# CM3010 midterm report

The project is a comprehensive mental health dashboard designed to analyze and visualize the mental health status of university students, focusing on various factors such as engagement, stress levels, sleep quality and the impact of mental health support. Utilizing a MySQL database, the application aggregates data from multiple sources, including student demographics, mental health assessments and lifestyle habits. Through interactive charts and graphs powered by Chart.js, users can explore correlations between academic performance (CGPA), study habits and mental health issues like depression and anxiety. The dashboard also provides insights into the prevalence of mental health problems across different genders and academic years, as well as the effectiveness of specialist treatment, thereby serving as a valuable tool for educators, mental health professionals and policymakers to better understand and address the mental health needs of students.

## Stage 1: Find and critique a dataset

### 1. Source of Open Data

Dataset Source: University Students' Mental Health

### 2. Dataset Assessment

**Quality:** The dataset is of high quality, with a significant number of records (over 1,000 entries) and a variety of attributes that cover multiple aspects of student life. This dataset contains 16 columns and 1,001 rows.

**Dataset Structure**

- **Timestamp:** Records the date the data was collected. Example: "13/7/2020".
- **Gender:** Indicates the gender of the student. Example: "Female".
- **Age:** Represents the age of the student. Example: "24".
- **Course:** Specifies the course or field of study the student is enrolled in. Examples: "Biotechnology", "Engineering".
- **YearOfStudy:** Indicates the year of study the student is in. Examples: "Year 3", "year 4".
- **CGPA:** Cumulative Grade Point Average of the student. Example: "2.38".
- **Depression:** Binary indicator (0 or 1) of whether the student is experiencing depression. Example: "1" (Yes), "0" (No).
- **Anxiety:** Binary indicator (0 or 1) of whether the student is experiencing anxiety. Example: "0" (No), "1" (Yes).

- **PanicAttack:** Binary indicator (0 or 1) of whether the student has had a panic attack. Example: "0" (No), "1" (Yes).
- **SpecialistTreatment:** Binary indicator (0 or 1) of whether the student has sought specialist treatment for mental health issues. Example: "0" (No).
- **SymptomFrequency_Last7Days:** Frequency of mental health symptoms experienced in the last 7 days, on a scale. Example: "5".
- **HasMentalHealthSupport:** Binary indicator (0 or 1) of whether the student has mental health support. Example: "0" (No), "1" (Yes).
- **SleepQuality:** Sleep quality rating, on a scale. Example: "4".
- **StudyStressLevel:** Study stress level rating, on a scale. Example: "5".
- **StudyHoursPerWeek:** Number of study hours per week. Example: "8".
- **AcademicEngagement:** Academic engagement level, on a scale. Example: "2".

This structure allows for a comprehensive analysis of how various factors such as demographics, academics and mental health interrelate and influence each other. Since this dataset is comprehensive and sufficient for building a database for this midterm project, I will not be utilizing other datasets. It ensures consistency in data collection methods and relevance to the mental health context. Introducing additional datasets may lead to inconsistencies, complicate analysis and dilute the focus of the project. Therefore, I will not utilize other datasets, ensuring a coherent and focused approach to achieve meaningful insights.

**Level of Detail:** The dataset provides a detailed breakdown of mental health issues, academic performance, and lifestyle factors. Each entry includes demographic information, which allows for nuanced analysis across different groups (e.g., gender, year of study). The data is detailed enough, yet not overly complicated, for me to create a database for Students Mental Health.

**Documentation:** This dataset was gathered and collected by a survey conducted by Google forms from Malaysia University students in order to examine their current academic situation and mental health. The owner of this dataset is **Jia Jun Chen**.

**Interrelation:** The dataset allows for interrelation between various attributes, such as the correlation between CGPA and mental health issues, or the impact of sleep quality on academic engagement. This relational aspect is crucial for conducting in-depth analyses.

**Use**: The dataset can be used for various analyses, including statistical modeling, trend analysis, and visualization of mental health issues among students. It can also serve as a basis for developing applications aimed at improving student well-being.

On kaggle, a score of 8.24 **Usability** suggests that the dataset is highly usable, meaning it is likely well-documented, easy to understand and straightforward to work

with. A high usability score indicates that a person will have fewer issues when trying to analyze or manipulate the data for the projects.

**Discoverability:** The dataset is easily discoverable on Kaggle, a well-known platform for open data. However, it may not be as easily accessible to individuals unfamiliar with the platform or those who do not actively seek out datasets like me.

**Terms of Use:** The dataset is provided under Kaggle's terms of use, which generally allow for personal and educational use. On kaggle, the dataset is licensed under **CC0: Public Domain** which means that the data is freely available for use by anyone, for any purpose, without needing permission from or giving credit to the original creator.

It was rather difficult to find a dataset that is suitable for this project. Many of them had either too much irrelevant or confusing data, too little data for the normalization process, only a single entity which formed only one table in the ERD diagram, or too many entities and tables which further complicated the dataset. There were other alternative datasets, but I chose this one as I felt it was the most complete and contained all the relevant fields needed for my analysis.

## 3. Interest in the Dataset

This dataset is particularly interesting due to the increasing awareness of mental health issues among university students, especially in the wake of the COVID-19 pandemic. Understanding the factors that contribute to mental health challenges can help educational institutions develop targeted interventions and support systems.

Some questions that could be explored using a database application include:

- **Question 1:** How does the average academic engagement and stress levels differ between students who have mental health support and those who do not?
- **Question 2:** What is the relationship between students' CGPA, study hours and sleep quality?
- **Question 3:** What is the distribution of specialist treatment among students in different courses?
- **Question 4:** How does the prevalence of mental health issues vary between male and female students for each mental health problem?
- **Question 5:** How does the prevalence of different mental health issues vary across different years of study?
- **Question 6:** How is the frequency of specialist treatment related to different levels of sleep quality for students in each course?

- **Question 7:** What is the frequency distribution of a selected mental health symptom in the last 7 days and how many reports have been made for each frequency level?

# Stage 2. Model your data

## 1. Entity-Relationship (E/R) Model

E/R model of the dataset with cardinalities:

**Entities and Relationships**

1. **Student**
    - Attributes: StudentID (PK), Gender, Age, Course, YearOfStudy, CGPA.
    - Cardinality: A student may have many mental health data and lifestyle data.
2. **MentalHealth**
    - Attributes: ID (PK), StudentID (FK), Depression, Anxiety, PanicAttack, SpecialistTreatment, SymptomFrequency_Last7Days, HasMentalHealthSupport.
    - Cardinality: Each record is associated with exactly one student (1:1).
3. **Lifestyle**
    - Attributes: LifestyleID (PK), StudentID (FK), SleepQuality, StudyStressLevel, StudyHoursPerWeek, AcademicEngagement.
    - Cardinality: Each record is associated with exactly one student (1:1).

*Justification for Subset Implementation:* The chosen entities and relationships focus on the core aspects of student demographics, mental health status, and lifestyle factors. This subset allows for comprehensive analysis while maintaining clarity and manageability in the database structure.

## 2. E/R Diagram with Cardinality

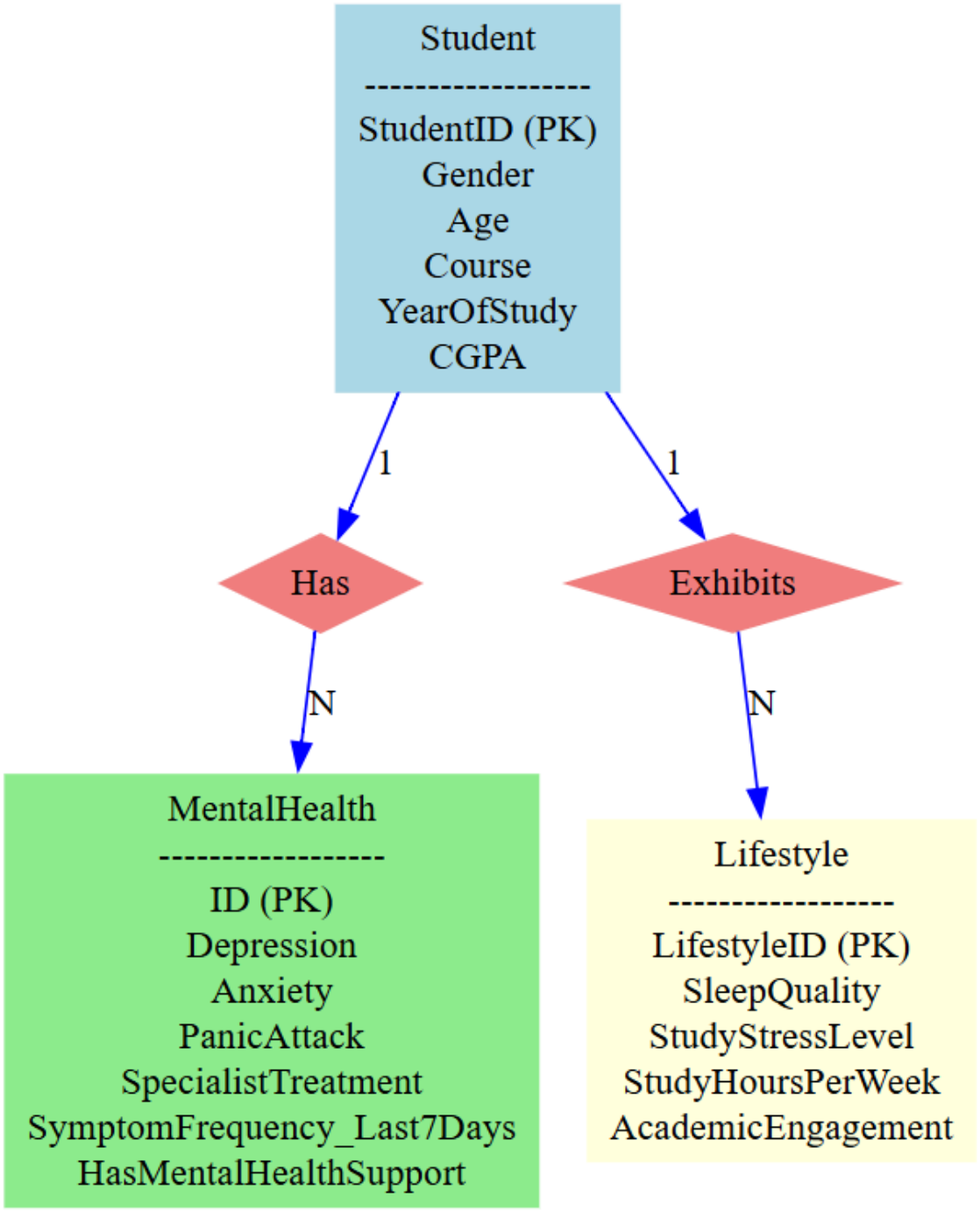



Figure 1: Diagram with Cardinality

1 student can have many mentalHealth because a single student can experience multiple mental health conditions such as Anxiety, depression or PanicAttack.

1 student can exhibit many Lifestyles entries because their lifestyle factors, such as sleep quality, study stress levels, study hours per week and academic engagement, can vary over time due to different circumstances like exams, breaks, or changes in personal routine. By allowing multiple entries, I can capture these variations and provide a more comprehensive, time-based view of each student's lifestyle.

If the provided E/R structure includes incompatible structures, such as direct many-to-many relationships or non-relational attributes (e.g., arrays), they should be adjusted. For example, a direct many-to-many relationship would need an associative table. In this dataset, no such issues exist.


Figure 2: ERD Flow Diagram

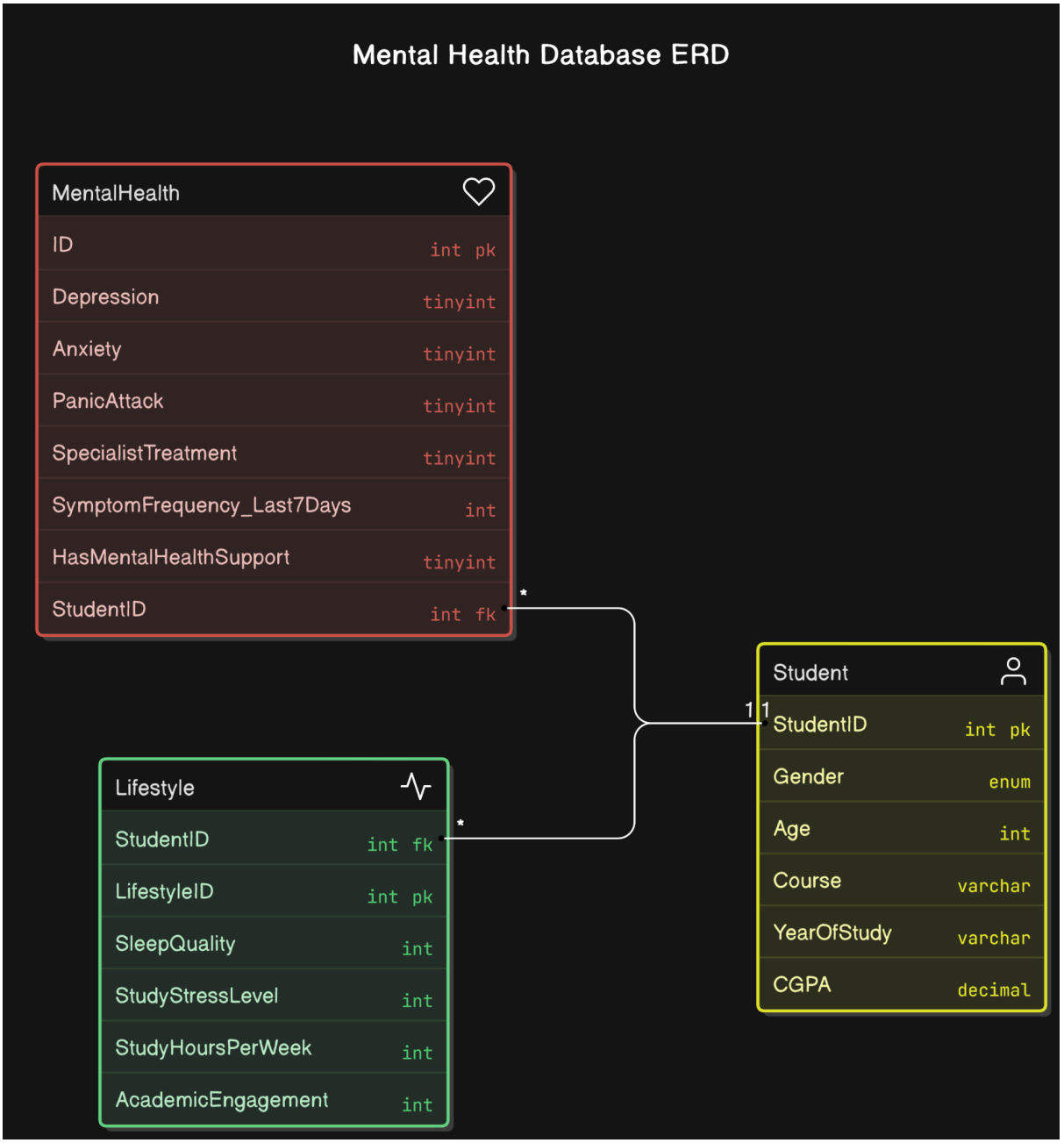



Figure 3: Relational Schema diagram

## 3. Database Tables and Fields

The following tables are created

**Table: Student**

| Field | Data Type | Description |
|---|---|---|

| | | |
|---|---|---|
| StudentID | INT (PK, Auto Increment) | Unique ID for each student |
| Gender | ENUM('Male', 'Female') | Gender of the student |
| Age | INT | Age of the student |
| Course | VARCHAR(255) | Course the student is enrolled in |
| YearOfStudy | VARCHAR(50) | Year of study |
| CGPA | DECIMAL(4,2) | Cumulative GPA |

**Table: MentalHealth**

| Field | Data Type | Description |
|---|---|---|
| ID | INT (PK, Auto Increment) | Unique mental health record |
| StudentID | INT (FK) | Reference to Student(StudentID) |
| Depression | TINYINT | Indicates depression (0 or 1) |
| Anxiety | TINYINT | Indicates anxiety (0 or 1) |
| PanicAttack | TINYINT | Indicates panic attack (0 or 1) |

| | | |
|---|---|---|
| SpecialistTreatment | TINYINT | Indicates specialist treatment (0 or 1) |
| SymptomFrequency_Last7Days | INT | Frequency of symptoms in the last 7 days |
| HasMentalHealthSupport | TINYINT | Whether the student has mental health support (0 or 1) |

**Table: Lifestyle**

| Field | Data Type | Description |
|---|---|---|
| LifestyleID | INT (PK, Auto Increment) | Unique lifestyle record |
| StudentID | INT (FK) | Reference to Student(StudentID) |
| SleepQuality | INT | Quality of sleep (scale: 1-5) |
| StudyStressLevel | INT | Level of stress from study |
| StudyHoursPerWeek | INT | Number of study hours per week |
| AcademicEngagement | INT | Academic engagement score |

## Normalization Evaluation

### 1NF (First Normal Form):

- All tables conform to 1NF:
    - Each table has a unique identifier (primary key).
    - Each column contains atomic values (no lists, arrays, or nested data).

**2NF (Second Normal Form):**

- All non-key attributes are fully functionally dependent on the primary key:
    - In the MentalHealth and Lifestyle tables, attributes are dependent only on the StudentID.
    - No partial dependency exists.

**3NF (Third Normal Form):**

- There are no transitive dependencies:
    - No attribute is dependent on another non-key attribute.
    - For example, HasMentalHealthSupport depends only on the primary key ID.

**BCNF and Beyond:**

- The database is not in Boyce-Codd Normal Form (BCNF) because there are no candidate keys that violate the BCNF conditions. However, the current structure does not require further normalization as it maintains data integrity and efficiency for the intended queries and operations.

- The database is also not in Fourth Normal Form (4NF) as there are no multi-valued dependencies present. The current design is sufficient for the analysis and reporting needs of the mental health data.

*Justification for Not Normalizing Further:* The current structure effectively supports the required queries and analyses without introducing unnecessary complexity. Further normalization could lead to performance issues due to increased joins and complexity in data retrieval. Thus, the design strikes a balance between normalization and practical usability.

**Database and Table creation:**

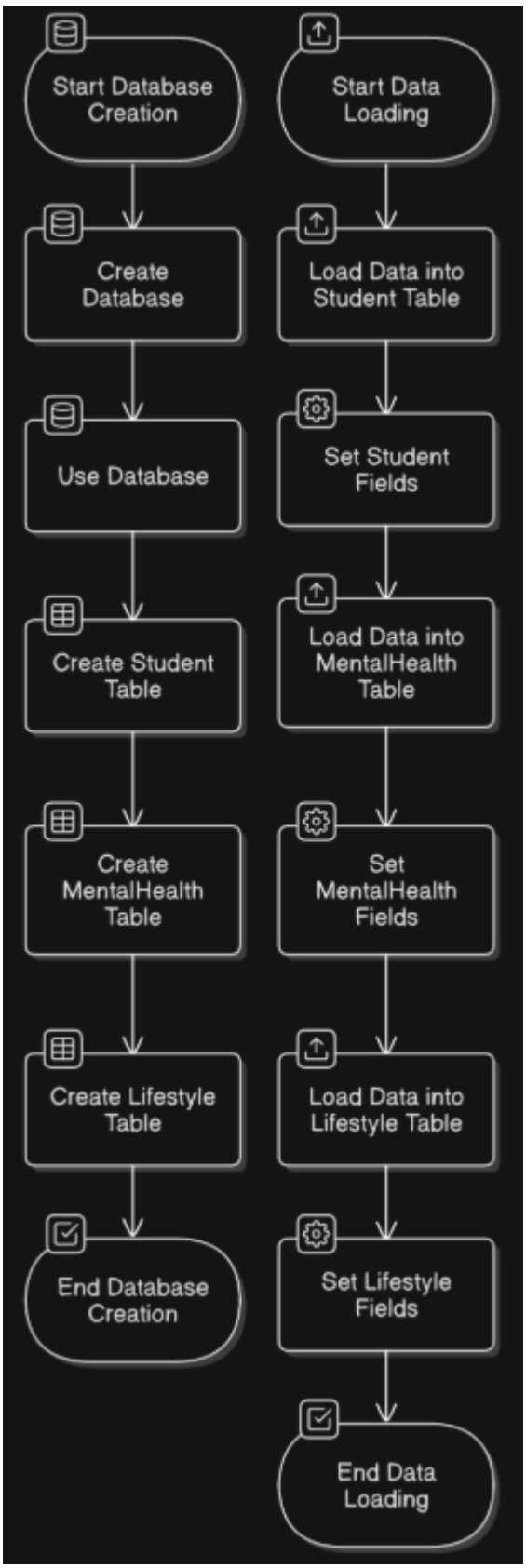



Figure 4: Database & Table Setup

## Stage 3. Create the database

### 1. Build the database structure in MySQL

To create the database structure in MySQL in the lab environment., the following CREATE commands are used and stored in a file named setup.sql. All the commands are then typed in the terminal:

setup.sql

```sql
-- Create the database
CREATE DATABASE mental_health;

USE mental_health; -- Use the database

-- Create the Student table
CREATE TABLE Student (
    StudentID INT AUTO_INCREMENT PRIMARY KEY,
    Gender ENUM('Male', 'Female') NOT NULL,
    Age INT NOT NULL,
    Course VARCHAR(255) NOT NULL,
    YearOfStudy VARCHAR(50) NOT NULL,
    CGPA DECIMAL(4, 2) NOT NULL
);

-- Recreate the MentalHealth table with HasMentalHealthSupport
CREATE TABLE MentalHealth (
    ID INT AUTO_INCREMENT PRIMARY KEY,
    StudentID INT NOT NULL,
    Depression TINYINT NOT NULL,
    Anxiety TINYINT NOT NULL,
    PanicAttack TINYINT NOT NULL,
    SpecialistTreatment TINYINT NOT NULL,
    SymptomFrequency_Last7Days INT NOT NULL,
    HasMentalHealthSupport TINYINT NOT NULL,
    FOREIGN KEY (StudentID) REFERENCES Student(StudentID) ON DELETE CASCADE
);

-- Create the Lifestyle table
CREATE TABLE Lifestyle (
    LifestyleID INT AUTO_INCREMENT PRIMARY KEY,
    StudentID INT NOT NULL,
    SleepQuality INT NOT NULL,
    StudyStressLevel INT NOT NULL,
    StudyHoursPerWeek INT NOT NULL,
    AcademicEngagement INT NOT NULL,
    FOREIGN KEY (StudentID) REFERENCES Student(StudentID) ON DELETE CASCADE
);
```

Figure 5: CREATE Commands

### 2. Enter Instance Data

To insert instance data into the database, the dataset is loaded using the LOAD DATA LOCAL command. The dataset mentalhealth_dataset.csv, is located in a subfolder named data. In the lab environment, loading local data was initially denied, so the command SET GLOBAL local_infile=ON; was used. After that, the following command was run to load the data: mysql --local-infile=1 -u root mental_health. The instance data was then successfully loaded into the respective tables:

setup.sql

```
-- Load data into the Student table
LOAD DATA LOCAL INFILE 'data/mentalhealth_dataset.csv'
INTO TABLE Student
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
IGNORE 1 ROWS
(@Timestamp, Gender, Age, Course, YearOfStudy, CGPA, @Depression, @Anxiety, @PanicAttack, @Specialis
SET
    Gender = TRIM(Gender),
    Age = CAST(Age AS UNSIGNED),
    Course = TRIM(Course),
    YearOfStudy = TRIM(YearOfStudy),
    CGPA = CAST(CGPA AS DECIMAL(4,2));

-- Load data into the MentalHealth table
LOAD DATA LOCAL INFILE 'data/mentalhealth_dataset.csv'
INTO TABLE MentalHealth
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
IGNORE 1 ROWS
(@Timestamp, @Gender, @Age, @Course, @YearOfStudy, @CGPA, Depression, Anxiety, PanicAttack, Special
SET
    StudentID = (
        SELECT StudentID
        FROM Student
        WHERE Gender = @Gender AND Age = CAST(@Age AS UNSIGNED) AND Course = TRIM(@Course) AND Year
        LIMIT 1
    ),
    Depression = CAST(Depression AS UNSIGNED),
    Anxiety = CAST(Anxiety AS UNSIGNED),
    PanicAttack = CAST(PanicAttack AS UNSIGNED),
    SpecialistTreatment = CAST(SpecialistTreatment AS UNSIGNED),
    SymptomFrequency_Last7Days = CAST(SymptomFrequency_Last7Days AS UNSIGNED),
    HasMentalHealthSupport = CAST(HasMentalHealthSupport AS UNSIGNED);

-- Load data into the Lifestyle table
LOAD DATA LOCAL INFILE 'data/mentalhealth_dataset.csv'
INTO TABLE Lifestyle
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
IGNORE 1 ROWS
(@Timestamp, @Gender, @Age, @Course, @YearOfStudy, @CGPA, @Depression, @Anxiety, @PanicAttack, @Spe
SET
    StudentID = (
        SELECT StudentID
        FROM Student
        WHERE Gender = @Gender AND Age = CAST(@Age AS UNSIGNED) AND Course = TRIM(@Course) AND Year
        LIMIT 1
    ),
    SleepQuality = CAST(SleepQuality AS UNSIGNED),
    StudyStressLevel = CAST(StudyStressLevel AS UNSIGNED),
    StudyHoursPerWeek = CAST(StudyHoursPerWeek AS UNSIGNED),
    AcademicEngagement = CAST(AcademicEngagement AS UNSIGNED);
```

Figure 6: LOAD DATA Commands

these commands are also stored in a same file named setup.sql

**3. Reflect on how well the database reflects the data**

The database structure successfully reflects the dataset by organizing it into three key tables: Student, MentalHealth and Lifestyle. The Student table stores basic personal and academic information, while the MentalHealth and Lifestyle tables capture the respective data about students' mental health and lifestyle factors.

The database structure is normalized, with separate tables for students, mental health records and lifestyle data. This prevents data redundancy and ensures that each piece of data (e.g., a student's personal information) is stored in one place, while related information (e.g., mental health symptoms, lifestyle attributes) is stored in separate tables.

The relationships between tables are well established through foreign keys, ensuring that data integrity is maintained. For instance, the MentalHealth and Lifestyle tables reference the Student table, ensuring that each record is linked to a valid student. For example, if a student is deleted from the Student table, their corresponding records in MentalHealth and Lifestyle are also deleted automatically, which is managed through the ON DELETE CASCADE clause.

The data types chosen for each field are generally appropriate, with TINYINT used for boolean-like values (e.g., mental health symptoms, treatment, and support) and INT or DECIMAL for numeric data like age, CGPA, and lifestyle factors.

The LOAD DATA commands are well-structured and efficiently import the dataset into the database, transforming and mapping the data to fit the schema. The data seems to be consistently processed, with fields like Gender, Course and CGPA being trimmed and cast to the appropriate types to maintain consistency.

Overall, the structure supports the dataset well, though there could be minor improvements in terms of handling NULL values and ensuring consistency in the data, particularly for cases where some attributes might be missing or unreported, such as in the case of depression, anxiety or panicAttack, etc.

**Data Cleaning:**

1. **Data Type Casting:** When loading data into the database, there are several instances where the code casts values to specific data types (e.g., CAST(Age AS UNSIGNED), CAST(CGPA AS DECIMAL(4,2))). This ensures that the data being inserted into the database conforms to the expected types defined

in the table schema.

2. **Trimming Whitespace:** The code uses `TRIM()` to remove any leading or trailing whitespace from string values (e.g., `TRIM(Course)`). This is a basic form of data cleaning to ensure that the data stored in the database is clean and consistent.

3. **Data Validation:** The application performs some validation when querying data (e.g., checking if the selected mental health issue is valid).

### 4. List SQL commands

All SQL commands are saved and stored in a file named setup.sql. Below are the commands for the questions listed in the same order as in **Stage 1: Step 3**.

```
94   -- Q 1: How does the average academic engagement and stress leve
95   -- health support and those who do not?
96   SELECT
97       HasMentalHealthSupport,
98       AVG(AcademicEngagement) AS AverageAcademicEngagement,
99       AVG(StudyStressLevel) AS AverageStudyStressLevel
100  FROM
101      MentalHealth
102  JOIN
103      Lifestyle ON MentalHealth.StudentID = Lifestyle.StudentID
104  GROUP BY
105      HasMentalHealthSupport;
106
107  -- Q 2: What is the relationship between students' CGPA, study h
108  SELECT
109      CGPA,
110      StudyHoursPerWeek,
111      SleepQuality
112  FROM
113      Student
114  JOIN
115      Lifestyle ON Student.StudentID = Lifestyle.StudentID;
```

```sql
-- Q 3: What is the distribution of specialist treatment among stud
SELECT
    Course,
    COUNT(SpecialistTreatment) AS SpecialistTreatmentCount
FROM
    Student
JOIN
    MentalHealth ON Student.StudentID = MentalHealth.StudentID
GROUP BY
    Course;

-- Q 4: How does the prevalence of mental health issues vary betwee
SELECT
    Gender,
    AVG(Depression) AS AverageDepression,
    AVG(Anxiety) AS AverageAnxiety,
    AVG(PanicAttack) AS AveragePanicAttack
FROM
    MentalHealth
JOIN
    Student ON MentalHealth.StudentID = Student.StudentID
GROUP BY
    Gender;
```

```sql
-- Q 5: How does the prevalence of different mental health i
SELECT
    YearOfStudy,
    AVG(Depression) AS AverageDepression,
    AVG(Anxiety) AS AverageAnxiety,
    AVG(PanicAttack) AS AveragePanicAttack
FROM
    MentalHealth
JOIN
    Student ON MentalHealth.StudentID = Student.StudentID
GROUP BY
    YearOfStudy;
```

setup.sql

```sql
-- Q 6: How is the frequency of specialist treatment related to
SELECT
    Course,
    SleepQuality,
    COUNT(SpecialistTreatment) AS TreatmentFrequency
FROM
    Student
JOIN
    MentalHealth ON Student.StudentID = MentalHealth.StudentID
JOIN
    Lifestyle ON Student.StudentID = Lifestyle.StudentID
GROUP BY
    Course, SleepQuality;

-- Q 7: What is the frequency distribution of a selected mental
-- been made for each frequency level?
SELECT
    SymptomFrequency_Last7Days,
    COUNT(*) AS FrequencyCount
FROM
    MentalHealth
GROUP BY
    SymptomFrequency_Last7Days;
```

Figure 7: SQL Commands

And below are the answers to these commands typed in a terminal, listed in the same order as the questions.

```
+------------------------+---------------------------+-------------------------+
| HasMentalHealthSupport | AverageAcademicEngagement | AverageStudyStressLevel |
+------------------------+---------------------------+-------------------------+
|                      0 |                    3.0775 |                  3.0697 |
|                      1 |                    3.2766 |                  3.0745 |
+------------------------+---------------------------+-------------------------+
2 rows in set (0.146 sec)
```

| CGPA | StudyHoursPerWeek | SleepQuality |
|---|---|---|
| 2.38 | 8 | 4 |
| 4.00 | 13 | 4 |
| 3.68 | 13 | 1 |
| 4.00 | 19 | 5 |
| 2.00 | 3 | 2 |
| 4.00 | 15 | 4 |
| 2.00 | 1 | 1 |
| 2.72 | 8 | 3 |
| 2.30 | 5 | 4 |
| 4.00 | 16 | 4 |
| 4.00 | 16 | 4 |
| 2.00 | 6 | 1 |
| 4.00 | 18 | 4 |
| 2.00 | 1 | 5 |

| Course | SpecialistTreatmentCount |
|---|---|
| Accounting | 11 |
| ALA | 6 |
| Banking Studies | 6 |
| BCS | 177 |
| BENL | 24 |
| Biomedical science | 33 |
| Biotechnology | 8 |
| BIT | 101 |
| Business Administration | 14 |
| Communication | 13 |
| CTS | 15 |
| Diploma Nursing | 8 |

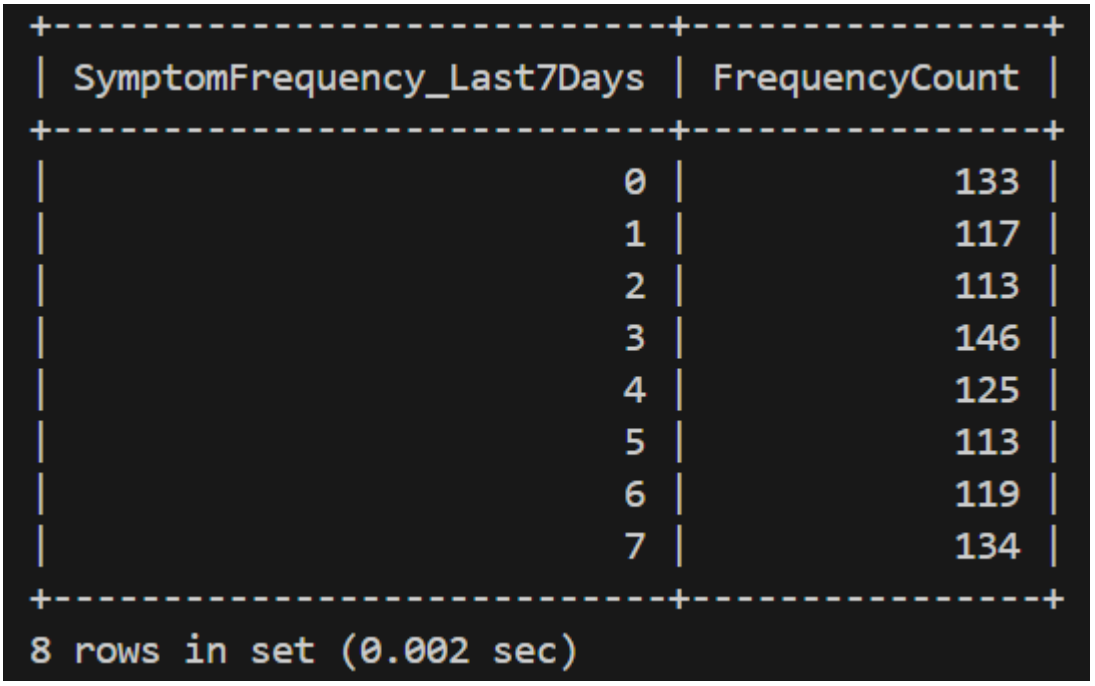
| Gender | AverageDepression | AverageAnxiety | AveragePanicAttack |
|---|---|---|---|
| Male | 0.4458 | 0.4792 | 0.4458 |
| Female | 0.4947 | 0.4724 | 0.4618 |

2 rows in set (0.007 sec)

| YearOfStudy | AverageDepression | AverageAnxiety | AveragePanicAttack |
|---|---|---|---|
| year 1 | 0.4684 | 0.4854 | 0.4490 |
| Year 2 | 0.4562 | 0.4708 | 0.4599 |
| Year 3 | 0.5292 | 0.4625 | 0.4667 |
| year 4 | 0.5135 | 0.4595 | 0.4730 |

4 rows in set (0.005 sec)

| Course | SleepQuality | TreatmentFrequency |
|---|---|---|
| Accounting | 1 | 5 |
| Accounting | 2 | 2 |
| Accounting | 3 | 1 |
| Accounting | 4 | 2 |
| Accounting | 5 | 1 |
| ALA | 2 | 2 |
| ALA | 3 | 1 |
| ALA | 4 | 2 |
| ALA | 5 | 1 |
| Banking Studies | 1 | 2 |
| Banking Studies | 3 | 4 |
| BCS | 1 | 56 |
| BCS | 2 | 42 |



| SymptomFrequency_Last7Days | FrequencyCount |
|---|---|
| 0 | 133 |
| 1 | 117 |
| 2 | 113 |
| 3 | 146 |
| 4 | 125 |
| 5 | 113 |
| 6 | 119 |
| 7 | 134 |

8 rows in set (0.002 sec)

Figure 8: Answers to SQL Commands

## Stage 4. Create a simple web application

**1. Write a node.js module to present a web application that queries the database.**

The web app is a simple data visualization tool that provides insights into students' mental health. The app's design is also kept simple, with interactive graphs created using Chart.js. In the lab environment, all the tables and instance data are already loaded into the database, as discussed in **Stage 3: Step 1 & 2 (Figures 5 & 6)**.

In the lab's browser, the dropdown does not open when clicked. Instead, you can use the up and down arrow keys to navigate the options after clicking the dropdown with the mouse. Once you've selected an option, you may need to press Enter. If the dropdown contains a button, such as 'Filter,' click that instead of pressing Enter.

In case the database information is lost or corrupted, I have created a dump file named **mental_health_dump.sql** to ensure that my work is reproducible and transportable. This backup ensures that I can restore the database if needed, share my work easily for grading and protect against potential data loss.

**Steps to run the app:**

1) `CREATE DATABASE mental_health;` ------> create the database (done by me)

2) USE mental_health; ------> connect to the database, create tables and enter instance data from setup.sql (done by me)

4) USE mental_health; ------> To answer any of the 7 questions, use the commands in setup.sql as seen in **figure 7**

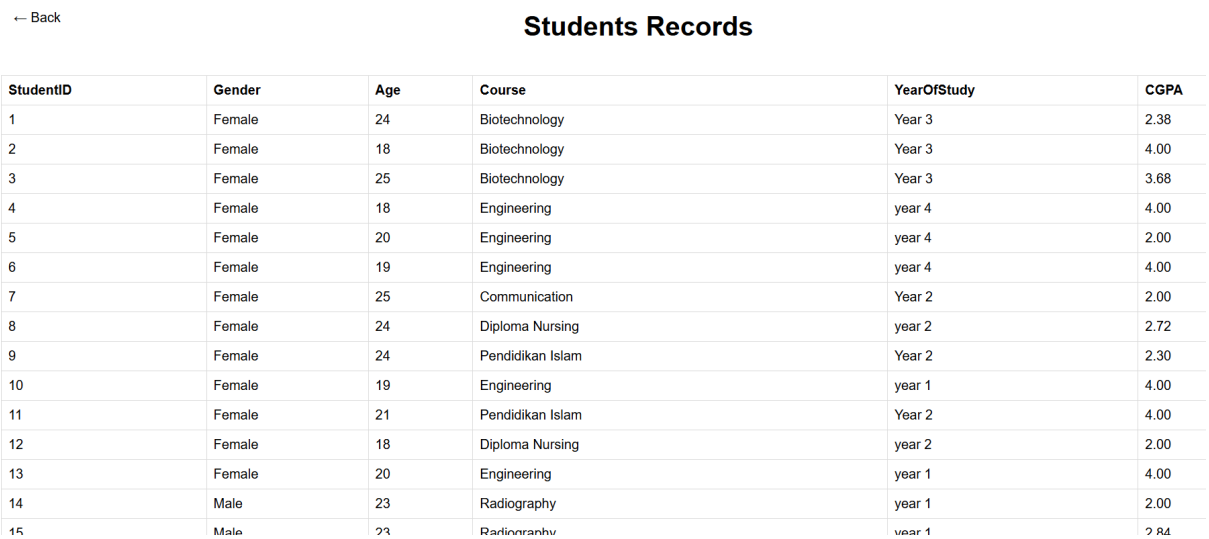
3) node app.js ------> run the web-app, then copy the URL http://localhost:3000 into the Lab's integrated browser

- Link To the lab (with the database already created) - Coursera Lab Link

## 2. Take screenshots

Note: Screenshots are limited and do not fully display the entire view of some pages due to not fitting in a screen.
Main Home page: Displays all the navigation links to other pages


Student page: Display records of all pages




| StudentID | Gender | Age | Course | YearOfStudy | CGPA |
|---|---|---|---|---|---|
| 1 | Female | 24 | Biotechnology | Year 3 | 2.38 |
| 2 | Female | 18 | Biotechnology | Year 3 | 4.00 |
| 3 | Female | 25 | Biotechnology | Year 3 | 3.68 |
| 4 | Female | 18 | Engineering | year 4 | 4.00 |
| 5 | Female | 20 | Engineering | year 4 | 2.00 |
| 6 | Female | 19 | Engineering | year 4 | 4.00 |
| 7 | Female | 25 | Communication | Year 2 | 2.00 |
| 8 | Female | 24 | Diploma Nursing | year 2 | 2.72 |
| 9 | Female | 24 | Pendidikan Islam | Year 2 | 2.30 |
| 10 | Female | 19 | Engineering | year 1 | 4.00 |
| 11 | Female | 21 | Pendidikan Islam | Year 2 | 4.00 |
| 12 | Female | 18 | Diploma Nursing | year 2 | 2.00 |
| 13 | Female | 20 | Engineering | year 1 | 4.00 |
| 14 | Male | 23 | Radiography | year 1 | 2.00 |
| 15 | Male | 23 | Radiography | year 1 | 2.84 |

Mental Health Issue page: Display all the students with selected mental health issue

← Back

## Students with Mental Health Issues Records

Select Mental Health Issue: Depression Filter

Depression
Anxiety
PanicAttack

| StudentID | Gender | Age | Course | YearOfStudy | CGPA | Depression |
|---|---|---|---|---|---|---|
| 1 | Female | 24 | Biotechnology | Year 3 | 2.38 | 1 |
| 5 | Female | 20 | Engineering | year 4 | 2.00 | 1 |
| 7 | Female | 25 | Communication | Year 2 | 2.00 | 1 |
| 10 | Female | 19 | Engineering | year 1 | 4.00 | 1 |
| 11 | Female | 21 | Pendidikan Islam | Year 2 | 4.00 | 1 |
| 12 | Female | 18 | Diploma Nursing | year 2 | 2.00 | 1 |
| 13 | Female | 20 | Engineering | year 1 | 4.00 | 1 |
| 16 | Female | 18 | psychology | year 1 | 3.28 | 1 |
| 17 | Female | 19 | Fiqh fatwa | Year 3 | 4.00 | 1 |
| 19 | Female | 25 | psychology | year 1 | 2.00 | 1 |

Relationship page: Display correlation between CGPA and mental health issues

← Back

## Correlation Between CGPA and Mental Health Issues

| CGPA | Depression | Anxiety | PanicAttack |
|---|---|---|---|
| 2.38 | 1 | 0 | 0 |
| 4.00 | 0 | 1 | 0 |
| 3.68 | 0 | 0 | 1 |
| 4.00 | 0 | 0 | 0 |
| 2.00 | 1 | 1 | 0 |
| 4.00 | 0 | 0 | 1 |
| 2.00 | 1 | 1 | 0 |
| 2.72 | 0 | 0 | 0 |
| 2.30 | 0 | 0 | 0 |
| 4.00 | 1 | 1 | 0 |
| 4.00 | 1 | 0 | 0 |
| 2.00 | 1 | 0 | 0 |

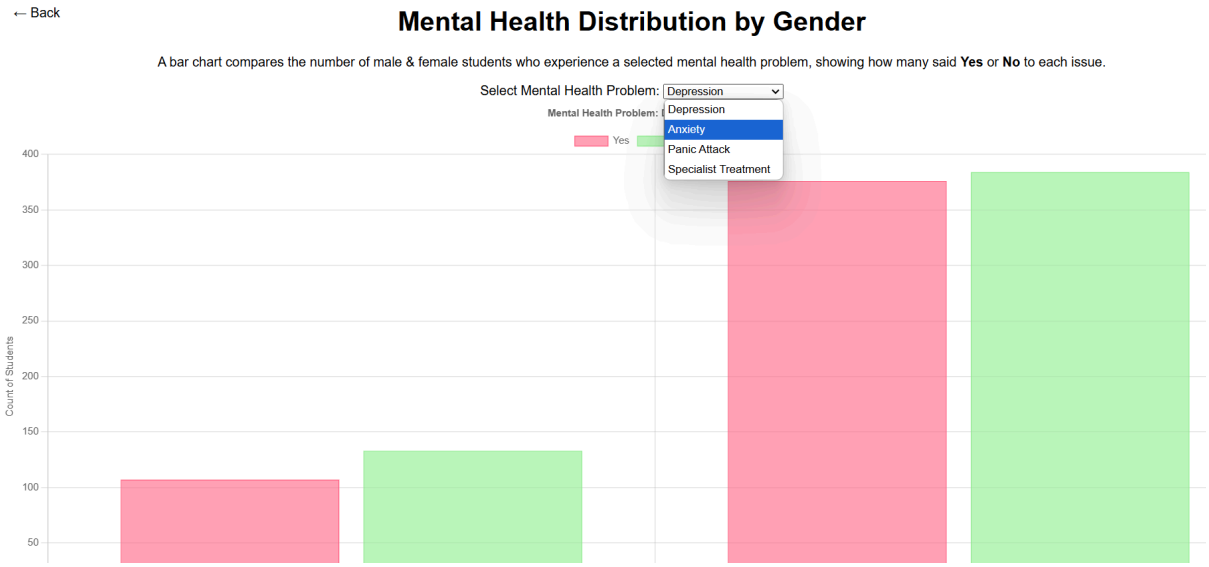
Statistics page: Display key statistics that provide insights into data

← Back

# Statistics Dashboard

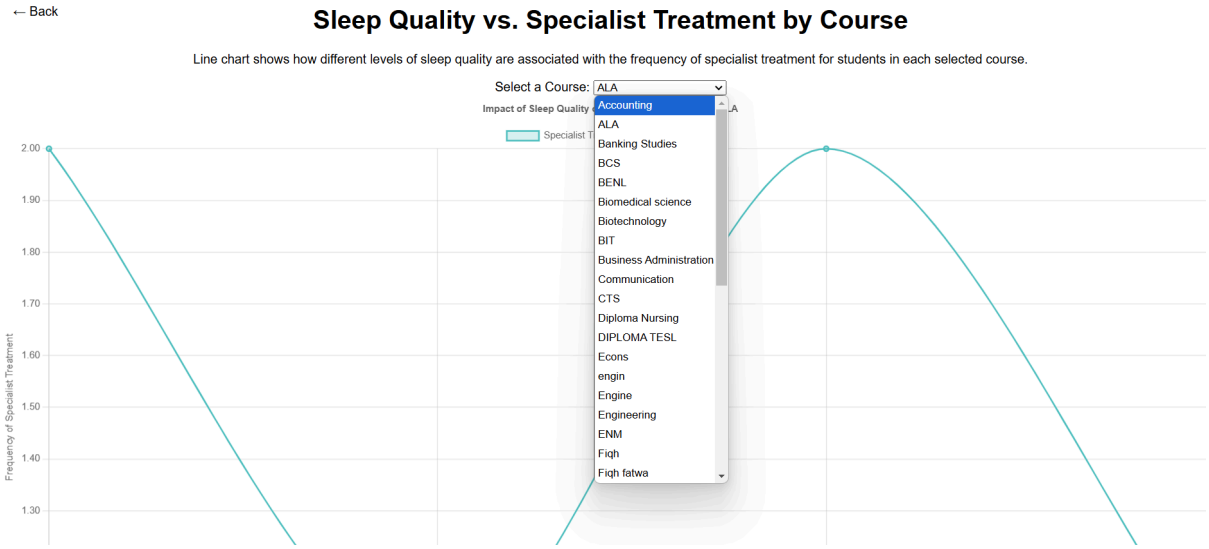
| Statistic | Value |
|---|---|
| Total Students | **TotalStudents:** 1000 |
| Average CGPA | **AvgCGPA:** 3.122530 |
| Average Age | **AvgAge:** 21.4020 |
| Total Depression Cases | **TotalDepressionCases:** 483 |
| Total Anxiety Cases | **TotalAnxietyCases:** 474 |
| Total Panic Attack Cases | **TotalPanicAttackCases:** 458 |
| Average Sleep Quality | **AvgSleepQuality:** 2.9830 |
| Average Study Stress Level | **AvgStudyStressLevel:** 3.0450 |
| Average Study Hours Per Week | **AvgStudyHoursPerWeek:** 9.7460 |
| Average Academic Engagement | **AvgAcademicEngagement:** 3.0550 |
| Gender Distribution | **Gender:** Male<br>**Total:** 240<br>**Gender:** Female |

Mental Health by Gender page: Answers Q4 with stacked bar charts

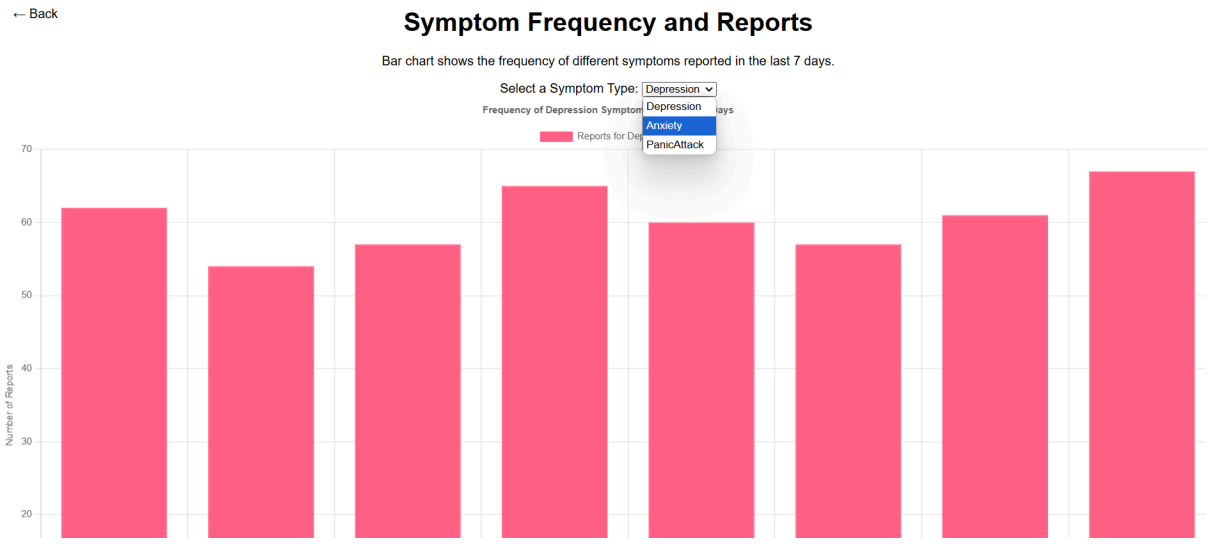



Mental Health by Year page: Answers Q5 by bar charts with dropdown menu selecting mental health issue

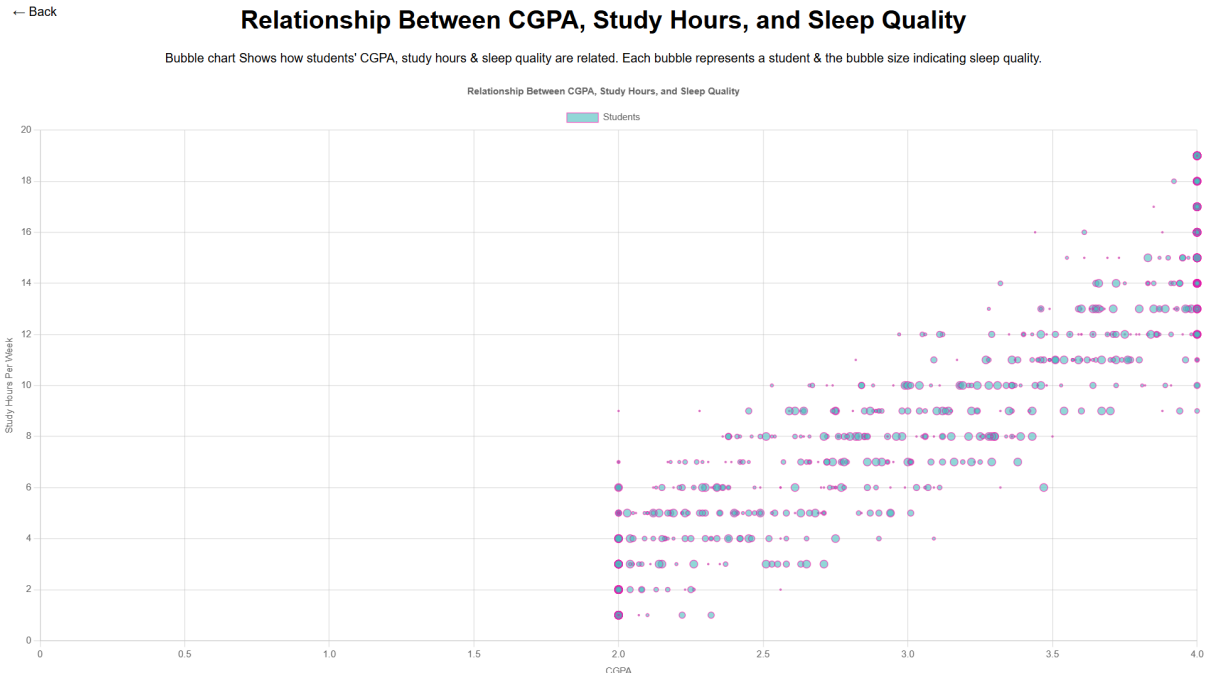

Mental Health by Course page: Answers Q3 by Line graph with a dropdown menu selecting course




Symptom Freq and Report page: Answers Q7 by bar charts with dropdown menu selecting mental health issue




CGPA, study hours and sleep quality page: Answers Q2 with a bubble chart




Stress and mental health support page: Answers Q1 with stacked bar charts

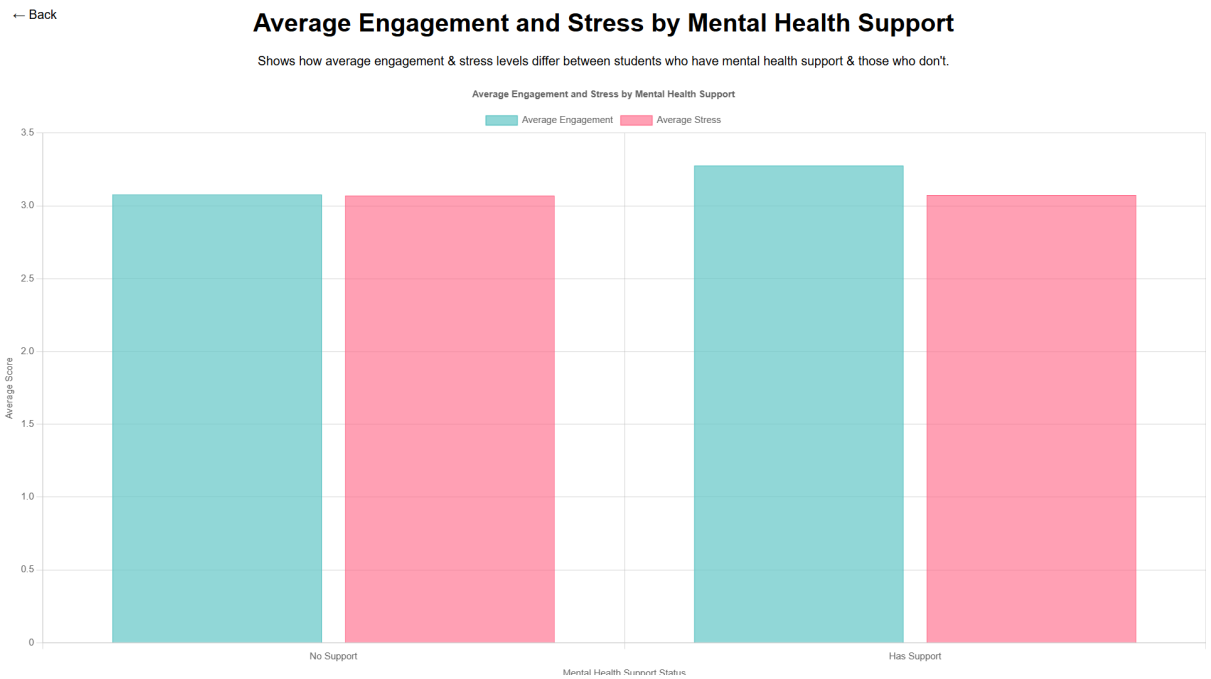



Figure 9: All the pages of web app

**Resources used:**

**Code Development:** All code was written by myself, except for the JavaScript code, for which I referred to various tutorials and documentation.

**JavaScript Resources:**

- JavaScript Documentation - MDN Web Docs - JavaScript
- JavaScript Tutorial - W3Schools - JavaScript
- Chart.js Documentation - Chart.js Docs
- Chart.js Examples - Chart.js Examples

**ER Diagram Development:**

Although the ER diagrams may appear to be automatically generated, I wrote the code myself to create them myself. The links to the code I used for generating the diagrams are as follows:

- Figure 1: Diagram with Cardinality - Graphviz Code for Figure 1
- Figure 2: ERD Flow Diagram - Graphviz Code for Figure 2

I Created diagrams For Figure 3 & 4 From - eraser.io