**Diplomatura en Programación**

**Cohorte Nº1: marzo del 2023.**

**Proyecto: carga de notas para docentes.**

**Emmanuel Matías Di Benedetto.**

**Profesores: Diego Caprioli y Carlos Cadabeira.**

**Tabla de contenido**

**Idea de proyecto**

El programa tiene como funcionalidad que el docente pueda cargar a la base de datos sus cursos y, dentro de esos cursos, cargar a los alumnos y a las notas de cada uno de ellos.

Habrá tres tablas. Una de ellas, llamada docentes, tendrá los campos con los datos generales que suele tener todo usuario: nombre, contraseña, rol y docente\_id. El campo docente\_id es clave para que los cursos que se muestren sean los correspondientes a ese docente\_id. El rol administrador podrá eliminar y modificar los campos de los otros docentes.

La segunda tabla se llama cursos, y tendrá los siguientes campos: curso\_id, docente\_id, cantidad de notas, nombre del colegio y nombre del curso. Al crear un curso nuevo, el docente ingreserá los valores de estos campos.

La última tabla es alumnos. Y tiene las columnas docente\_id, curso\_id, nombre del alumno, alumno\_id y notas. Gracias a las primeras dos columnas, el alumno puede identificarse con el curso y el docente correcto. Así, a la hora de listar los datos, es necesario que se muestren aquellos correspondientes al curso\_id y docente\_id que corresponda.

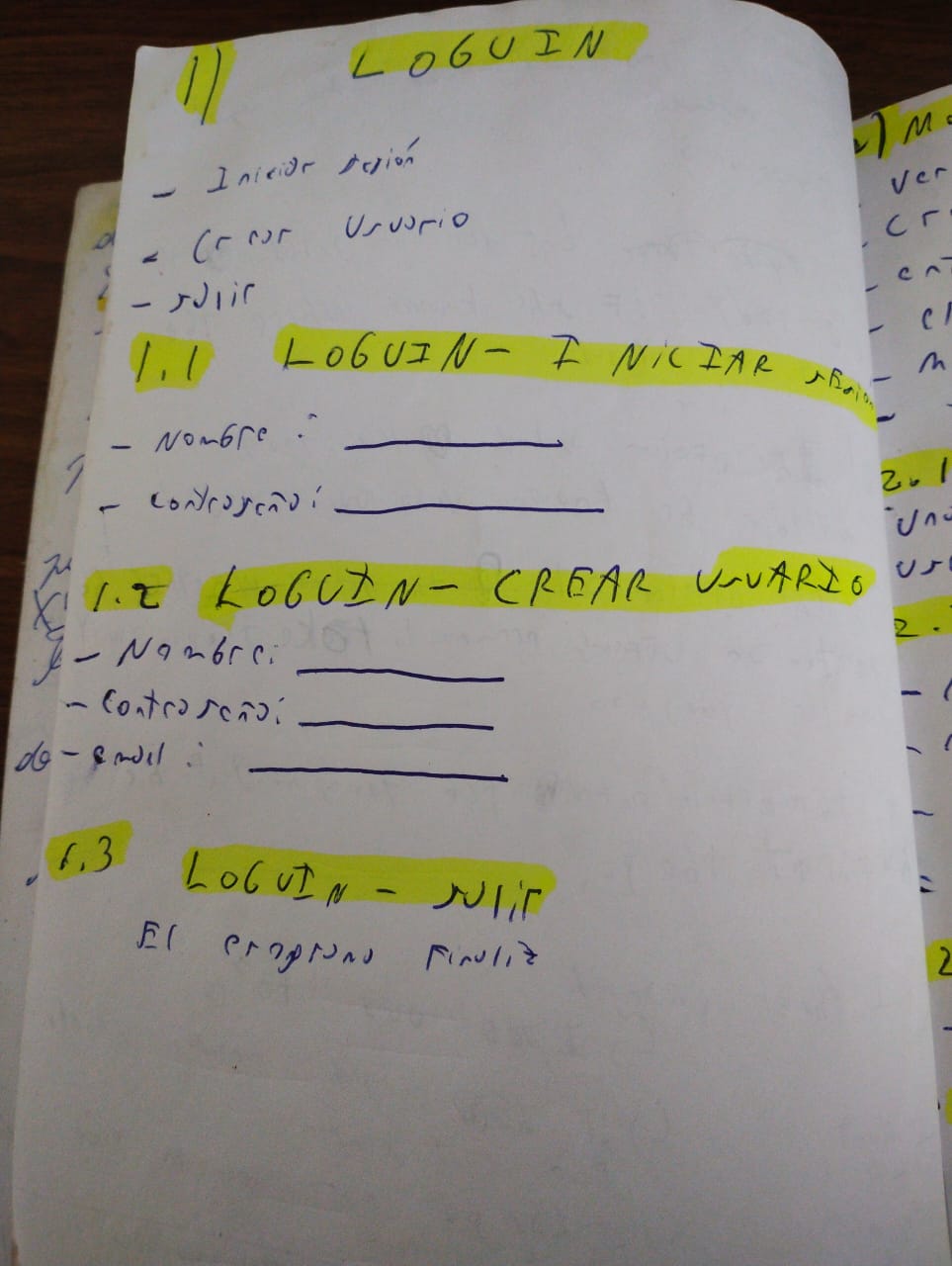
El docente-usuario será capaz de eliminar los registros, modificarlos y de agregar uno nuevo. Así, podrá crear un nuevo curso, eliminar un curso y modificarlo (cambiarle el nombre, por ejemplo). También será capaz de agregar nuevos alumnos, así como de aregarles una nota, modificarla y eliminarla. También podrá modificar los nombres de los alumnos.

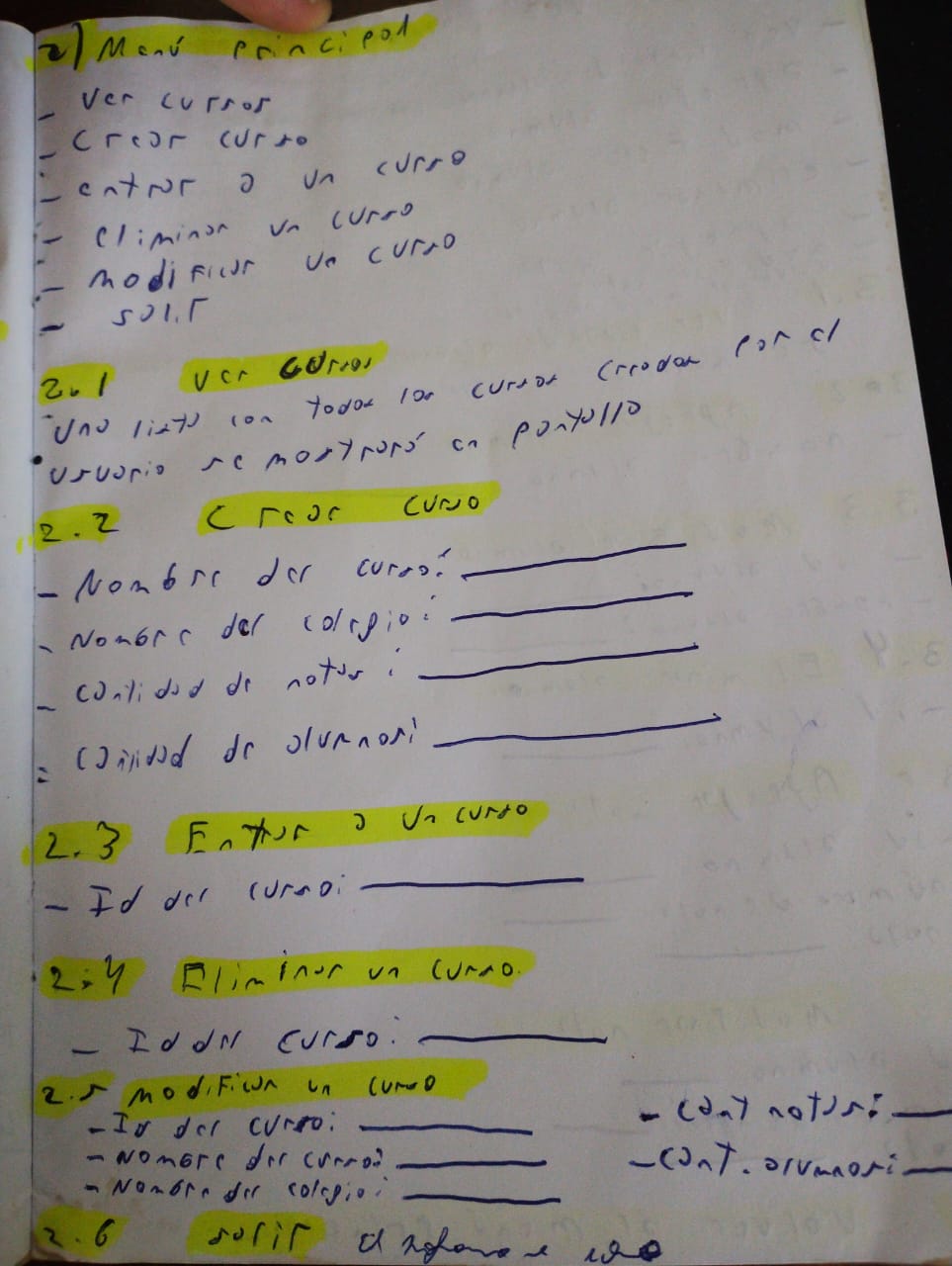
Al abrir el programa, aparecerán dos opciones: crear un usuario o iniciar sesión. Al iniciar sesión, se buscará en la base de datos una coincidencia entre los datos ingresados por el usuario y los que allí existen. De haber coincidencia, ingresa al menú principal; sino, aparecerá un mensaje de error.

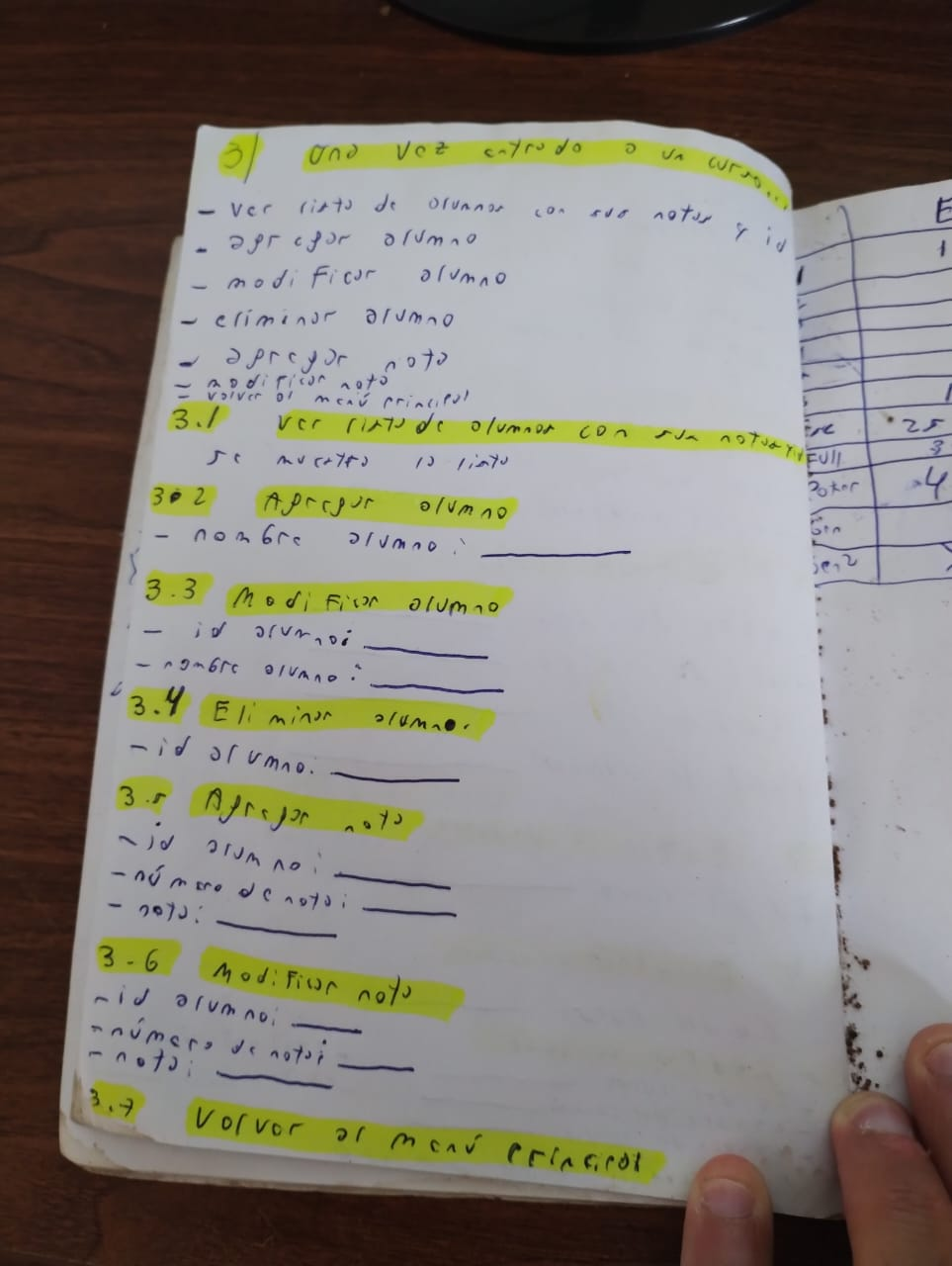
Una vez en el menú principal, aparecerán las siguientes opciones: ver cursos, agregar curso, ingresar a curso y cerrar sesión. Si selecciona la primera opción, se listarán todos los cursos que existen. En “agregar curso”, el usuario puede crear un nuevo curso, completando los campos necesarios. Para ingresar a un curso, debe escribir su nombre o su id y ahí podrá ver todos los alumnos que hay dentro, junto con sus notas.

Una vez en el curso, aparecen las siguientes opciones: ver listado, agregar alumno, agregar nota, borrar alumno y modificar nota. Cuando el usuario sleeccione la opción “ver listado”, aparecerán todos los alumnos correspondientes a ese curso\_id y a ese docente\_id con todos sus campos. Al agregar un alumno, el docente, llenará los campos requeridos.

**MAPA DE PANTALLA**







**ESTRUCTURA DE LA BASE DE DATOS**

crearTabla = "CREATE TABLE IF NOT EXISTS docentes(docente\_id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, nombre VARCHAR NOT NULL, contrasenia VARCHAR NOT NULL, email VARCHAR NOT NULL);"

uardarDocente = "INSERT INTO docentes(nombre, contrasenia, email) VALUES (?, ?, ?);"

id = "SELECT docente\_id FROM docentes WHERE nombre = ? AND contrasenia = ?;"

registros\_cursos = "SELECT \* FROM cursos WHERE docente\_id = ?;"

crearTablaCurso = "CREATE TABLE IF NOT EXISTS cursos(curso\_id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, cant\_notas INTEGER NOT NULL, nombre\_curso VARCHAR NOT NULL, escuela VARCHAR NOT NULL,  docente\_id INTEGER NOT NULL);"

guardarCurso = "INSERT INTO cursos(cant\_notas, nombre\_curso, escuela, docente\_id) VALUES (?, ?, ?, ?);"

registro\_curso\_seleccionado = "SELECT \* FROM cursos WHERE docente\_id = ? AND curso\_id = ?;"

registros\_alumnos = "SELECT \* FROM alumnos WHERE docente\_id = ? AND curso\_id = ?"

crearTablaAlumnos = "CREATE TABLE IF NOT EXISTS alumnos(alumno\_id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, nombre VARCHAR NOT NULL, notas VARCHAR NULL, concepto VARCHAR NULL, docente\_id INTEGER NOT NULL, curso\_id INTEGER NOT NULL);"

guardarAlumno = "INSERT INTO alumnos(nombre, docente\_id, curso\_id) VALUES (?, ?, ?);"

eliminarAlumno = "DELETE FROM alumnos WHERE alumno\_id = ?;"

editarAlumno = "UPDATE alumnos SET nombre = ? WHERE alumno\_id = ?"

registrosDelAlumnoAEvaluar = "SELECT \* FROM alumnos WHERE alumno\_id = ?;"

ingresarNota = "UPDATE alumnos SET notas = ? WHERE alumno\_id = ?;"

eliminarCurso = "DELETE FROM cursos WHERE curso\_id = ?;"

eliminar\_alumnos\_del\_curso = "DELETE FROM alumnos WHERE curso\_id = ?;"

editarCurso = "UPDATE cursos SET cant\_notas = ?, nombre\_curso = ?, escuela = ? WHERE curso\_id = ?;"

**MANUAL DE USUARIO**

El programa le permite al docente crear sus cursos y, en cada uno de ellos, agregar los alumnos y las notas, así como hacer las respectivas modificaciones.

**Consideraciones técnicas:**

Tenés que tener instalado Python. El archivo, llamado proyecto.py, debe estar guardado en la misma carpeta en la que está el archivo mp3 llamado LOTR.mp3.

Para ejecutar el programa, desde la consola del Visual Studio Code o en la de Windows, te dirigís a la carpeta donde guardaste los archivos. Una vez allí, escribís el código python proyecto.py

**Crear un nuevo usuario:**