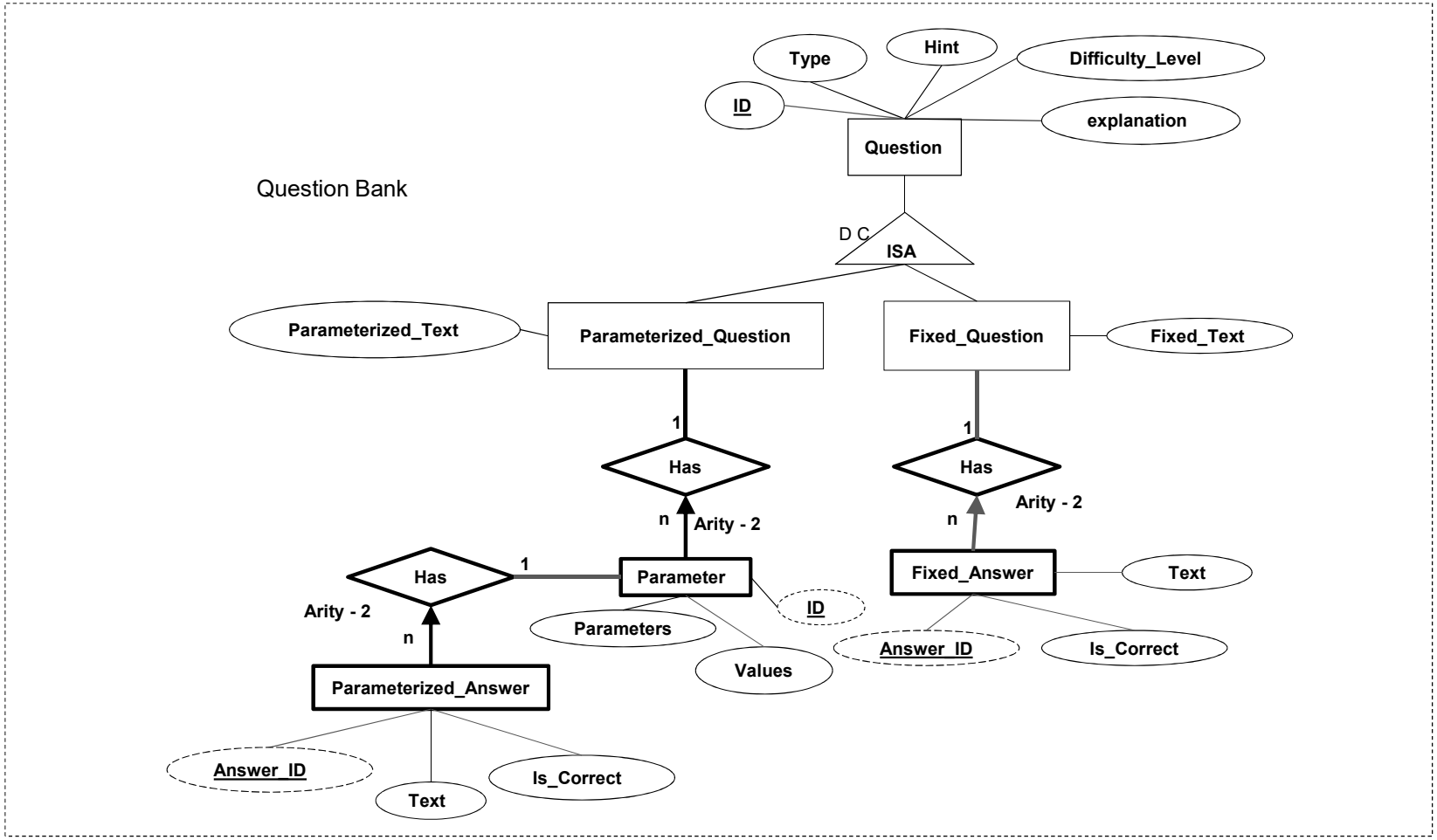


Note: The entities Student, Professor and Course are same as the entities of the same names in the next page



Note: For Weak entities the key is shown as a dotted circle (instead of a dotted underline as was taught in class)

Note: The aggregate entity Question Bank is defined in the next page



Note: For Weak entities the key is shown as a dotted circle (instead of a dotted underline as was taught in class)

Sl.No	Table name	constraints	Functional dependencies	Normal form choice/Justification
1	ASSOCIATED_TOPIC	topic_id and course_id are foreign keys which are cascaded on delete, referencing TOPIC table and COURSE table respectively, together used as primary key.	No functional dependencies. All attributes are part of the key	BCNF. There are no non trivial functional dependencies.
2	ATTEMPT_INFO	Attempt_id and Question_id form the primary key and are foreign keys referencing ATTEMPT and QUESTIONS table. None of the attributes can be NULL and is_course can only be 0 or 1..	Attempt_id, Question_id -> is_course, ans_id	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
3	ATTEMPTS	The id field is the primary key. hw_id, and student_id are FOREIGN KEYS referencing HW_EX,and STUDENT tables respectively. Hw_id, student_id, and attempt_no must NOT be NULL.	Id -> attempt_no, submission_time, total_score, total_correct_questions Id -> student_id, hw_id	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
4	COURSE_ENROLLMENT	Student_id and Course_id form the primary key and are also FOREIGN KEYS referenced from the STUDENT and COURSES tables.	No functional dependencies. All attributes are part of the key	BCNF. There are no non trivial functional dependencies.
5	COURSE	Course_id is the primary key. Instructor_id is the foreign key. Id, name, start_date, and end_date cannot be null. Check constraint for start_date < end_date.	Course_id -> name, start_date, end_date, instructor_id	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
6	FIXED_ANSWERS	Answer_id is the primary key. Question_id is a foreign key referenced from the FIXED_QUESTION table.None of the fields can be NULL.	Answer_id -> Question_id, is_correct, text	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF

Sl.No	Table name	constraints	Functional dependencies	Normal form choice/Justification
7	FIXED_QUESTION	Question_id is the primary key and is a foreign key referenced from the QUESTIONS table.	Question_id -> fixed_text	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
8	HW_EX	The id is the primary key. Instructor_id, and course_id are foreign keys from the INSTRUCTOR and COURSES tables. None of the fields can be NULL. Scoring_policy can only be ('LATEST_ATTEMPT', 'MAXIMUM_SCORE', 'AVERAGE_SCORE'). Hw_mode can only be ('STANDARD',' ADAPTIVE'). No_of_ques has to be greater than 0. Start_date has to be less than end_date.	Id -> name, corr_ans_pts, penalty_pts, start_date, end_date, scoring_policy, no_of_ques, retries, hw_mode	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
9	HW_EX_QUESTIONS	Hw_id, and question_id form the primary key. They are foreign keys referenced from the HW_EX and QUESTIONS tables. Both of them cannot be NULL.	No functional dependencies. All attributes are part of the key	BCNF. There are no non trivial functional dependencies.
10	INSTRUCTORS	Id is the primary key and also a foreign key referenced from the USERS table.	Id -> first_name, last_name	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
11	PARAMETERIZED_ANSWERS	Answer_id is the primary key. Question_id is a foreign key referenced from the PARAMETERIZED_QUESTION table. Parameter_id is a foreign key referenced from the PARAMETERS table. None of the fields can be NULL. is_correct has to be 0 or 1.	Answer_id -> is_correct, text, question_id, parameter_id	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
12	PARAMETERIZED_QUESTION	Question_id is the primary key and is a foreign key referenced from the QUESTIONS table.	Question_id -> parameterized_text	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF

Sl.No	Table name	constraints	Functional dependencies	Normal form choice/Justification
13	PARAMETERS	The id is the primary key. Question_id is a foreign key from the PARAMETERIZED_QUESTION table.	Id -> question_id, parameters, value	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
14	QUESTION_BANK	Topic_id, question_id, and course_id form the primary key and are foreign keys referenced from the TOPICS, QUESTIONS, and COURSES table. None of them can be NULL.	No functional dependencies. All attributes are part of the key	BCNF. There are no non trivial functional dependencies.
15	QUESTIONS	The id is the primary key. Difficulty_level and type cannot be NULL. Difficulty_level has to be between 0 and 7. Type has to be in ('FIXED', 'PARAMETERIZED').	Id -> difficulty_level, type, hint, explanation	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
16	ROLES	Role_id is the primary key.	Role_id -> role_name	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
17	STUDENTS	Id is the primary key and is also a foreign key referenced from the USERS table.	Id -> first_name, last_name, stud_level	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
18	TA	Id and course_id form the primary key. Id is referenced from the USERS table. Course_id is referenced from the COURSES table. Student_id is referenced from the STUDENTS table.	Id -> course_id, student_id	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
19	TOPICS	Id is the primary key.	Id -> text	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF
20	USERS	Id is the primary key. Id and password cannot be NULL. role_id is a foreign key referenced from the ROLES table.	Id -> password, role_id	BCNF. The only functional dependency has the key on the left side and hence this table is in BCNF

DBMS to implement constraints V/S implementing in Application Code

We were not able to handle following constraints as part of table definitions.

1. A student can attempt a home-work exercise more than the number of retries.
2. Only graduate student can be the TA.
3. A graduate student's level can't be updated if he is a TA.

The above constraint had dependency on multiple tables, hence we were not able to capture it as a part of table definition. However, we were able to capture those constraints using triggers. Triggers are attached in the SQL file separately.

We were able to handle all the constraints on database side except for the below.

1. User access management part. We have handled role based access through the application code.
2. When instructor creates a parameterized question, he/she must provide concrete parameter value options.