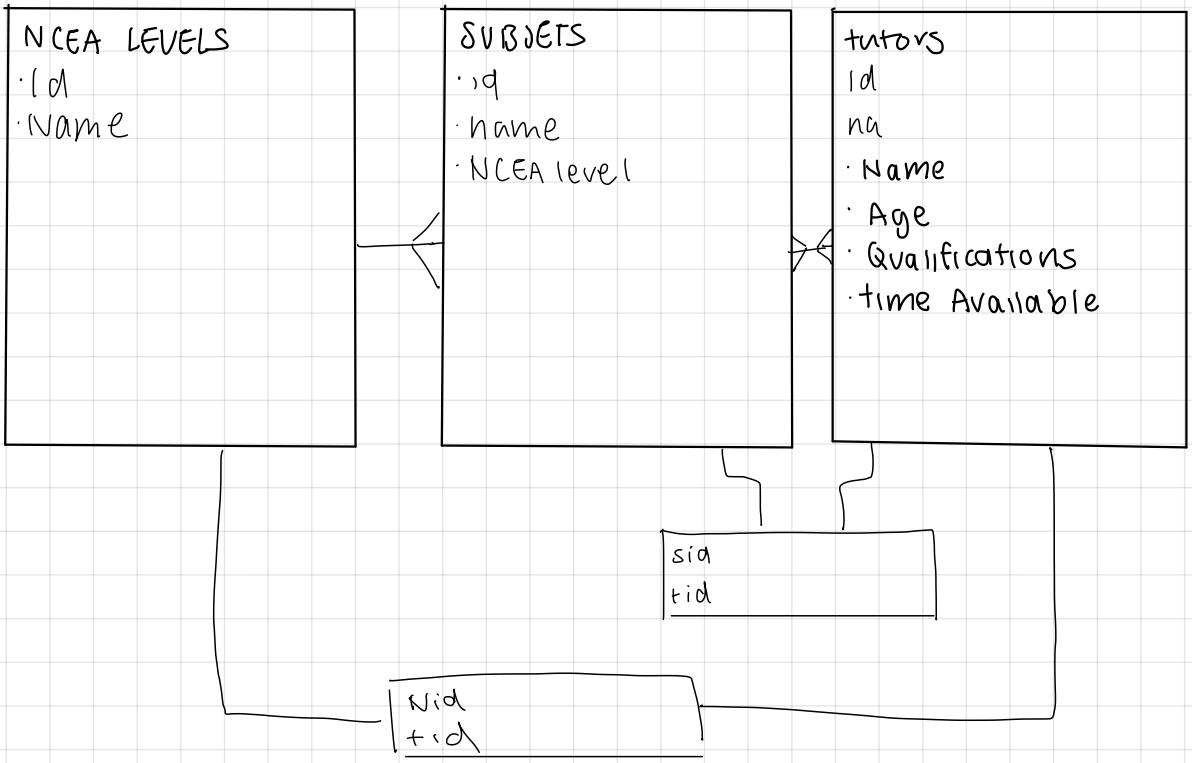
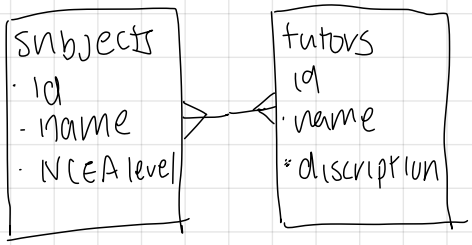
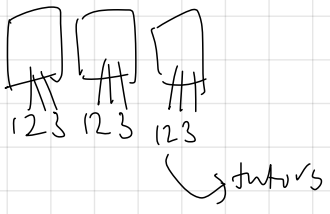
Database notes and design

(NCEA levels } one
subjects } many
tutors — Name
Age
Available times







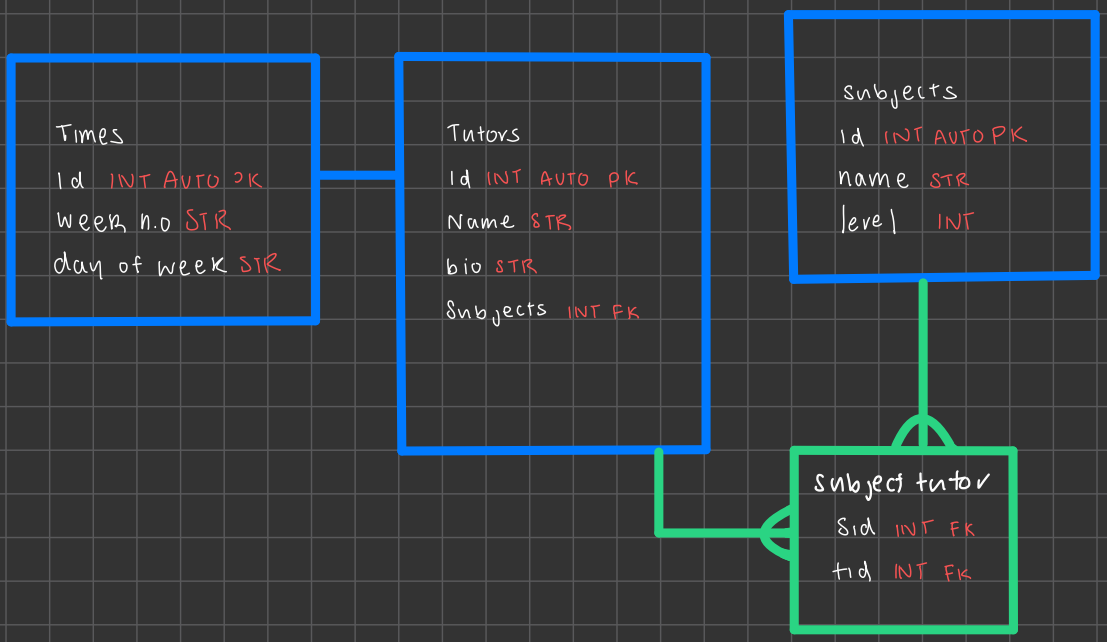
This was the initial rough ER diagram I drew for this project. I soon realised that it was impractical and the box for NCEA levels was unnecessary.

What is your project about, who is it for, what does it do.

Tutoring Services: A website where highschool students who study any NCEA levels from 1-3 can find student tutors within the same school that can help them with their subjects.
This website will allow students and tutors to plan a time that works for both of them and will also allow students to have control over who their tutor is.

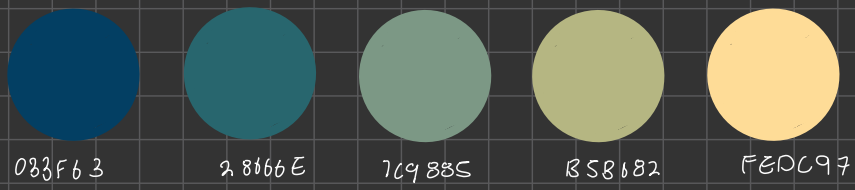
Database notes & design:

1 tutor = multiple subjects & year levels.*(1, 2, 3 or scholarship)
subject = 4 different year levels & multiple tutors (some are only in some year levels)
year levels = multiple tutors & subjects.



Website layout and Design

General colour scheme

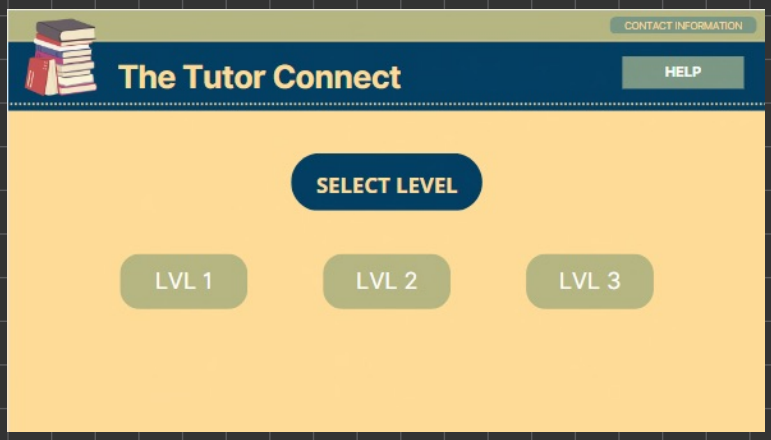


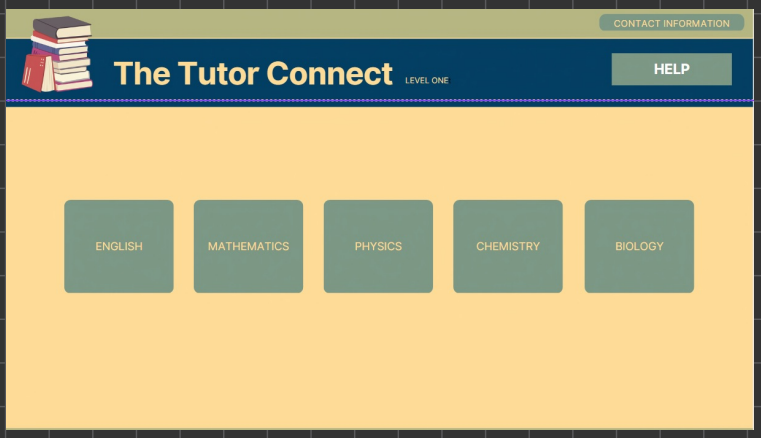
033F63 2866E 7C9885 B58182 FEDC97

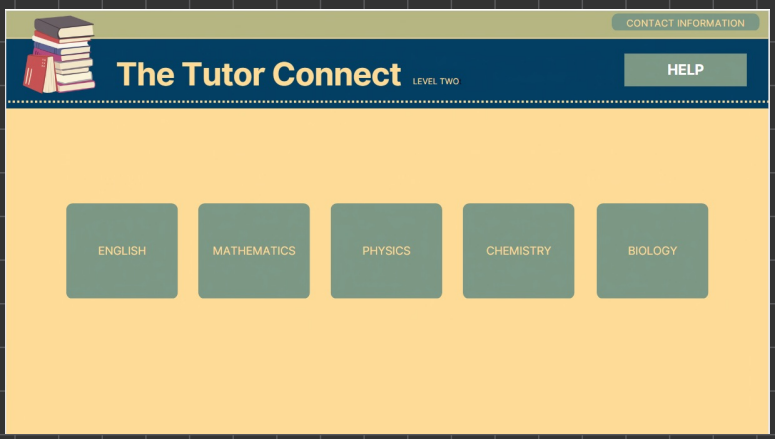
Fonts

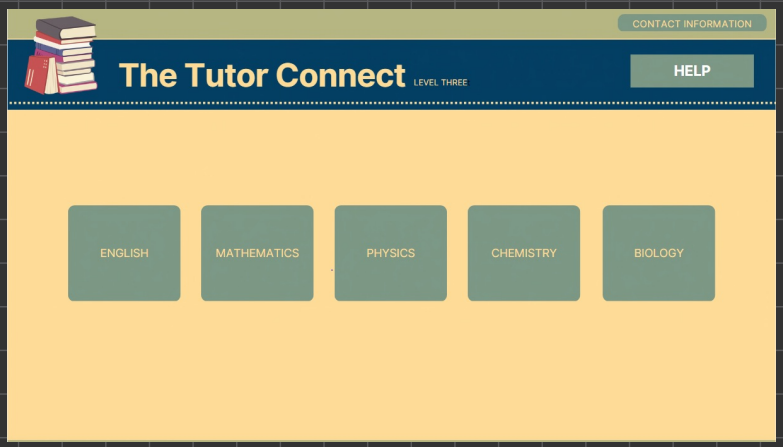
Inter

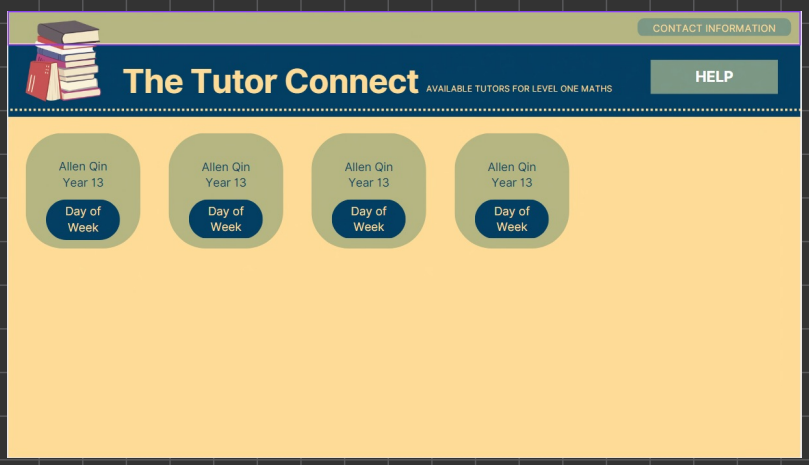
Website layout

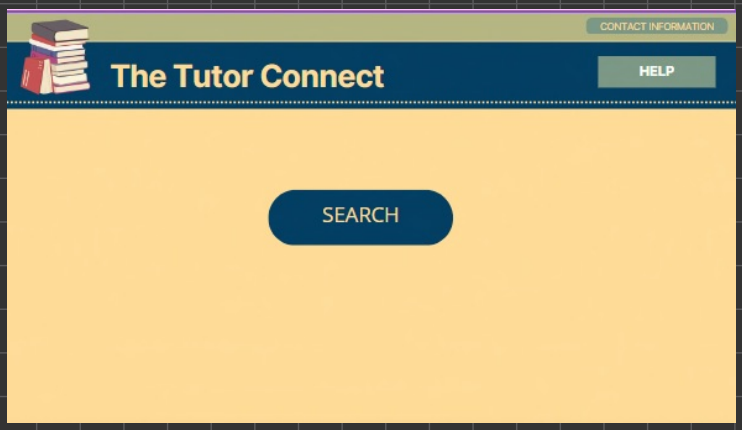


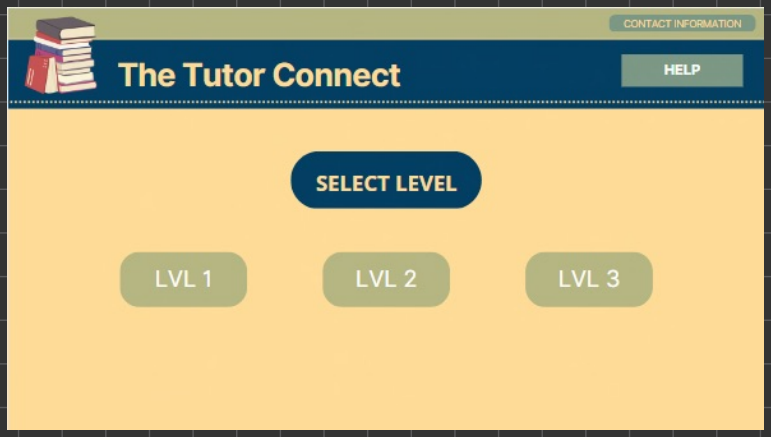






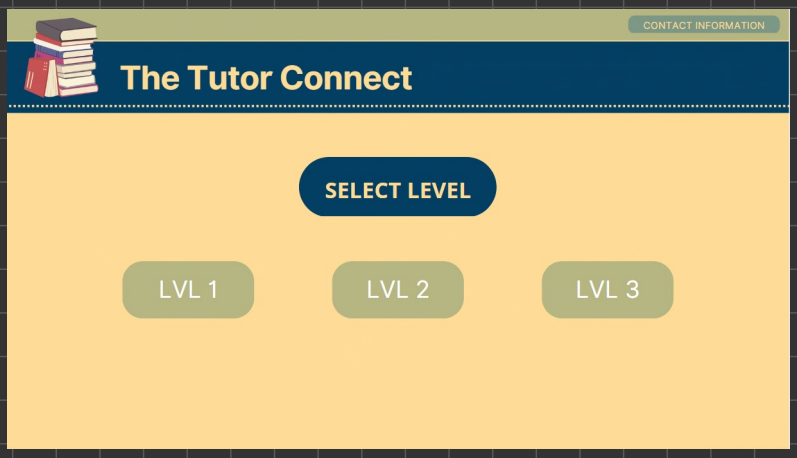




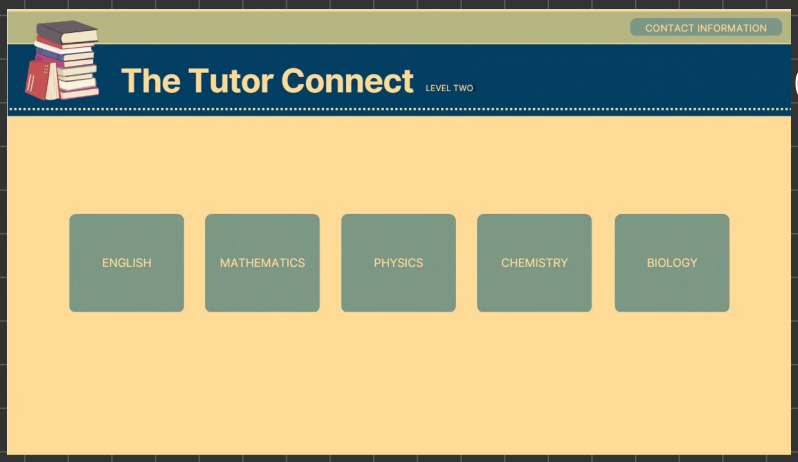


I was debating between these two home pages and after consideration, decided to go with the second option as I thought it would be easier to code.

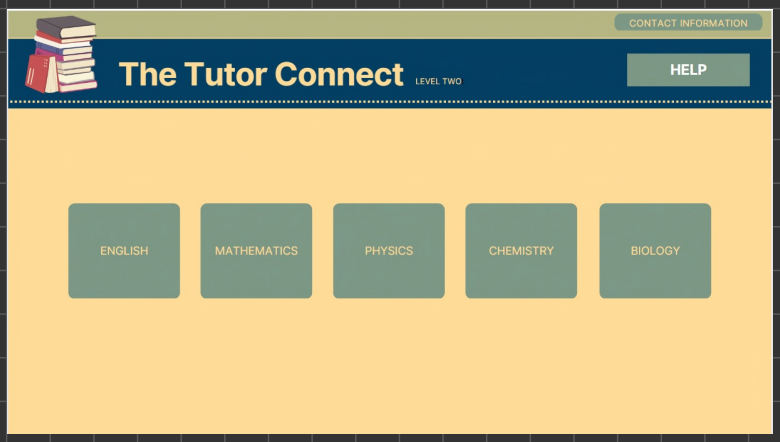
Routes / function signatures for each page



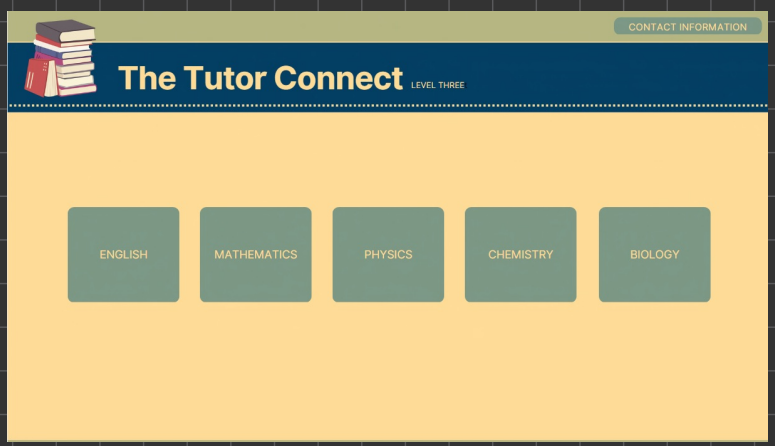

```
@app.route('/')
def home():
```




```
@app.route('/Levelone')
def Levelone():
SELECT name FROM subjects WHERE
Level = 1
```




```
@app.route('/Leveltwo')
def Leveltwo():
SELECT name FROM subjects WHERE
Level = 2
```

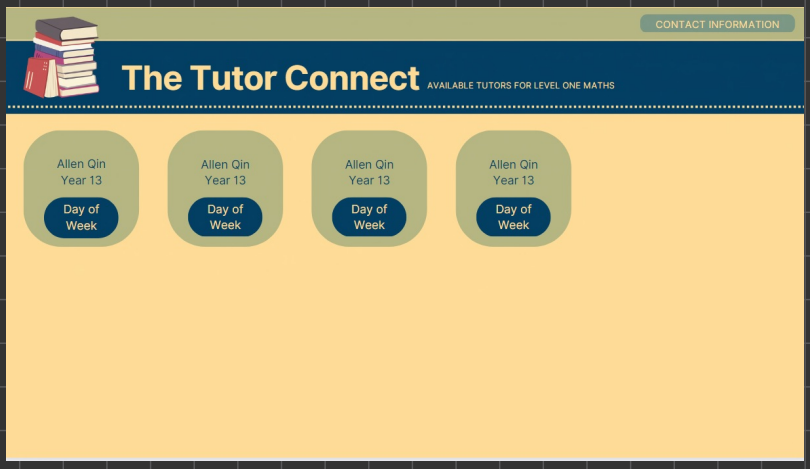



```
@app.route('/Levelthree')
def Levelthree():
SELECT name FROM subjects WHERE
Level = 3
```

This was repetative and inefficient therefore I can change it into

```
SELECT name FROM
subject WHERE
Level = Level
```

```
@app.route('/level/<int:level>')
def show_level(level):
    if level < 1 or level > 3
        abort(404)
```




```
def Tutors():
    SELECT name, bio, subjects FROM
    Tutors WHERE subject = 1
```

```
@app.route('/Tutors/<str:Subject>')
```

```
SELECT name, bio, subjects FROM tutor WHERE
id IN(SELECT tid FROM subjecttutor
WHERE sid = (SELECT id FROM subject
WHERE UPPER(name) = "Maths1 "))
```

this was my first attempt

```
SELECT week.no, day of week, FROM
times WHERE id IN [SELECT tid FROM
tutors WHERE UPPER (name) = (Allen Qin)
```

JUSTIFICATION

I deliberately chose these contrasting colours as my target audience is high school students looking for a tutor as efficiently as possible. The contrasting colours allow for the student to quickly identify the buttons. These colours have very low saturation which makes it easy on the eyes and helps the student quickly find what they are looking for.

After showing Allen Qin, a student who I know that has an interest in tutoring services, I received his feedback and he states “That it is pleasing to the eye yet there is a lot of empty space that could be utilised.”. He also notes that the tutoring services should help with other subjects other than the main five.

The scientific research behind my design choices can be found in Nielsen’s Heuristics. User control and freedom is shown in my design by the large heading acting as a home button and allowing the user to move freely back to the original page. An aesthetic and minimalist design is also achieved by the use of minimal words and lack of buttons. It is also achieved through only using a set number of colours for every element.

This website design is based on the accepted best practice of placing a title in the top left corner along with a banner/bar over the top. Contact information is also placed within the top banner and the contents of the page below this banner. This design is the same on all the pages and only the contents change