UTM
UNIVERSITI TEKNOLOGI MALAYSIA

UNIVERSITI TEKNOLOGI MALAYSIA
FACULTY OF ENGINEERING, SCHOOL OF COMPUTING
SKUDAI, 81310 JOHOR BAHRU, JOHOR DARUL TAKZIM

(ISKANDAR MALAYSIA ECOLIFE CHALLENGE)

P5 – PROJECT IMPLEMENTATION

SCSD2523-03: DATABASE

LECTURER

DR. NOR HAWANIAH ZAKARIA

PREPARED BY

MUHAMMAD AMIRUL FAHMI BIN NOOR ANIM (B19EC0018)
NURAMYRA NATASHA BINTI ISMALLUDIN (B19EC0035)
NURUL NAJIHA BINTI HAMDAN (B19EC0047)
AIMI BINTI RUSDI (B19EC0001)

Bachelor of Computer Science (Computer Graphics and Multimedia Software)

TABLE OF CONTENTS

GUIDELINES FOR P5 WORK PAPER

- Build a database based on Oracle / MySQL DBMS.
- **Database:** Create the database and tables (SQL-DDL) and identify primary key for each table. Then create relationships for your database.
- **Test Data:** Compile or generate a set of test data (at least 10 tuples or records for each table) for your relations.
- **User Interface:** Create the interface for user transaction (user view).
- **Test Queries:** Formulate at least FIFTEEN (15) SQL queries to be run against your database tables based on the transactions identified. Run the test queries on your database.
- **Application Programming:** Develop a simple interface to allow INSERT, UPDATE, DELETE and SELECT operation on your database from a form menu.

1. DATABASE PLANNING AND SYSTEM DEFINITION

1.1 EXISTING SYSTEM OVERVIEW

Iskandar Malaysia Eco-Life Challenge (IMELC) was started as Iskandar Malaysia Eco-Life Challenge in 2013 as a part of a research project funded by the Science and Technology Research Partnership for Sustainable Development (SATREPS). The aim of this research project is to create awareness of Low Carbon Society (LCS) among students, teachers and their families. At the moment, IMELC was conducted manually. Teachers were called to a venue to be trained. Workbooks are printed and distributed to all the teachers to be given to their students to answer. The workbooks, together with the proofs of other modules were later delivered to a pick up point for the organizer to pick up and checked manually.

This is then analysed to get the result of the best 15 schools to go to the final presentation. It will not be practical to run at a nationwide scale. Therefore, a computerized solution with the tailor made system must be developed to incorporate the operation of IMELC to ensure that all the students from the whole country can participate the project. This new project shall simply be called "Ecolife Challenge". All the contents of the workbook need to be made interactive. The teachers and students can access anytime, anywhere. However, from the trial in 2015, the teachers preferred a computerized solution that can allow them to monitor the students' progress and that they can submit all the relevant documents for the other modules.

1.2 DATABASE PLANNING

1.2.1 Mission Statement

To create a database management system (DBMS) for web based of the Iskandar Malaysia Eco-Life Challenge (IMELC) that create awareness of Low Carbon Society (LCS) among students, teachers and the families in achieving the target of carbon reduction for Iskandar Malaysia by year 2025.

1.2.2 Mission Objective

At the moment, IMELC was conducted manually. In this project, database is used for collecting the answer from questions provided by IMELC so that it can be easily stored, accessed, managed and updated. In this project, it can help to upgrade the student’s class activities into the use of technology. Students are needed to use the computer, logging into their individual account that is a web based system and answer the questions online. They need to submit all the relevant answer of the workbook through the system. In this case, we can also reduce the use of papers. Besides this, through this system UTM can also monitor student’s activities either they are actively participating this program or not and UTM can also check the answer for each student in the chosen schools for the prize at the end of the program.

1.3 SYSTEM DEFINITION

1.3.1 System Boundaries

This system won't provide option for user to log report for any problem. However, the system will generate feedback if the user wants to report any problems or improvements for future research. The system also allow organization to give announcement and messaging to alert students and teachers.

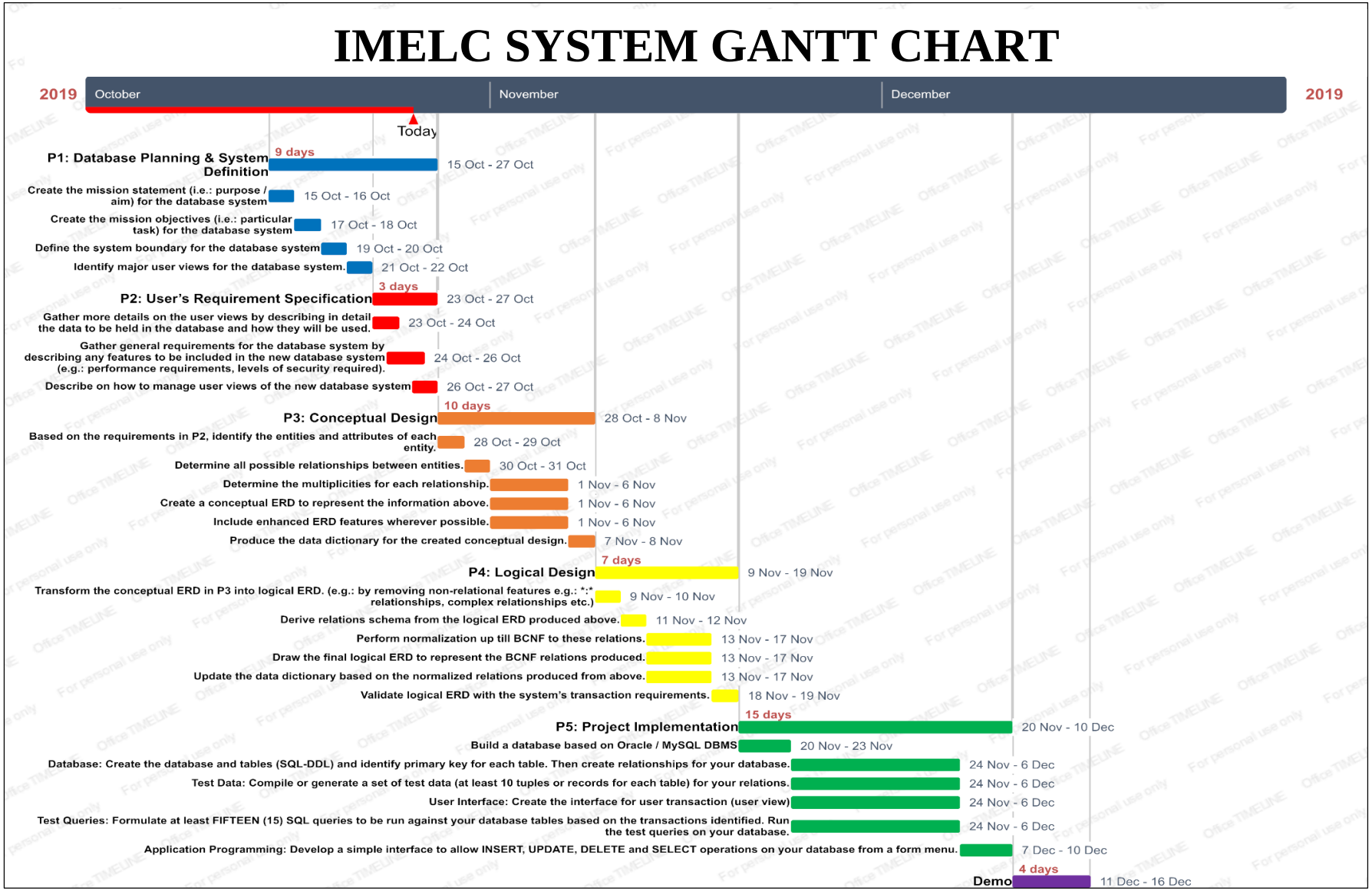
While students answer the questions, the system allows the assigned teacher and their parents to monitor them. The parents and teachers have to create an account to monitor but they cannot edit the data. After the users are done filling up the answers, the system will submit the data to the organization for data storage. The organization will provide or generate e-certificate and report through the system.

1.3.2 Major User Views

Below is the list for major user views:

Data	Access Type	Student	Teacher	Parents	Organization
Student	Maintain	**X**	**X**		**X**
	View	**X**	**X**	**X**	**X**
Parents	Maintain				
	View				
Teacher	Maintain		**X**		
	View		**X**	**X**	
Question	Maintain		**X**		**X**
	View	**X**	**X**	**X**	**X**
Answer	Maintain		**X**		**X**
	View		**X**	**X**	**X**
School	Maintain		**X**		**X**
	View		**X**	**X**	**X**
Organization	Maintain				
	View			**X**	

IMELC SYSTEM GANTT CHART



2. USER'S REQUIREMENT SPECIFICATION

2.1 DATA REQUIREMENT

Entity	Data to Be Stored	Requirements of Data
Student	1. Student ID 2. Student name 3. Student information 4. School Code	ID for the student has been created according for each school code
Parents	1. Student ID 2. Parent Name 3. Answer ID	Uses the created student ID for access of system activity of answers ID
Teacher	1. Teacher ID 2. Teacher Name 3. School Code 4. Answer ID	ID for teacher is created according to school code and has access to students answer ID
Questions	1. Challenge Sec 1 2. Challenge Sec 2 3. Challenge Sec 3	Category for each questions for the sections provided
Answer	1. Answer ID 2. Ans Challenge Sec 1 3. Ans Challenge Sec 2 4. Ans Challenge Sec 3 5. Student ID	Answers that are keyed in specifically for each question Student ID is created to detect each answer update by students
School	1. School Code 2. School information 3. Answer ID 4. School Name	Code created for each school Answer ID created for each school according to the school code
Organization	1. Grade 2. Result 3. School Code 4. Answer ID	The grade after each section for each student answer is recorded Result is sent to every school according to the school code

2.2 TRANSACTION REQUIREMENTS

Entity	Data	Data Entry	Data Update	Data Deletion	Data Queries
Student	Student ID Student Name Student Information School Code	Student add the student details	Update Student personal details	Delete student details by student themselves	Queries on Student
Parents	Student ID Parent Name Answer ID	Can only access for system activity using Student ID	Cannot update any data	Cannot delete any data	Queries on Parents
Teacher	Teacher ID Teacher Name School Code Answer ID	Teachers are providing their details	Update Teacher personal details	Delete Teacher details	Queries on Teacher
Questions	Challenge Sec 1 Challenge Sec 2 Challenge Sec 3	Questions were added by Organization	Update admin personal details	Delete Questions by Organization	Questions were answered by Student
Answer	Answer ID Ans Challenge Sec 1 Ans Challenge Sec 2	Organization views the Answer from Student	Organization updates the correct Answer details	No deletion but student can edit the answer	Answers by Student Answer was managed by Organization

	Ans Challenge Sec 3				
School	School Code	Add school code, name and information	Update school details	Delete school details	Search for School by school code, name and information School can view and sent the Answer ID to the Organization
	School Information				
	Answer ID				
	School Name				
Organization	Grade	Organization gives grade and result	Organization updates the grade and result	-	Give grade and result to the particular school code based on correct Answer ID
	Result				
	School Code				
	Answer ID				

3. ENTITY RELATIONSHIP DESIGN

In Iskandar Malaysia Eco-Life Challenge (IEMLC) database, the students will have to register their personal information first. Each student will have a unique student ID, name of the student, age of the student, gender, address of home, their city, zip code, their email and school code given by their school.

The student will be monitored by their parents. Each parent can monitor only one student at a time. A student will be monitored by one parent at a time. Their parents will have unique name, the ID of the student, school code, and student answer ID. The students will also be checked by their teacher. Many students can be checked by more than one teacher. A teacher can check more than one student. The teacher will have a unique teacher ID, name of the teacher, their school code and the answer ID.

Each student will answer questions given from the organization. The question contains challenge guide and school project. Then, the students will key in the answers. The answers will contain the answers of challenge guidance answer and answers for school project. Many questions provided can key in many answers by students. Many answers can be keyed in on many questions provided.

Before the students' hands in the answers to the teachers, parents can check the answers given by the student. The teacher will send the answers to the school. The school contain a unique school code, address of the school, city, zip code and the answer ID. A teacher will send the answers only to one school.

The school will submit the answers to the organization. The organization will have answer checked for the results, give grade to student, monitor of the school and the status of sending the results. The organization will notify to alert the parents about the answers. An organization will notify only one parents at a time about the answers provided by the students.

3.1 CONCEPTUAL ENTITY RELATIONSHIP DIAGRAM

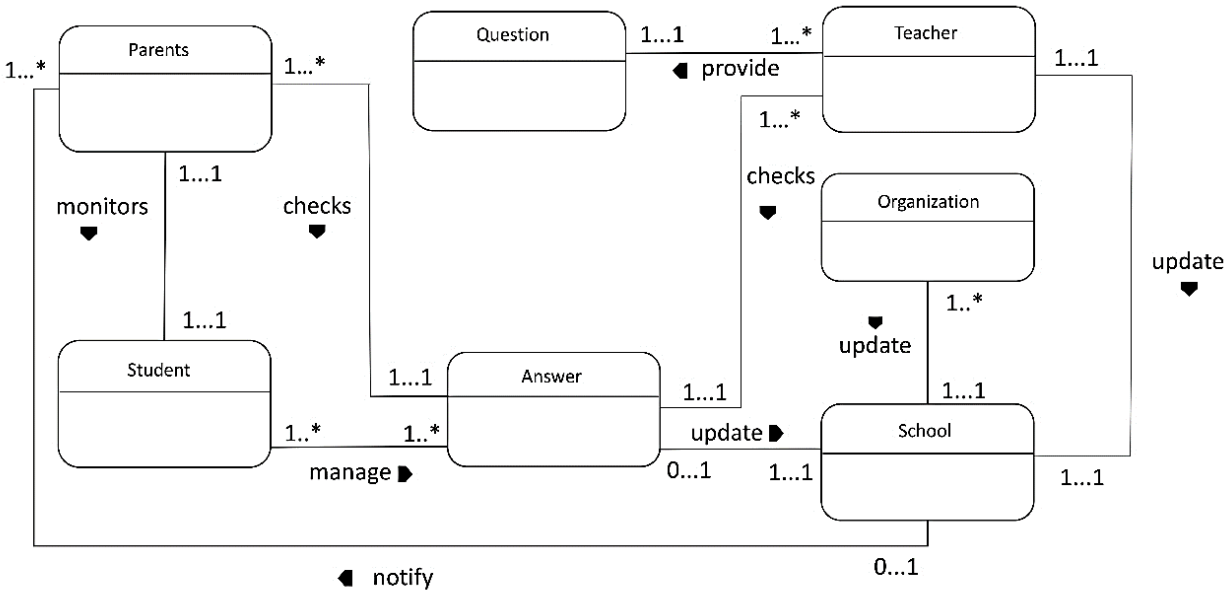


Figure 3.1. Conceptual Entity Relationship Diagram

3.2 LOGICAL ENTITY RELATIONSHIP DIAGRAM (NORMALIZED)

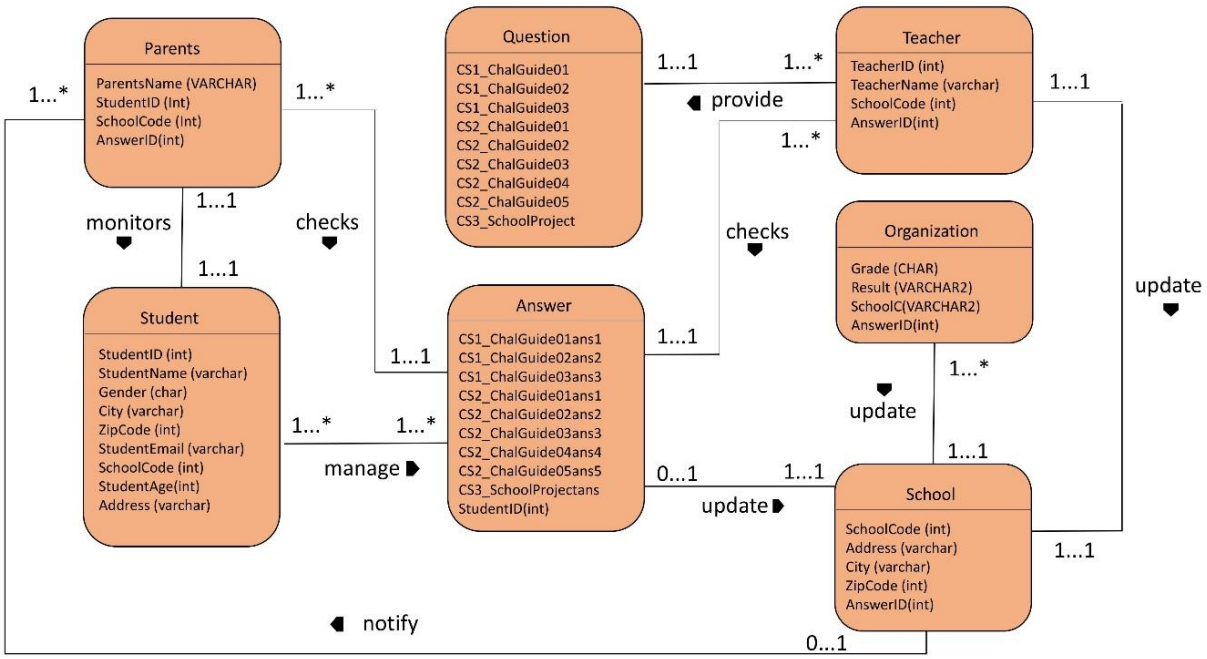
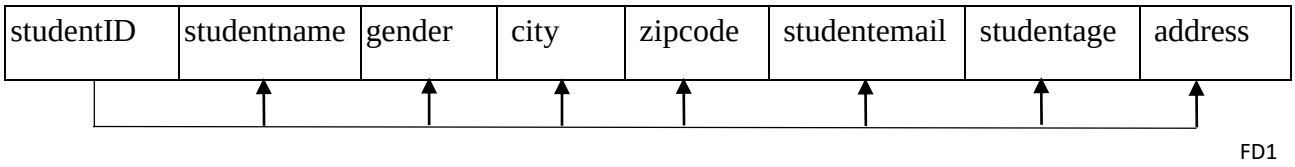


Figure 3.2. Logical Entity Relationship Diagram

3.4 DATA RELATIONAL SCHEMA

i. student (<u>studentID</u>, studentname, gender, city, zipcode, studentemail, studentage, address)

student

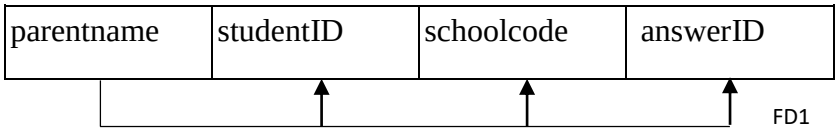


FD1: <u>studentID</u> → studentname, gender, city, zipcode, studentemail, studentage, address

FD1 is fully functional dependency. Relation STUDENT is in 3NF.

ii. parent (<u>parentname</u>, studentID, schoolcode, answerID)

parent

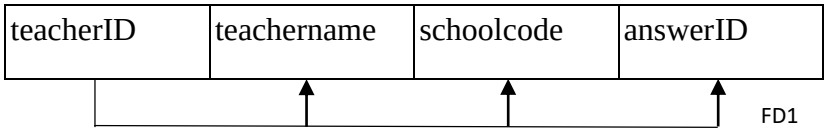


FD1: <u>parentname</u> → studentID, schoolcode, answerID

FD1 is fully functional dependency. Relation PARENTS is in 3NF.

iii. teacher (<u>teacherID</u>, teachername, schoolcode, answerID)

teacher

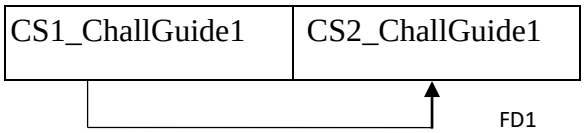


FD1: <u>teacherID</u> → teachername, schoolcode, answerID

FD1 is fully functional dependency. Relation TEACHER is in 3NF.

iv. question (CS1_ChallGuide1, CS2_ChallGuide1)

question

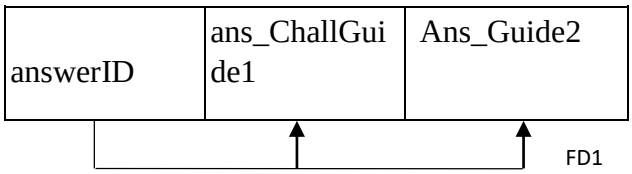


FD1: CS1_ChallGuide1, CS2_ChallGuide1

FD1 is fully functional dependency. Relation QUESTION is in 3NF.

v. answer (<u>answerID</u>, ans_ChallGuide1, ans_ChallGuide2)

answer

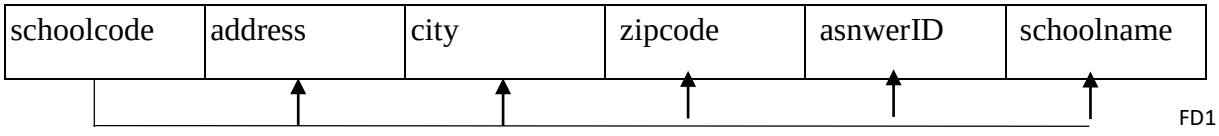


FD1: <u>answerID</u> → ans_ChallGuide1, ans_ChallGuide2

FD1 is fully functional dependency. Relation ANSWER is in 3NF.

vi. school (<u>schoolcode</u>, , address, city, zipcode, answerID, schoolname<u>)</u>

school

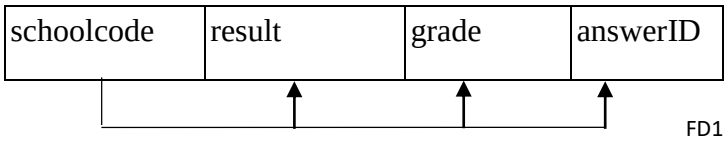


FD1: <u>schoolcode</u> → address, city, zipcode, answerID, schoolname

FD1 is fully functional dependency. Relation SCHOOL is in 3NF.

vii. organization (<u>schoolcode</u>, result, grade, answerID)

organization



FD1: <u>schoolcode</u> → result, grade, answerID

FD1 is fully functional dependency. Relation ORGANIZATION is in 3NF.

3.5 NORMALIZED DATA DICTIONARY

DATA STUDENT

Field Number	Data type	Field Size	Example	Description
studentID	number	4	0001	Unique number ID for all student
studentName	varchar2	100	Muhammad Fikri bin Muhammad Faliq	Name for student
Gender	char	1	M	Gender for Gender
City	varchar2	30	Skudai, Johor	City for the address
ZipCode	number	5	81300	Code for Zip
studentEmail	varchar2	25	fikri@gmail.com	Email for student
SchoolCode	number	10	101	The code for School
studentAge	number	2	12	The age for student
Address	varchar2	100	11, Taman Universiti	Student Address

DATA TEACHER

Field Number	Data type	Data Format	Field Size	Example
TeacherID	number	TNN	10	T01
TeacherName	Varchar2		100	Khatijah Binti Amri
SchoolCode	number	NNN	10	101
AnswerID	Number	4	1001	Foreign key represents School (AnswerID)

DATA PARENTS

Field Number	Data Type	Field Size	Example	Description
ParentName	varchar2	100	Hayati Binti Rahmat	Name for student's parents
studentID	number	4	0001	Unique number ID for all student
SchoolCode	number	10	101	Foreign key represents School (SchoolCode)
AnswerID	Number	4	1001	Foreign key represents School (AnswerID)

DATA ORGANISATION

Field Number	Data Type	Field Size	Example	Description
Grade	char	1	A, B, C	Grade for the answers
Result	varchar2	10	Passed the test	Result of the answers
SchoolCode	number	10	2014	Foreign key represents School (SchoolCode)
AnswerID	number	4	1001	Foreign key represents School (AnswerID)

DATA ANSWERS

Field Number	Data Type	Field Size	Example	Description
AnswerID	number	4	2014	Unique number ID for asnwers
CS1_ChallGuideAns1	varchar2	100	Electric	Answers for the workbook
CS1_ChallGuideAns2	varchar2	100	Water	Answers for the workbook
CS1_ChallGuideAns3	varchar2	100	Bill	Answers for the workbook
CS2_ChallGuideAns1	varchar2	100	Water	Answers for the workbook
CS2_ChallGuideAns2	varchar2	100	Electrc	Answers for the workbook
CS2_ChallGuideAns3	varchar2	100	100 kilowatts	Answers for the workbook
CS2_ChallGuideAns4	varchar2	100	Electric	Answers for the workbook
CS2_ChallGuideAns5	varchar2	100	Water	Answers for the workbook
CS3_SchoolProjectAns	varchar2	100	Ozone Layer	Answers for the workbook
StudentID	number	4	10001	Foreign key represents Student (StudentID)

DATA SCHOOL

Field Number	Data Type	Field Size	Example	Description
SchoolCode	number	10	8805	The unique code of school
Address	varchar2	100	11, Taman Universiti	The address of school
City	varchar2	50	Electric questions	City for the address
Zipcode	number	10	Water questions	Zip code for the address
AnswerID	number	4	1001	Foreign key represents Answer (AnswerID) and Unique number ID for answers
SchoolName	varchar2	100	Sekolah Kebangsaan Medini	Name of the school

DATA QUESTIONS

Field Number	Data Type	Field Size	Example	Description
CS1_ChallGuide1	varchar2	100	Electric questions	Questions for the workbook in Section 1
CS1_ChallGuide2	varchar2	100	Water bill questions	Questions for the workbook in Section 1
CS1_ChallGuide3	varchar2	100	What is the value	Questions for the workbook in Section 1
CS2_ChallGuide1	varchar2	100	The amount of water	Questions for the workbook in Section 2
CS2_ChallGuide2	varchar2	100	The amount of electricity	Questions for the workbook in Section 2
CS2_ChallGuide3	varchar2	100	The amount of bill	Questions for the workbook in Section 2
CS2_ChallGuide4	varchar2	100	The amount of electric bill	Questions for the workbook in Section 2
CS2_ChallGuide5	varchar2	100	The amount of water bill	Questions for the workbook in Section 2
CS3_SchoolProject	varchar2	100	The effect of pollution on Earth	Questions for the workbook in Section 3

4. PROJECT IMPLEMENTATION

4.1 STRUCTURED QUERY LANGUAGE

Structure for table "Student"

```
CREATE TABLE  "STUDENT"
   (    "STUDENTID" NUMBER(4,0),
        "STUDENTNAME" VARCHAR2(100),
        "GENDER" CHAR(1),
        "CITY" VARCHAR2(30),
        "ZIPCODE" NUMBER(5,0),
        "STUDENTEMAIL" VARCHAR2(25),
        "SCHOOLCODE" NUMBER(10,0),
        "STUDENTAGE" NUMBER(2,0),
        "ADDRESS" VARCHAR2(100),
         PRIMARY KEY ("STUDENTID") ENABLE
   ) ;
```

Structure for table "Parents"

```
CREATE TABLE  "PARENTS"
   (    "PARENTNAME" VARCHAR2(100),
        "STUDENTID" NUMBER(4,0),
        "SCHOOLCODE" NUMBER(10,0),
        "ANSWERID" NUMBER(4,0),
         PRIMARY KEY ("PARENTNAME") ENABLE
   ) ;
ALTER TABLE  "PARENTS"
       ADD FOREIGN KEY ("STUDENTID")
       REFERENCES  "STUDENT" ("STUDENTID") ENABLE;
ALTER TABLE  "PARENTS"
       ADD FOREIGN KEY ("SCHOOLCODE", "ANSWERID")
       REFERENCES  "SCHOOL" ("SCHOOLCODE", "ANSWERID") ENABLE;
```

Structure for table "Teacher"

```
CREATE TABLE  "TEACHER"
   (    "TEACHERID" NUMBER(10,0),
        "TEACHERNAME" VARCHAR2(100),
        "SCHOOLCODE" NUMBER(10,0),
        "ANSWERID" NUMBER(4,0),
         PRIMARY KEY ("TEACHERID") ENABLE
   ) ;
ALTER TABLE  "TEACHER"
       ADD FOREIGN KEY ("SCHOOLCODE", "ANSWERID")
       REFERENCES  "SCHOOL" ("SCHOOLCODE", "ANSWERID") ENABLE;
```

Structure for table "Answer"

```
CREATE TABLE  "ANSWER"
   (              "ANSWERID" NUMBER(4,0),
                  "CS1_CHALLGUIDEANS1" VARCHAR2(100),
                  "CS1_CHALLGUIDEANS2" VARCHAR2(100),
                  "CS1_CHALLGUIDEANS3" VARCHAR2(100),
                  "CS2_CHALLGUIDEANS1" VARCHAR2(100),
                  "CS2_CHALLGUIDEANS2" VARCHAR2(100),
                  "CS2_CHALLGUIDEANS3" VARCHAR2(100),
                  "CS2_CHALLGUIDEANS4" VARCHAR2(100),
                  "CS2_CHALLGUIDEANS5" VARCHAR2(100),
                  "CS3_SCHOOLPROJECTANS" VARCHAR2(100),
                  "STUDENTID" NUMBER(4,0),
                   PRIMARY KEY ("ANSWERID") ENABLE
   ) ;

ALTER TABLE  "ANSWER"
                 ADD FOREIGN KEY ("STUDENTID")
                 REFERENCES  "STUDENT" ("STUDENTID") ENABLE;
```

Structure for table "School"

```
CREATE TABLE  "SCHOOL"
   (    "SCHOOLCODE" NUMBER(10,0),
        "ADDRESS" VARCHAR2(100),
        "CITY" VARCHAR2(50),
        "ZIPCODE" NUMBER(10,0),
        "ANSWERID" NUMBER(4,0),
        "SCHOOLNAME" VARCHAR2(100),
         PRIMARY KEY ("SCHOOLCODE", "ANSWERID") ENABLE
   ) ;
ALTER TABLE  "SCHOOL"
       ADD FOREIGN KEY ("ANSWERID")
       REFERENCES  "ANSWER" ("ANSWERID") ENABLE;
```

Structure for table "Organisation"

```
CREATE TABLE  "ORGANISATION"
   (    "GRADE" CHAR(1),
        "RESULT" VARCHAR2(10),
        "SCHOOLCODE" NUMBER(10,0),
        "ANSWERID" NUMBER(4,0)
   ) ;
ALTER TABLE  "ORGANISATION"
       ADD FOREIGN KEY ("SCHOOLCODE", "ANSWERID")
       REFERENCES  "SCHOOL" ("SCHOOLCODE", "ANSWERID") ENABLE;
```

Structure for table "Question"

```
CREATE TABLE  "QUESTION"
   (    "CS1_CHALLGUIDE1" VARCHAR2(100),
        "CS1_CHALLGUIDE2" VARCHAR2(100),
        "CS1_CHALLGUIDE3" VARCHAR2(100),
        "CS2_CHALLGUIDE1" VARCHAR2(100),
        "CS2_CHALLGUIDE2" VARCHAR2(100),
        "CS2_CHALLGUIDE3" VARCHAR2(100),
        "CS2_CHALLGUIDE4" VARCHAR2(100),
        "CS2_CHALLGUIDE5" VARCHAR2(100),
        "CS3_SCHOOLPROJECT" VARCHAR2(100)
   ) ;
```

4.2 DATA TABLE QUERY LANGUAGE

Data for table "Student"

```
INSERT INTO Students
VALUES (1001,'NURSYAMIMI
HANIS','F','SKUDAI',81300,'mimi1212@gmail.com',1078,12,'22 JALAN
BAKTI 41');
INSERT INTO Students
VALUES (1002,'ABU
BAKAR','M','SKUDAI',81300,'abakar@gmail.com',1126,12,'40 JALAN DESA
BAKTI');
INSERT INTO Students
VALUES
(1003,'FAKHRI','M','SKUDAI',81300,'fakhri07@yahoo.com',1104,12,'1
JALAN JAMBU UTAMA');
INSERT INTO Students
VALUES (1004,'JACKSON','M','PASIR
GUDANG',81700,'jackie@yahoo.com',1109,12,'10 TAMAN BUKIT INDAH');
INSERT INTO Students
VALUES
(1005,'RAVI','M','SKUDAI',81300,'ravi2007@yahoo.com',1133,12,'37
TAMAN PULAI PERDANA');
INSERT INTO Students
VALUES (1006,'DANISHA
SHAMILA','F','SKUDAI',81300,'danishhaa@yahoo.com',1104,12,'12 JALAN
KEBANGSAAN');
INSERT INTO Students
VALUES (1007,'NURUL
ADILLA','F','SKUDAI',81300,'nrladilla@gmail.com',1078,12,'5 JALAN
KEBUDAYAAN');
INSERT INTO Students
```

```
VALUES (1008,'MUHAMMAD
SYAMIL','M','SKUDAI',81300,'syamilamir@gmail.com',1133,12,'18 JALAN
UTAMA 5');
INSERT INTO Students
VALUES (1009,'SITI
SARAH','F','SKUDAI',81300,'sitisarah07@gmail.com',1109,12,'22 JALAN
UTAMA 5');
INSERT INTO Students
VALUES
(1010,'BALAMURUGAM','M','SKUDAI',81300,'balamurugam2007@yahoo.com',1
126,12,'30 JALAN PERTANIAN 3');
INSERT INTO Students
VALUES (1011,'INTAN
KARMILA','F','SKUDAI',81300,'intnkrmila@yahoo.com',1126,12,'10 JALAN
MUTIARA MAS');
INSERT INTO Students
VALUES (1012,'MEI
YING','F','SKUDAI',81300,'meiying@gmail.com',1078,12,'16 JALAN DATO
ABDULLAH 20');
INSERT INTO Students
VALUES (1013,'AMIN
FARID','M','SKUDAI',81300,'faridamin@gmail.com',1104,12,'9 TAMAN
PULAI UTAMA');
INSERT INTO Students
VALUES (1014,'ANIES
SOFEA','F','SKUDAI',81300,'aniesss@gmail.com',1078,12,'4 JALAN
PENDIDIKAN 10');
INSERT INTO Students
VALUES (1015,'SORHANA IZZAH','F','SKUDAI',81300','NURFIQA
AIDA',1109,12,'22 JALAN BAKTI 56');
```

Data for table "Parents"

```
INSERT INTO Parents
VALUES (‘ANITA SYAHIRAH’,1001,1078,0001);
INSERT INTO Parents
VALUES ('ALYA NASUHA',1002,1126,0002);
INSERT INTO Parents
VALUES ('ZAINAL ABIDIN',1003,1104,0003);
INSERT INTO Parents
VALUES ('LILY ANNE',1004,1109,0004);
INSERT INTO Parents
VALUES ('SUBRAMANIAM',1005,1133,0005);
INSERT INTO Parents
VALUES ('AMEERAH SYAKIRA',1006,1104,0006);
INSERT INTO Parents
VALUES ('SYAFIQ IDHAM',1007,1078,0007);
INSERT INTO Parents
VALUES ('AMIR HAFIZ',1008,1133,0008);
```

```
INSERT INTO Parents
VALUES ('WAHID AINUDDIN',1009,1109,0009);
INSERT INTO Parents
VALUES ('JEREMIAH GOH',1010,1126,0010);
INSERT INTO Parents
VALUES ('NUR ATHIRAH',1011,1126,0011);
INSERT INTO Parents
VALUES ('NG FEI',1012,1078,0012);
INSERT INTO Parents
VALUES ('MUHD AZFAR',1013,1104,0013);
INSERT INTO Parents
VALUES ('AMIR FIRDAUS',1014,1078,0014);
INSERT INTO Parents
VALUES ('NURFIQA AIDA',1015,1109,0015);
```

Data for table "Teacher"

```
INSERT INTO Teacher
VALUES (2120,'ZARINA ABDULLAH',1078,1);
INSERT INTO Teachers
VALUES (6710,'FAUZIAH NAWI',1126,2);
INSERT INTO Teachers
VALUES (1020,'HISYAM RAHMAT',1104,3);
INSERT INTO Teachers
VALUES (5001,'HARIZ ZAIDI',1109,4);
INSERT INTO Teachers
VALUES (1477,'FUADDUDDIN',1133,5);
```

Data for table "Question"

```
INSERT INTO Questions
VALUES ('Electric questions','Water bill questions','What is the
value',
'The amount of water','The amount of electricity','The amount of
bill',
'The amount of electric bill','The amount of water bill','The effect
of pollution on Earth');
```

Data for table "Answer"

Answers for each Challenge Section key in by five students.

```
INSERT INTO Answer
VALUES
(0001,1001,'Save electricity','Off the lights','use when
necessary','Computers','108kgCO2','Use public
transports','Safer','Low Carbon Society','Low Carbon');

INSERT INTO Answer
VALUES (0002,1002,'Close Fridge','Save electric','play outside','Air
conditioners','98kgCO2','Carpool','Healthier','Explain the
advantages','Global Warming');

INSERT INTO Answer
VALUES (0003,1003,'Save power','Switch off','Use low
power','Television','156kgCO2','Use the train','Reduce energy','off
all electricity','Green house effect');

INSERT INTO Answer
VALUES (0004,1004,'Save electricity','Limit the time used','Only use
fan','Computers','138.2kgCO2','Carpooling','Safer
environment','Teach about global warming','Cut trees');

INSERT INTO Answer
VALUES (0005,1005,'Off electricities','Play outside','Off
lights','Refrigerator','103kgCO2','Use one car','Better air','Low
Carbon Society','Biomass');
```

Data for table "Organisation"

```
INSERT INTO Organisation
VALUES ('A','Passed',1078,0001);
INSERT INTO Organisation
VALUES ('A','Passed',1126,0002);
INSERT INTO Organisation
VALUES ('A','Passed',1104,0003);
INSERT INTO Organisation
VALUES ('A','Passed',1109,0004);
INSERT INTO Organisation
VALUES ('A','Passed',1133,0005);
INSERT INTO Organisation
VALUES ('B','Passed',1104,0006);
INSERT INTO Organisation
VALUES ('B','Passed',1104,0007);
INSERT INTO Organisation
```

```
VALUES ('B','Passed',1133,0008);
INSERT INTO Organisation
VALUES ('C','Passed',1109,0009);
INSERT INTO Organisation
VALUES ('C','Passed',1126,0010);
INSERT INTO Organisation
VALUES ('C','Passed',1126,0011);
INSERT INTO Organisation
VALUES ('F','Failed',1078,0012);
INSERT INTO Organisation
VALUES ('F','Failed',1104,0013);
INSERT INTO Organisation
VALUES ('F','Failed',1078,0014);
INSERT INTO Organisation
VALUES ('F','Failed',1109,0015);
```

Data for table "School"

```
INSERT INTO School
VALUES (1078,'JALAN PENDIDIKAN','SKUDAI',81300,0001,'SEKOLAH
KEBANGSAAN TAMAN UNIVERSITI 1');
INSERT INTO School
VALUES (1126,'JALAN BAKTI','SKUDAI',81300,0002,'SEKOLAH KEBANGSAAN
TAMAN MUTIARA RINI 2');
INSERT INTO School
VALUES (1104,'JALAN NIPAH','JOHOR BAHRU',81300,0003,'SEKOLAH
KEBANGSAAN TAMAN DAYA 2');
INSERT INTO School
VALUES (1109,'JALAN PERSISIRAN SUTERA','SKUDAI',81300,0004,'SEKOLAH
KEBANGSAAN TAMAN SUTERA');
INSERT INTO School
VALUES (1133,'EDUCITY','NUSAJAYA',79100,0005,'SEKOLAH KEBANGSAAN
MEDINI');
INSERT INTO School
VALUES (1104,'JALAN NIPAH','JOHOR BAHRU',81300,0006,'SEKOLAH
KEBANGSAAN TAMAN DAYA 2');
INSERT INTO School
VALUES (1078,'JALAN PENDIDIKAN','SKUDAI',81300,0007,'SEKOLAH
KEBANGSAAN TAMAN UNIVERSITI 1');
INSERT INTO School
VALUES (1133,'EDUCITY','NUSAJAYA',79100,0008,'SEKOLAH KEBANGSAAN
MEDINI');
INSERT INTO School
VALUES (1109,'JALAN PERSISIRAN SUTERA','SKUDAI',81300,0009,'SEKOLAH
KEBANGSAAN TAMAN SUTERA');
INSERT INTO School
```

```
VALUES (1126,'JALAN BAKTI','SKUDAI',81300,0010,'SEKOLAH KEBANGSAAN
TAMAN MUTIARA RINI 2');
INSERT INTO School
VALUES (1126,'JALAN BAKTI','SKUDAI',81300,0011,'SEKOLAH KEBANGSAAN
TAMAN MUTIARA RINI 2');
INSERT INTO School
VALUES (1078,'JALAN PENDIDIKAN','SKUDAI',81300,0012,'SEKOLAH
KEBANGSAAN TAMAN UNIVERSITI 1');
INSERT INTO School
VALUES (1104,'JALAN NIPAH','JOHOR BAHRU',81300,0013,'SEKOLAH
KEBANGSAAN TAMAN DAYA 2');
INSERT INTO School
VALUES (1078,'JALAN PENDIDIKAN','SKUDAI',81300,0014,'SEKOLAH
KEBANGSAAN TAMAN UNIVERSITI 1');
INSERT INTO School
VALUES (1109,'JALAN PERSISIRAN SUTERA','SKUDAI',81300,0015,'SEKOLAH
KEBANGSAAN TAMAN SUTERA');
```

4.3 TEST DATA QUERY LANGUAGE

1. Insert Data into Table

Insert student name ‘Amir Irfan’ into the table

```
INSERT INTO Student
VALUES (1016,'AMIR IRFAN','M',
'PASIR GUDANG',81700,'amir_irfan@gmail.com',1126,12,'32 TAMAN NUSA
DAMAI');
```

STUDENTID	STUDENTNAME	GENDER	CITY	ZIPCODE	STUDENTEMAIL	SCHOOLCODE	STUDENTAGE	ADDRESS
1001	NURSYAMIMI HANIS	F	SKUDAI	81300	mimi1212@gmail.com	1078	12	22 JALAN BAKTI 41
1002	ABU BAKAR	M	SKUDAI	81300	abakar@gmail.com	1126	12	40 JALAN DESA BAKTI
1016	AMIR IRFAN	M	PASIR GUDANG	81700	amir_irfan@gmail.com	1126	12	32 TAMAN NUSA DAMAI
1003	FAKHRI	M	SKUDAI	81300	fakhri07@yahoo.com	1104	12	1 JALAN JAMBU UTAMA
1004	JACKSON	M	PASIR GUDANG	81700	jackson@gmail.com	1109	12	10 TAMAN BUKIT INDAH

Figure 4.3.1 Insert Data in Student Table

2. Update Data in Table

Update student email in the table

```
UPDATE Student
SET StudentEmail = 'ameerfan@gmail.com'
WHERE StudentID = 1016;
```

STUDENTID	STUDENTNAME	GENDER	CITY	ZIPCODE	STUDENTEMAIL	SCHOOLCODE	STUDENTAGE	ADDRESS
1001	NURSYAMIMI HANIS	F	SKUDAI	81300	mimi1212@gmail.com	1078	12	22 JALAN BAKTI 41
1002	ABU BAKAR	M	SKUDAI	81300	abakar@gmail.com	1126	12	40 JALAN DESA BAKTI
1016	AMIR IRFAN	M	PASIR GUDANG	81700	ameerfan@gmail.com	1126	12	32 TAMAN NUSA DAMAI
1003	FAKHRI	M	SKUDAI	81300	fakhri07@yahoo.com	1104	12	1 JALAN JAMBU UTAMA
1004	JACKSON	M	PASIR GUDANG	81700	jackson@gmail.com	1109	12	10 TAMAN BUKIT INDAH

Figure 4.3.2 Update student email in Student Table

3. Delete Data from Table

Delete student name ‘Amir Irfan’ from table

```
DELETE
FROM Student
WHERE StudentID = 1016;
```

STUDENTID	STUDENTNAME	GENDER	CITY	ZIPCODE	STUDENTEMAIL	SCHOOLCODE	STUDENTAGE	ADDRESS
1001	NURSYAMIMI HANIS	F	SKUDAI	81300	mimi1212@gmail.com	1078	12	22 JALAN BAKTI 41
1002	ABU BAKAR	M	SKUDAI	81300	abakar@gmail.com	1126	12	40 JALAN DESA BAKTI
1003	FAKHRI	M	SKUDAI	81300	fakhri07@yahoo.com	1104	12	1 JALAN JAMBU UTAMA
1004	JACKSON	M	PASIR GUDANG	81700	jackson@gmail.com	1109	12	10 TAMAN BUKIT INDAH

Figure 4.3.3 Delete student information

4. Select Data from Table

Select Student and Parents table to show information

```
SELECT StudentName, Student.StudentID, ParentName
FROM Student, Parents
WHERE Student.StudentID = Parents.StudentID;
```

STUDENTNAME	STUDENTID	PARENTNAME
NURSYAMIMI HANIS	1001	ANITA SYAHIRAH
ABU BAKAR	1002	ALYA NASUHA
FAKHRI	1003	ZAINAL ABIDIN
JACKSON	1004	LILY ANNE
RAVI	1005	SUBRAMANIAM
MEI YING	1012	NG FEI
AMIN FARID	1013	MUHD AZFAR
ANIES SOFEA	1014	AMIR FIRDAUS
SORHANA IZZAH	1015	NURFIQA AIDA
DANISHA SHAMILA	1006	AMEERAH SYAKIRA
NURUL ADILLA	1007	SYAFIQ IDHAM
MUHAMMAD SYAMIL	1008	AMIR HAFIZ
SITI SARAH	1009	WAHID AINUDDIN
BALAMURUGAM	1010	JEREMIAH GOH
INTAN KARMILA	1011	NUR ATHIRAH

Figure 4.3.4 Select Data of student and parents

Select Student and Parents table then join with Answer ID to show information

```
SELECT StudentName, Student.StudentID, AnswerID, ParentName
FROM Student
NATURAL JOIN Parents;
```

STUDENTNAME	STUDENTID	ANSWERID	PARENTNAME
NURSYAMIMI HANIS	1001	1	ANITA SYAHIRAH
ABU BAKAR	1002	2	ALYA NASUHA
FAKHRI	1003	3	ZAINAL ABIDIN
JACKSON	1004	4	LILY ANNE
RAVI	1005	5	SUBRAMANIAM
MEI YING	1012	12	NG FEI
AMIN FARID	1013	13	MUHD AZFAR
ANIES SOFEA	1014	14	AMIR FIRDAUS
SORHANA IZZAH	1015	15	NURFIQA AIDA
DANISHA SHAMILA	1006	6	AMEERAH SYAKIRA
NURUL ADILLA	1007	7	SYAFIQ IDHAM
MUHAMMAD SYAMIL	1008	8	AMIR HAFIZ
SITI SARAH	1009	9	WAHID AINUDDIN
BALAMURUGAM	1010	10	JEREMIAH GOH
INTAN KARMILA	1011	11	NUR ATHIRAH

Figure 4.3.5 Select Data join tables

4.4 DATA FLOW DIAGRAM

4.4.1 DFD Level 0 Diagram

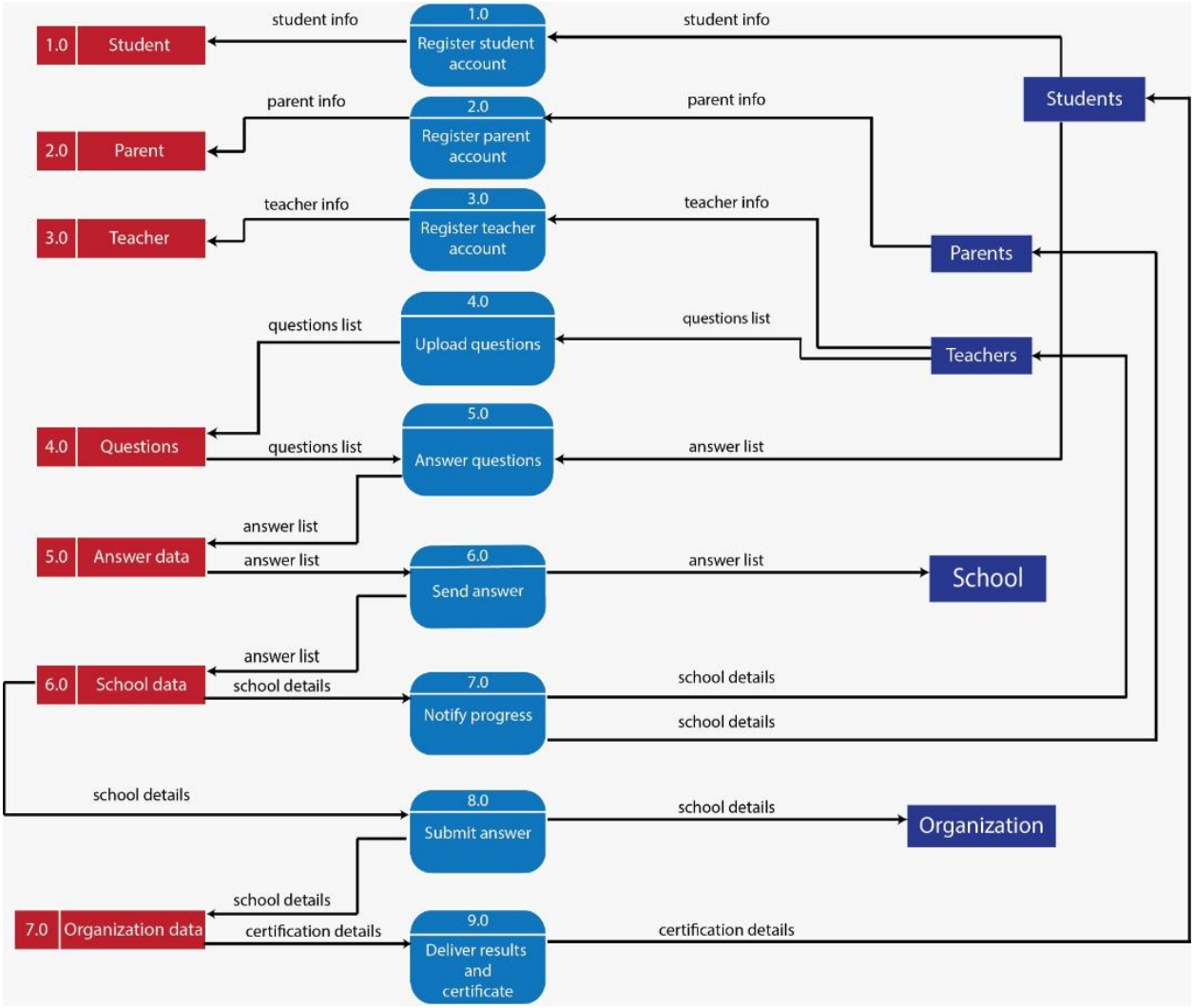


Figure 4.4.1. DFD Level 0 Diagram.