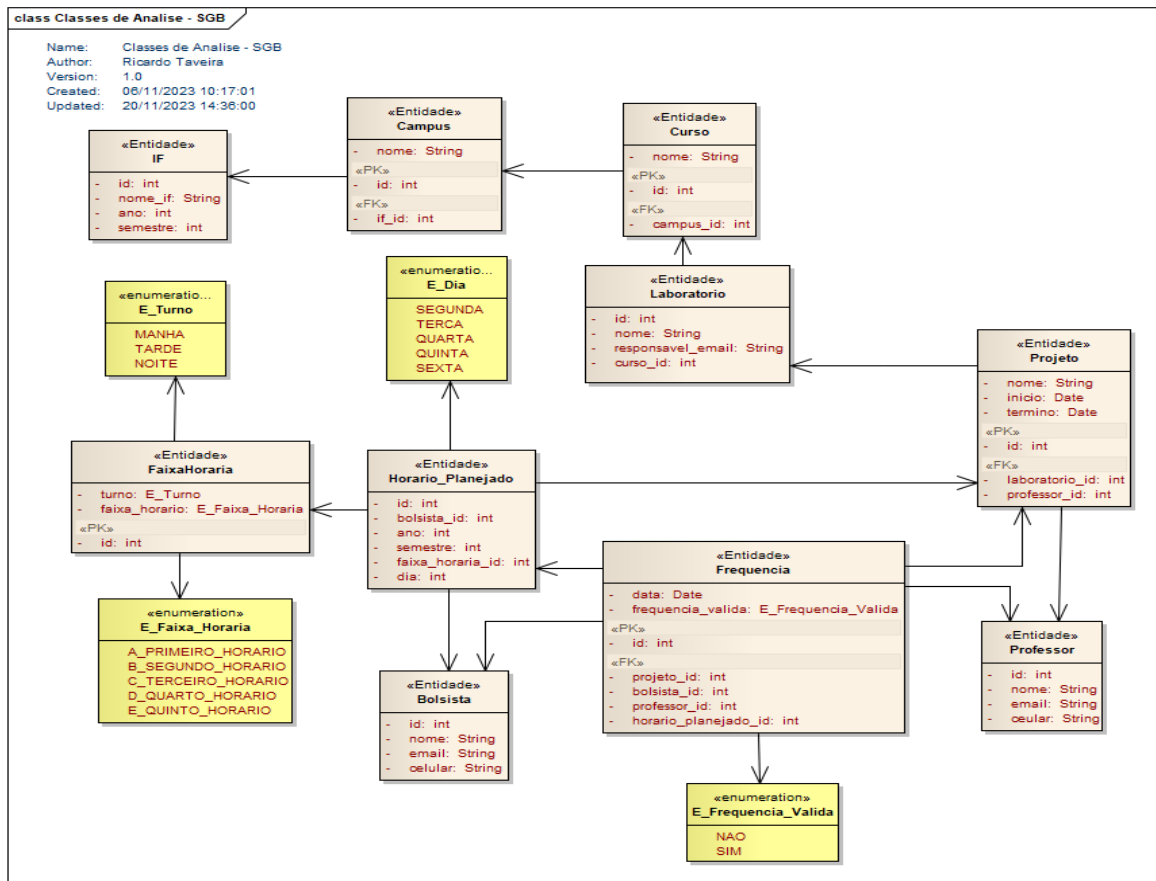


# AVALIAÇÃO 08 - BANCO DE DADOS

Aluno: Anderson Maia Santos  
Curso: P4 de Informática



```
SQLite version 3.43.1 2023-09-11 12:01:27
Enter ".help" for usage hints.
sqlite> .schema
CREATE TABLE TB_IF (id integer primary key autoincrement, nome_if text not null, ano int not null, semestre int not null);
CREATE TABLE sqlite_sequence(name,seq);
CREATE TABLE TB_CAMPUS (id integer primary key autoincrement, nome text not null, if_id int not null, foreign key(if_id) references TB_IF(id));
CREATE TABLE TB_CURSO (id integer primary key autoincrement, nome text not null, campus_id int not null, foreign key (campus_id) references TB_CAMPUS (id));
CREATE TABLE TB_LABORATORIO (id integer primary key autoincrement, nome text not null, responsavel_email text not null, curso_id int not null, foreign key(curso_id) references TB_CURSO(id));
CREATE TABLE TB_PROFESSOR (id integer primary key autoincrement, nome text not null, email text not null, celular text not null);
CREATE TABLE TB_PROJETO (id integer primary key autoincrement, nome text not null, inicio date not null, termino date not null, laboratorio_id int not null, professor_id int not null, foreign key (laboratorio_id) references TB_LABORATORIO(id), foreign key (professor_id) references TB_PROFESSOR(id));
CREATE TABLE TB_BOLSISTA (id integer primary key autoincrement, nome text not null, email text not null, celular text not null);
CREATE TABLE TB_FREQUENCIA (id integer primary key autoincrement, data date not null, frequencia_valida text check(frequencia_valida in('NAO', 'SIM')), projeto_id int not null, bolsista_id int not null, professor_id int not null, horario_planejado_id int not null, foreign key (projeto_id) references TB_PROJETO(id), foreign key (bolsista_id) references TB_BOLSISTA(id), foreign key(professor_id) references TB_PROFESSOR(id), foreign key(horario_planejado_id) references TB_HORARIO_PLANEJADO(id));
CREATE TABLE TB_HORARIO_PLANEJADO (id integer primary key autoincrement, bolsista_id int not null, ano int not null, semestre int not null, faixa_horaria_id int not null, dia text check(dia in('SEGUNDA', 'TERCA', 'QUARTA', 'QUINTA', 'SEXTA')), foreign key(bolsista_id) references TB_BOLSISTA(id), foreign key(faixa_horaria_id) references TB_FAIXA_HORARIA(id));
CREATE TABLE TB_FAIXA_HORARIA (id integer primary key autoincrement, turno text check (turno in('MANHA', 'TARDE', 'NOITE')), faixa_horario text check(faixa_horario in('A_PRIMEIRO_HORARIO', 'B_SEGUNDO_HORARIO', 'C_TERCEIRO_HORARIO', 'D_QUARTO_HORARIO', 'E_QUINTO_HORARIO')));
sqlite> ^S
```

**Obs:** Pelo o que encontrei, o sqlite não aceita diretamente o tipo ENUM, e quando deu certo colocar, havia um limite na quantidade de valores que podiam ser postos entre os parêntes (2), sendo que a maioria das informações que demandam esse tipo de dado tinham mais de duas possibilidades. Tendo isso em vista, no lugar do ENUM, foi utilizado o "text check" que funciona de forma similar.