System Analysis

SCHOOL DATABASE

Group Members

1. SENG CHETRA
2. PUN MENGLY
3. PHAT CHETRA
4. TRY SORTA

INTRODUCTION

Behind every successful system, there's always a good design and planning. Before we can make a reliable and stable database, System analysis is required. That's why, our team took this approach and put our best efforts into designing a simplified, yet reliable database. We took our time and made our own system analysis to help us virtualize how our database would look like, understand what attributes should be included in each table, study how tables are connected, and we've also gathered some data to test when implementing our database.

TABLES

Having a lot of data doesn't mean our database is ready to use. We need ways to construct and categorize how those data should be organized. Our database's main purpose is to help managing students' grades in our school. Therefore, any information relating students' grade such are teachers, courses, grades are mandatory. We've tested many ways to simplified how our students' data should be categorize inside specific tables.

1. MANDATORY TABLES

To successfully structure and organize the data, we need to understand the main focus and the fundamental tables inside our database. As previously mentioned, information about a particular student is essential for our database. That's why the tables listed below will definitely be included:

- Students table: this table should all information about a specific student
- Grades table: the most important table and sole purpose of our database creation
- Teachers table: schools need teachers, so this table is required.
- Courses table: this table will show the courses that students have enrolled for
- Subjects: every course has subjects, so this table is required.

2. OTHER SUPPORTED TABLES

Although the tables listed below are crucial, some tables must be not be overlooked. Some tables might not be our primary focus in the system, but they exist to support our database. Without them, the database would not be able to function properly, and these tables include:

- Enrollments table: a link between students and courses tables
- Instructors table: a link between subjects and teachers tables
- Attendances table: this table should also be included.
- Parents table: this table should include students' parents contact

METADATA

To make sure our system is reliable, our database should be able store as many crucial information as we can, that relates to students. In order to achieve that, our team has decided that each table will have the columns as list below:

1. **students table**
- stu_id: student's ID, datatype INT and can't be null
- stu_name: student's name, datatype VARCHAR(25) and can't be null
- stu_gender: student's gender, datatype CHAR(1) and can be null
- stu_bd: student's birthday, datatype DATE and can't be null
- stu_phone: student's phone number, datatype VARCHAR(10) and can't be null
- stu_address: student's address, datatype VARCHAR(100)
- stu_class: student's class, datatype VARCHAR(4) and can't be null
- stu_year: student's academic year, datatype SMALLINT and can't be null
- stu_status: datatype BIT, 1 if the student is active in school and can't be null
- parent_id: datatype INT and will be used in database's relationship

2. **attendances table**
- att_id: student's attendance ID, datatype INT and can't be null
- att_date: attendance's date, datatype DATETIME and can't be null
- att_status: datatype BIT, 1 if attended and 0 if absent, can't be null
- att_remark: datatype VARCHAR(25), attendance's note such as late or permission
- stu_id: datatype INT and will be used in database's relationship

3. **parents table**
- parent_id: parent's ID, datatype INT and can't be null
- mother_name: mother's name, datatype VARCHAR(25) but can be null if diseased
- father:_name father's name, datatype VARCHAR(25) but can be null if diseased
- parent_phone: parent's phone number, datatype VARCHAR(10) and can't be null
- parent_address: parent's address, datatype VARCHAR(100)

4. **enrollments table**
- course_id: datatype INT and will be used in database's relationship
- stu_id: datatype INT and will be used in database's relationship
- enroll_id: enrollment's ID, datatype INT and can't be null
- enroll_date: enrollment's date, datatype DATE and can't be null

5. **courses table**
- course_id: course's ID, datatype INT and can't be null
- course_name: course's name, datatype VARCHAR(25) and can't be null
- course_dur: course's duration, datatype TINYINT and can't be null
- course_des: course's description, datatype VARCHAR(100) can be null

6. **instructors table**
- sub_id: datatype INT and will be used in database's relationship
- teach_id: datatype INT and will be used in database's relationship
- instruct_id: instructor's ID, data type INT and can't be null

7. **subjects table**
- sub_id: subject's ID, datatype INT and can't be null
- sub_name: subject's name, datatype VARCHAR(25) and can't be null
- sub_credit: subject's total credit that students need to take, datatype TINYINT and can't be null
- sub_startdate: first day of the subject, datatype DATE and can't be null
- sub_enddate: last day of the subject, datatype DATE and can't be null
- course_id: datatype INT and will be used in database's relationship

8. **grades table**
- grade_id: grade's ID, datatype INT and can't be null
- grade_score: the given score on a particular subject, datatype TINYINT and can't be null
- grade_year: the academic year of the grade, datatype TINYINT and can't be null
- grade_semester: grade's semester either 1 or 2, datatype TINYINT and can't be null
- sub_id: datatype INT and will be used in database's relationship
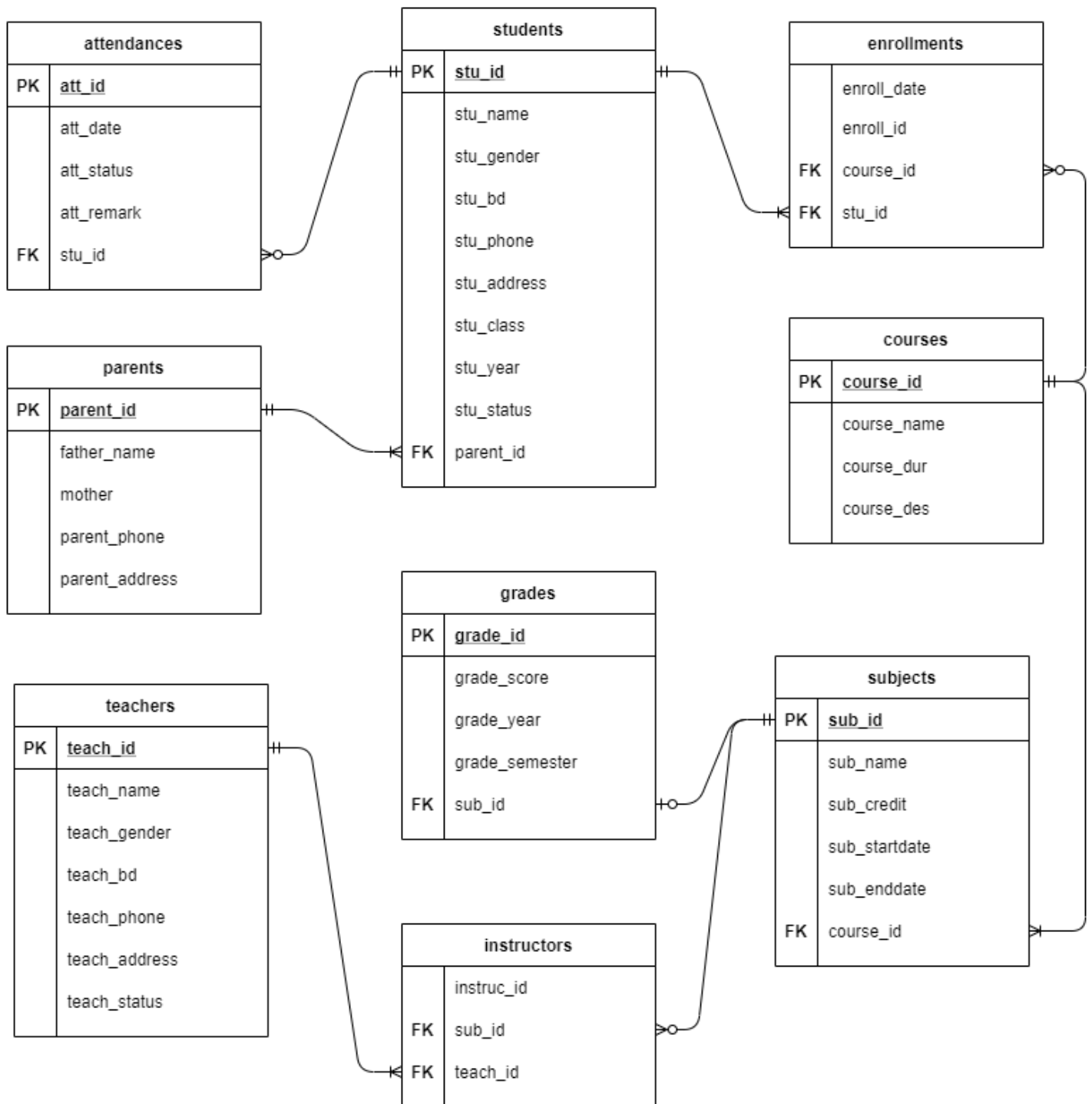
9. **teachers table**
- teach_id: teacher's ID, datatype INT and can't be null
- teach_name: teacher's name, datatype VARCHAR(25) and can't be null
- teach_gender: teacher's gender, datatype CHAR(1) but can be null
- teach_bd: teacher's birthday, datatype DATE and can be null
- teach_phone: teacher's phone number, datatype VARCHAR(10) and can't be null
- teach_address: teacher's address, datatype VARCHAR(150) can be null
- teach_status: datatype BIT and should 1 if the teacher is still teaching, can't be null.

CONSTRAINTS

We've categorized the data into multiple tables to further improve our database's data organization and management, but there's still one more step to complete our design. Having a well-structured organization is nice, but giving those data the ability to reference each other is more preferable. That is when constraints come into play. Constraints are the core fundamentals in relational database design and without them, our well-organized data wouldn't be able to communicate, making them unmanageable.

We'll start by adding primary keys and foreign keys to our tables to enforce table relationship allowing all the tables to communicate with each other. This will in turn, give us the ability to perform advances operations in database such as joining tables for instances. That's why our team have made an **Entity Relationship Diagram** to demonstrate our database's table relationships.

School Database ERD

attendances

PK	att_id
	att_date
	att_status
	att_remark
FK	stu_id

students

PK	stu_id
	stu_name
	stu_gender
	stu_bd
	stu_phone
	stu_address
	stu_class
	stu_year
	stu_status
FK	parent_id

enrollments

	enroll_date
	enroll_id
FK	course_id
FK	stu_id

parents

PK	parent_id
	father_name
	mother
	parent_phone
	parent_address

courses

PK	course_id
	course_name
	course_dur
	course_des

grades

PK	grade_id
	grade_score
	grade_year
	grade_semester
FK	sub_id

teachers

PK	teach_id
	teach_name
	teach_gender
	teach_bd
	teach_phone
	teach_address
	teach_status

subjects

PK	sub_id
	sub_name
	sub_credit
	sub_startdate
	sub_enddate
FK	course_id

instructors

	instruc_id
FK	sub_id
FK	teach_id

As shown in the diagram above:

- **parents** has parent_id as its primary key and has one to many relationship with students
- **attendances** has att_id as its primary key and stu_id as its foreign key
- **enrollments** has course_id and stu_id as its foreign keys with no primary key
- **students** has stu_id as the primary key and parent_id as the foreign key, it also has one to many relationship with **enrollments**
- **courses** has the primary key of course_id, and it has one to many relationship with **enrollments** and one to many relationship with **subjects**
- **subjects** has sub_id as its primary key and also have one to one relationship with **grades** and one to many relationship with **instructors**
- **grades** has grade_id as the primary key and sub_id as its foreign key.
- **instructors** has sub_id and teach_id as its foreign keys with no primary key
- **teachers** has teach_id as the primary key.

Among these table relationships, it's also important to note that:

- **students** table has many to many relationship with **courses** table as a student can register for multiple courses and one course can also have many students, it was made possible due to the creation of **enrollments** table
- **teachers** and **subjects** is also a many to many relationship because a teacher can't teach multiple subjects and a subject can be also taught by multiple teachers, and **instructors** table is a link between the two tables.
- Additional constraints will also be added to reinforce referential integrity in our database to ensure data consistencies among all the tables. We'll be adding more constraints to prevent unauthorized insertion or deletion. On top of that, we'll also add clustered and non-clustered in indexes to improve our database speed when retrieving data. Keep all the data safe while maintaining speed is always our priority and we'll be doing our best to improve referential integrity and prevent any data violation with the constraints.

TESTING DATA

While database design is probably one of the most important part of system analysis, we will also need data before we get into system implementation. Finding data beforehand might pave the way for the following stage as we will get more time to focus only on building our system rather wasting time finding data to test the database. Moreover, it can save us more testing time giving our team more opportunity to improve our system. That's the reason why, we have gathered serval data to prepare for system implementation.

1. COURSES

courses_id	courses_name	courses_dur	courses_des
1	Information Technology	4	
2	Mathematic	4	
3	English Literature	4	
4	Khmer Literature	4	
5	Physics	4	
6	International Relation	4	
7	Tourism	4	
8	Business	4	
9	Management	4	
10	Chemistry	4	
11	Biology	4	
12	Environment Science	4	
13	Sociology	4	
14	Psychology	4	
15	Accounting	4	
16	Linguistics	4	
17	Information Engineering	4	
18	Philosophy	4	
19	Geography	4	
20	Photography	4	

2. TEACHERS

teach_id	teach_name	teach_gender	teach_bd	teach_phone	teach_address	teach_st
1000001	Hun Sroy	M	1988-01-28	010251699	Phnom Penh	1
1000002	Kan Leang	M	1979-02-19	016125899	Kampong Cham	1
1000003	So Pheak	M	1995-03-13	092888799	Phnom Penh	1
1000004	Thea Srun	M	1978-04-18	08828899	Battambang	1
1000005	Vy Thida	F	1981-05-06	011231892	Banteay Meanchey	1
1000006	Dy Mangky	M	1988-06-09	012888859	Phnom Penh	1
1000007	Chan Dara	M	1985-02-12	012356865	Phnom Penh	1
1000008	Chhom Davit	M	1987-12-30	015522525	Phnom Penh	1
1000009	Thy Danuth	M	1991-08-12	012250654	Prey Veng	1
1000010	Kong Lida	F	1973-05-01	015580425	Phnom Penh	1
1000011	Long Rathana	M	1899-02-20	011522501	Phnom Penh	1
1000012	Lin Darith	M	1990-02-20	010020302	Phnom Penh	1
1000013	Kin Dina	F	1994-04-30	098474521	Phnom Penh	1
1000014	Lin Naroth	M	1994-12-06	098461521	Phnom Penh	1
1000015	Yin Vanuth	M	1989-04-04	012580656	Kampong Cham	1
1000016	Dy Rotha	F	1991-02-06	093543524	Phnom Penh	1
1000017	Yuth Chivon	M	1978-08-25	012823871	Takav	1
1000018	Hong Vitou	M	1987-12-30	016557854	Battambang	1
1000019	Chea Davin	M	1977-04-11	012542487	Phnom Penh	1
1000020	Chak Riya	F	1992-12-05	092518792	Phnom Penh	1

3. STUDENTS

stu_id	stu_name	stu_ge	stu_bd	stu_pho	stu_address	stu_cl	stu_ye	stu_s	parent
1000001	Bo Phanit	M	1999-01-09	097123456	Phnom Penh	B107	2	1	1000006
1000002	Pal Bora	M	2000-02-08	070133456	Battambang	B107	2	1	1000010
1000003	Nget Sayha	M	2001-03-07	097143456	Phnom Penh	B107	2	1	1000005
1000004	Mut Tola	M	2000-04-06	077153456	Phnom Penh	B107	2	1	1000004
1000005	Srey Lin	F	2000-05-05	097163456	Takeo	B107	2	1	1000001
1000006	Ma Brosa	F	2000-06-04	092173456	Phnom Penh	B107	2	1	1000020
1000007	Reach Bot	M	2000-07-03	097183456	Battambang	B107	2	1	1000002
1000008	Kao Kanha	F	2000-08-27	012193456	Phnom Penh	B107	2	1	1000003
1000009	So Polin	M	2000-09-01	088103456	Phnom Penh	B107	2	1	1000015
1000010	So Visal	M	2000-10-11	097123416	Battambang	B107	2	1	1000011
1000011	Meas Kakda	M	1998-11-20	097123426	Phnom Penh	B107	2	1	1000014
1000012	Sin Sophea	F	2000-12-12	016123436	Phnom Penh	B107	2	1	1000013
1000013	Ly Ta	F	2001-01-01	097123446	Phnom Penh	B107	2	1	1000012
1000014	Seam Sothy	M	2001-02-02	093123466	Phnom Penh	B107	2	1	1000018
1000015	Thean Long	M	2001-03-03	097123476	Takeo	B107	2	1	1000009
1000016	Mao Sola	M	2001-04-14	010123486	Phnom Penh	B107	2	1	1000017
1000017	Vi Lady	F	2001-05-05	097123496	Phnom Penh	B107	2	1	1000007
1000018	Sen Mesa	M	2001-07-18	097123406	Battambang	B107	2	1	1000016
1000019	Ma Sudan	F	2001-04-14	096123457	Phnom Penh	B107	2	1	1000008
1000020	Nhom China	M	2001-05-24	097123458	Phnom Penh	B107	2	1	1000019

4. SUBJECTS

sub_id	sub_name	sub_credit	sub_startdate	sub_enddate	course_id
1	C Programming 1	25	2019-11-06	2020-03-05	1
2	Fundamental Computer 1	25	2019-11-09	2020-03-04	1
3	Mathematics 1	25	2019-11-07	2020-03-05	1
4	Electronic 1	25	2019-11-07	2020-03-07	1
5	21st Century	25	2019-11-06	2020-03-04	1
6	English for Computer 1	25	2019-11-08	2020-03-08	1
7	C Programming 2	25	2020-07-28	2020-11-03	1
8	Fundamental Computer 2	25	2020-07-27	2020-11-02	1
9	Electronic 2	25	2020-07-30	2020-11-05	1
10	Khmer civilization	25	2020-07-29	2020-11-04	1
11	History	25	2020-07-27	2020-11-02	1
12	Mathematic 2	25	2020-07-28	2020-11-03	1
13	English 2	25	2020-07-27	2020-11-06	1
14	Database System 1	25	2020-12-24	2021-04-22	1
15	C++ Programming 1	25	2020-12-24	2021-04-22	1
16	Computer Architecture 1	25	2020-12-21	2021-04-19	1
17	Data structure 1	25	2020-12-23	2021-04-21	1
18	Data communication 1	25	2020-12-21	2021-04-19	1
19	English 3	25	2020-12-22	2021-04-20	1
20	Statistic Math 1	25	2019-11-06	2020-03-05	2

5. GRADES

grade_id	grade_score	grade_year	grade_semester	sub_id
1	90	2	1	14
2	73	2	1	14
3	92	2	1	14
4	86	2	1	14
5	95	2	1	14
6	81	2	1	14
7	82	2	1	14
8	74	2	1	14
9	81	2	1	14
10	96	2	1	14
11	88	2	1	14
12	90	2	1	14
13	87	2	1	14
14	70	2	1	14
15	90	2	1	14
16	87	2	1	14
17	92	2	1	14
18	73	2	1	14
19	85	2	1	14
20	90	2	1	14

6. PARENTS

parent_id	father	mother	parent_phone	Par_address
1000001	Chan Sopheak	So Lina	0979869751	Phnom Penh
1000002	Sorn Dara	Heng Dalin	098964512	Kampot
1000003	Toch Vannut	Sok Sreynit	0978732324	Kampong Cham
1000004	Ly Mengleang	Heng Sevmei	098786898	Phnom Penh
1000005	Eng Davith	Soth Kaknika	098246679	Phnom Penh
1000006	Som Narith	Van Dina	086836497	Phnom Penh
1000007	Vuth Oudom	Chan Sreyneang	0976828690	Phnom Penh
1000008	Vann Din	Chhang Lyly	0972342423	Kampot
1000009	Chhun Davit	Heng Reaksmey	087123741	Kampong Thom
1000010	Hort Ratana	Ear Hengly	083479713	Kampong Cham
1000011	Lun Kimheng	Lay Dary	012372342	Phnom Penh
1000012	Kan Chetra	Sim Sina	092414144	Phnom Penh
1000013	Choun Visal	Srang Lika	098213445	Phnom Penh
1000014	Seng Vibol	Heng Menghorn	099761334	Kampong Cham
1000015	Try Kimseng	Ly Yanet	012243432	Phnom Penh
1000016	Hun Davin	Si Sreytoch	098348934	Phnom Penh
1000017	Van Henglong	Sreang Sreylin	095634223	Phnom Penh
1000018	An Vandet	Leng Sinat	076435623	Phnom Penh
1000019	Tha Darith	Ly Kunthea	0971343134	Phnom Penh
1000020	Sorn Vannak	Pich Monita	099112345	Phnom Penh

7. ENROLLMENTS

enroll_id	Enroll_date	course_id	stu_id
1	2019-09-25	1	1000001
2	2019-09-26	1	1000002
3	2019-09-25	1	1000003
4	2019-09-26	1	1000004
5	2019-09-28	1	1000005
6	2019-09-28	1	1000006
7	2019-09-28	1	1000007
8	2019-10-01	1	1000008
9	2019-10-03	1	1000009
10	2019-10-06	1	1000010
11	2019-10-06	1	1000011
12	2019-10-07	1	1000012
13	2019-10-10	1	1000013
14	2019-10-12	1	1000014
15	2019-10-13	1	1000015
16	2019-10-13	1	1000016
17	2019-10-17	1	1000017
18	2019-10-18	1	1000018
19	2019-10-25	1	1000019
20	2019-11-01	1	1000020

8. INSTRUCTORS

instruc_id	teach_id	sub_id
1	1000006	14
2	1000007	14
3	1000008	14
4	1000003	15
5	1000002	15
6	1000004	16
7	1000005	16
8	1000001	15
9	1000017	17
10	1000018	18
11	1000019	18
12	1000015	17
13	1000014	19
14	1000013	19
15	1000016	17
16	1000009	16
17	1000020	18
18	1000012	19
19	1000010	1
20	1000011	1

9. ATTENDANCES

att_id	att_date	att_status	att_remark	stu_id
1	2020-12-24 17:00:00	1		1000001
2	2020-12-24 17:00:00	1		1000002
3	2020-12-24 17:00:00	1		1000003
4	2020-12-24 17:00:00	1		1000004
5	2020-12-24 17:00:00	0	Permission	1000005
6	2020-12-24 17:00:00	1		1000006
7	2020-12-24 17:00:00	0		1000007
8	2020-12-24 17:00:00	0		1000008
9	2020-12-24 17:00:00	1		1000009
10	2020-12-24 17:00:00	1		1000010
11	2020-12-24 17:00:00	1		1000011
12	2020-12-24 17:00:00	1		1000012
13	2020-12-24 17:00:00	1		1000013
14	2020-12-24 17:00:00	0	Permission	1000014
15	2020-12-24 17:00:00	1		1000015
16	2020-12-24 17:00:00	1		1000016
17	2020-12-24 17:00:00	1		1000017
18	2020-12-24 17:00:00	1		1000018
19	2020-12-24 17:00:00	1	Late	1000019
20	2020-12-24 17:00:00	1	Late	1000020

SUMMARY

In conclusion, we've looked at tables that are needed for the database and we also saw the attributes within each table. We've covered several constraints specifically tables relationship and more constraints will be added later in system implementation. As previously mentioned, we also need to add some fake data to pave the way for the next stage and to save us time. These sum up our plan and system analysis for our database. We're looking forward to implement these into our final product.