# Artefact 6:

This page contains a list of queries developed for release 2. These Queries went alongside Angular app and Node.JS development also done by me.

The queries include:

* Teacher Registration Queries
  + Check Email
  + Insert Password
  + Insert Teacher
  + Insert Experience
  + Insert Languages
* Teacher/Student View/Cancel Lesson Queries
  + Get Lessons
  + Get Lesson
  + Finish Lesson
  + Get Info for emailing
* User Update Details Queries
  + Update User Details
  + Compare passwords
  + Create new password
  + Update Experience
  + Update Languages
* Update Instrument Details

## Teacher Registration Queries:

### Check Email

var checkEmail = {

text: "SELECT 1 FROM music\_school.teachers WHERE email = $1",

name: "check-teacher-email",

values: [teacher.email]

};

### Insert Password

var passwordCols = "salt, password"

var newTeacherPasswordQuery = {

text: "INSERT INTO music\_school.passwords("+passwordCols+") VALUES("

+"$1,$2"

+")",

name: "create-new-teacher-password",

values: [

n

, hashedPassword

]

};

### Insert Teacher

var teacherCols = "first\_name, middle\_name, last\_name, dob, address, phone\_no, email, password\_id, is\_terminated, date\_employed, staff\_description, gender"

var newTeacherQuery = {

text: "INSERT INTO music\_school.teachers("+teacherCols+") VALUES("

+"$1,$2,$3,"

+"to\_date($4, 'DD MM YYYY'),$5,$6,$7,"

+"(SELECT MAX(id) FROM music\_school.passwords),"

+"FALSE,"

+"now(),$8,$9"

+")",

name: "create-new-teacher",

values: [

teacher.firstName

, teacher.middleName

, teacher.lastName

, teacher.birthday

, teacher.address

, teacher.phoneNumber

, teacher.email

, teacher.description.escapeHtml()

, teacher.gender

]

};

### Insert Teacher Experience

var experienceColumns = "teacher\_id, inst\_type\_id, grade"

var experienceText = "INSERT INTO music\_school.teacher\_experience("+experienceColumns+") VALUES";

var experienceValues = [];

var counter = 1;

for(var i = 0; i < teacher.instrumentTypeIds.length; i++) {

var instrumentInsert = '';

if(i != 0) {

instrumentInsert += ',';

}

instrumentInsert += "((SELECT MAX(ID) FROM music\_school.teachers), $"+ counter++ +", $"+ counter++ +")";

experienceValues.push(teacher.instrumentTypeIds[i]);

experienceValues.push(teacher.instrumentTypeGrades[i]);

experienceText += instrumentInsert;

}

### Insert Teacher Languages

var languageColumns = "teacher\_id, language\_id"

var languagesText = "INSERT INTO music\_school.teacher\_languages("+languageColumns+") VALUES";

var languageValues = [];

counter = 1;

for(var i = 0; i < teacher.languages.length; i++) {

var languageText = '';

if(i != 0) {

languageText += ',';

}

languageText += "((SELECT MAX(ID) FROM music\_school.teachers), $"+ counter++ +")";

languageValues.push(teacher.languages[i]);

languagesText += languageText;

}

## Teacher View Lessons Queries as well as Student View Lessons Queries:

### Get Lessons Details

var getLessonsQuery = {

text: "SELECT l.id, "

+"s.first\_name, "

+"s.last\_name, "

+"it.name as instrument\_type, "

+"se.grade as student\_grade, "

+"l.lesson\_day, "

+"l.lesson\_start\_time as start\_time, "

+"l.lesson\_end\_time as end\_time "

+"FROM music\_school.lessons l, "

+"music\_school.students s, "

+"music\_school.instrument\_types it, "

+"music\_school.student\_experience se "

+"WHERE l.student\_id = s.id "

+"AND l.student\_id = se.student\_id "

+"AND l.inst\_type\_id = se.inst\_type\_id "

+"AND l.inst\_type\_id = it.id "

+"AND l.teacher\_id = $1 "

+"AND l.request\_status\_id = 2 "

+"ORDER BY l.lesson\_start\_time ASC",

name: "get-teacher-lessons",

values: [

teacherID

]

}

### Get Lesson Details

var getLessonQuery = {

text: "SELECT l.id, "

+"s.first\_name, "

+"s.last\_name, "

+"it.name as instrument\_type, "

+"se.grade as student\_grade, "

+"lang.language, "

+"l.lesson\_day, "

+"l.lesson\_start\_time as start\_time, "

+"l.lesson\_end\_time as end\_time, "

+"l.lesson\_year as year, "

+"l.lesson\_term as term "

+"FROM music\_school.lessons l, "

+"music\_school.students s, "

+"music\_school.instrument\_types it, "

+"music\_school.languages lang, "

+"music\_school.student\_experience se "

+"WHERE l.student\_id = s.id "

+"AND l.language\_id = lang.id "

+"AND l.student\_id = se.student\_id "

+"AND l.inst\_type\_id = it.id "

+"AND l.teacher\_id = $1 "

+"AND l.id = $2 "

+"AND l.request\_status\_id = 2 "

+"ORDER BY l.lesson\_start\_time ASC",

name: "get-teacher-lesson",

values: [

teacherId,

lessonId

]

}

### Finish Lesson

var finishLessonQuery = {

text: "UPDATE music\_school.lessons "

+"SET request\_status\_id = 7 "

+"WHERE id = $1 "

+"AND teacher\_id = $2",

name: "finish-lesson",

values: [

cancelDetails.lessonId,

cancelDetails.teacherId

]

};

### Get Info for emailing

var getInfoQuery = {

text: "SELECT s.first\_name || ' ' || s.last\_name as studentname, "

+"s.email as studentEmail, "

+"t.first\_name || ' ' || t.last\_name as teachername, "

+"t.email as teacheremail, "

+"it.name as instrument, "

+"l.lesson\_start\_time as starttime, "

+"l.lesson\_end\_time as endtime, "

+"l.lesson\_day as day "

+"FROM music\_school.lessons l, music\_school.students s, music\_school.teachers t, music\_school.instrument\_types it "

+"WHERE l.student\_id = s.id "

+"AND l.teacher\_id = t.id "

+"AND l.inst\_type\_id = it.id "

+"AND l.id = $1",

name: "get-cancelled-lesson-info",

values: [

lessonId

]

};

## User Update Details Query

### Update User Details

var counter = 1;

var updateUserQuery = "UPDATE music\_school." + user.type + "s ";

var updateUserValues = [];

Object.keys(user).forEach(function (key) {

if(key != 'id' && key != 'type') {

if(counter == 1) {

updateUserQuery += "SET "

} else {

updateUserQuery += ", "

}

if(key == 'dob') {

updateUserQuery += key+"=to\_date($"+ counter++ +",'DD/MM/YYYY') ";

} else {

updateUserQuery += key+"=$"+ counter++ +" ";

}

updateUserValues.push(user[key]);

}

});

### Compare passwords

var checkOldPasswordQuery = {

text: "SELECT p.salt, p.password from music\_school." + password.type + "s u, music\_school.passwords p "

+"WHERE u.password\_id = p.id "

+"AND u.id=$1",

name: "get-old-password",

values: [password.id]

};

### Create new password

var passwordCols = "salt, password"

var newPasswordQuery = "INSERT INTO music\_school.passwords("+passwordCols+") VALUES("+n+", "+hashedPassword+"); "

+"UPDATE music\_school."+password.type+"s SET password\_id = ("

+"SELECT MAX(id) FROM music\_school.passwords"

+") WHERE id="+password.id+";"

### Update Experience

var experienceColumns = "teacher\_id, inst\_type\_id, grade";

var experienceText = "INSERT INTO music\_school.teacher\_experience("+experienceColumns+") VALUES";

var experienceValues = [skills.id];

var counter = 2;

for(var i = 0; i < skills.instruments.length; i++) {

var instrumentInsert = '';

if(i != 0) {

instrumentInsert += ',';

}

instrumentInsert += "($1, $"+ counter++ +", $"+ counter++ +")";

var id = skills.instruments[i]

experienceValues.push(id);

experienceValues.push(skills.grades[id - 1]);

experienceText += instrumentInsert;

}

experienceText += " ON CONFLICT (teacher\_id, inst\_type\_id) DO UPDATE SET grade = EXCLUDED.grade";

### Update Languages

var languageColumns = "teacher\_id, language\_id";

var languageText = "INSERT INTO music\_school.teacher\_languages("+languageColumns+") VALUES";

var languageValues = [skills.id];

var counter = 2;

for(var i = 0; i < skills.languages.length; i++) {

var languageInsert = '';

if(i != 0) {

languageInsert += ',';

}

languageInsert += "($1, $"+ counter++ +")";

languageValues.push(skills.languages[i]);

languageText += languageInsert;

}

languageText += " ON CONFLICT (teacher\_id, language\_id) DO NOTHING";

## Update Instrument Details Query:

var counter = 1;

var updateInstrumentQuery = "UPDATE music\_school.instruments ";

var updateInstrumentValues = [];

Object.keys(instrument).forEach(function (key) {

if(key != 'id') {

if(counter == 1) {

updateInstrumentQuery += "SET "

} else {

updateInstrumentQuery += ", "

}

if(key == 'purchase\_date') {

updateInstrumentQuery += key+"=to\_date($"+ counter++ +",'DD/MM/YYYY') ";

} else {

updateInstrumentQuery += key+"=$"+ counter++ +" ";

}

updateInstrumentValues.push(instrument[key]);

}

});

updateInstrumentQuery += "WHERE id=$" + counter++;

updateInstrumentValues.push(instrument.id);