

PROJECT PLAN

1. General information

Title: NIGHTMARE ASSISTANT (own topic)

Creator: Rosanna Wang

Student number: 100928587

Study program: Bachelor of Arts in Design

Study year and date: 01.09.2022-01.07.2025

2. General description and difficulty level

The program is designed to help individuals manage their nightmares through 'nightmare coaching' sessions, which incorporate elements of Imagery Rehearsal Therapy (IRT). As a self-guided approach to self-care, the program empowers individuals to gain positive control over their nightmares.

NB: Although Image Rehearsal Therapy is used for both PTSD and non-PTSD patients, this project is primarily intended for non-trauma-related nightmares, as trauma-related nightmares may benefit more from specific therapy sessions with a therapist for an effective approach.

The goal of the program is to provide a safe space for users to process emotions and feelings, hopefully allowing them to overcome the negative effects associated with their nightmares. The chosen difficulty level for the project is hard, as agreed with the teacher.

3. Use case description and draft of the user interface

At the beginning of the therapeutic session, a short description of Image Rehearsal Therapy is provided through a pop-up window: "The key steps of Image Rehearsal Therapy consist of first writing and describing the main narrative of the bad dream (and recalling negative emotions associated with it), then rewriting the storyline by modifying the negative elements with positive elements to make the

dream less distressing. Finally, the last step involves mentally rehearsing the new dream."

A second window will show a short description of the program: "The program is designed to help individuals manage their nightmares through 'nightmare coaching' sessions, which incorporate elements of Imagery Rehearsal Therapy (IRT), a cognitive-behavioral treatment that has been scientifically proven to be effective in reducing the number and intensity of nightmares".

Then, the possible flow of the program would be as follows:

0. The user has the option to start from scratch or upload a .txt file (with the correct formatting) to the program, so it can read and process the information. Jump to 3 if a file is uploaded.
1. A short pre-session survey for the user to indicate their emotional connections to the nightmare (e.g., distress, sleep quality, fear towards the nightmare, etc.). User action: The user drags a scale from 1 to 5 to indicate their response. Program output: Text and elements screen.
2. The program first asks the user to input/describe the main narrative of the bad dream by answering questions. A storyline is generated based on user inputs. User action: Typing. Program output: Screen text.
3. The program then asks the user to identify/highlight the text with respective negative elements. User action: Highlighting and press done. Program output: Screen text.
4. The new text with white spaces is returned to the user. User action: Reading. Program output: Screen text.
5. The program asks the user to replace the white spaces with positive elements. The program asks the user if they want to choose replacements from a predefined library* (a collection of positive elements collected by the developer) or their personal library (a collection of their own favourite positive elements, such as text and images, that the user can use when rewriting the storyline). The user

drags the elements from the library and puts them on the text. User actions: Choosing and dragging. Program output: Library display.

5. The program rewrites the storyline by replacing negative elements with positive ones and returns it to the user. Additionally, the program prompts the user to **write a positive ending based on the new storyline**. User action: Typing. Program outputs: Processing and text-screen.

6. The program prompts **post-session the survey** again and asks the user to complete it. The program compares the results and returns them to the user. User action: Dragging. Program output: Text and elements screen.

7. At the end, the program congratulates the user for the success of the therapy and shows if there have been improvements according to the survey results. It will also encourage the user to come back again through a pop-up window message. User action: None (or eventually exit). Program output: Text pop-up window message.

8. A **report (txt.file)** is generated afterwards about the session for the user to share with others if they want. User action: None (or eventually exit). Program output: Text window.

*For the objective of the project, the predefined library is primarily intended to support the running of the program. This means the choice of elements in the library may not be based on detailed research or analysis. Instead, the priority is to ensure that the program operates as intended (e.g., getting classes and functions to work, along with interactions between the program and the user, input-output); however, the library can be refined at a later stage.

The first version of the GUI will be a text-based interface, with graphical elements being gradually added. An additional feature might include that after writing the positive ending, the program asks the user if they want to **create the new scenario in the form of a collage (+ library feature here)**. Here, they will first choose a background image (either from the developer library or they can upload their own one) and fill the board with the elements of the new dream. Another

additional feature would be allowing users to track their progress over time. This could include saving, (loading) or reviewing previous sessions.

With these features, the program offers the possibility to create a personalised nightmare experience simulated within a 2D environment. Future developments could incorporate more advanced natural language processing models to enhance text parsing and improve user interactions.

4. Program's structure plan

[Link](#) to the figma prototype (showing UI and an example of the user flow)

Main.py: Responsible for starting the programme, coordinating interactions between classes and between the program and the user, and defining the main processes.

DataManager: This class will handle user inputs and perform all data-related operations.

GUI classes (in one unique .py file):

1. Start

NIGHTMARE COUCH

Welcome to the Nightmare Coaching Program!

Start

User action: press start button



2. Enter a title

| NIGHTMARE COUCH | NIGHTMARE COUCH |
|---|--|
| <p>What is your nightmare about? Please give a short title.</p> <input type="text"/> <input type="button" value="Save"/> | <p>What is your nightmare about? Please give a short title.</p> <input type="text" value="Being chased"/> <input type="button" value="Save"/> |

User actions: type something > press save button

Program actions: process and store user input as string within the same .py file (`nightmare_title`)

3. pre-session survey

| NIGHTMARE COUCH | NIGHTMARE COUCH |
|---|--|
| <p>On a scale of 0-10, how scared does your nightmare make you feel?</p> <p>0 - Neutral 10 - Highest possible</p>  <p><input type="button" value="Continue"/></p> | <p>On a scale of 0-10, how scared does your nightmare make you feel?</p> <p>0 - Neutral 10 - Highest possible</p>  <p><input type="button" value="Continue"/></p> |

User actions: choose a number on the slider

Program actions: process and store the value as integer within the same .py file

4. Describe the main narrative of the bad dream

| NIGHTMARE COUCH | NIGHTMARE COUCH |
|--|---|
| <p>Could you describe your nightmare?</p> <input type="text" value="how does it begin?"/> <input type="text" value="how does it continue?"/> <input type="text" value="how does it end?"/> <input type="text" value="how were you feeling?"/> <input type="text" value="any other details?"/> <input type="button" value="Continue"/> | <p>Could you describe your nightmare?</p> <input type="text" value="I'm walking alone in a dark forest, surrounded by towering trees and silence."/> <input type="text" value="how does it continue?"/> <input type="text" value="how does it end?"/> <input type="text" value="how were you feeling?"/> <input type="text" value="any other details?"/> <input type="button" value="Continue"/> |

NIGHTMARE COUCH

Could you describe your nightmare?

I'm walking alone in a dark forest, surrounded by towering trees and silence.

Shadows begin to follow me, whispering faintly as they get closer no matter how fast I run.

how does it end?

how were you feeling?

any other details?

Continue

NIGHTMARE COUCH

Could you describe your nightmare?

I'm walking alone in a dark forest, surrounded by towering trees and silence.

Shadows begin to follow me, whispering faintly as they get closer no matter how fast I run.

I trip, and faceless figures surround me, leaning closer as I try to scream but can't make a sound.

how were you feeling?

any other details?

Continue

NIGHTMARE COUCH

Could you describe your nightmare?

I'm walking alone in a dark forest, surrounded by towering trees and silence.

Shadows begin to follow me, whispering faintly as they get closer no matter how fast I run.

I trip, and faceless figures surround me, leaning closer as I try to scream but can't make a sound.

I felt frozen with fear, unable to breathe or move.

any other details?

Continue

NIGHTMARE COUCH

Could you describe your nightmare?

I'm walking alone in a dark forest, surrounded by towering trees and silence.

Shadows begin to follow me, whispering faintly as they get closer no matter how fast I run.

I trip, and faceless figures surround me, leaning closer as I try to scream but can't make a sound.

I felt frozen with fear, unable to breathe or move.

The whispers sounded like my name, and I woke up feeling cold and disoriented.

Continue

User actions: type > press the continue button

Program actions: process and store user inputs in a string
(nightmare_narrative) > rebuild the text for the next step

NIGHTMARE COUCH

program is processing...

This takes 5 seconds

5. List negative elements

NIGHTMARE COUCH

Well done!
Sounds like your nightmare? Feel free to edit if necessary!

I'm walking alone in a dark forest, surrounded by towering trees and silence. Shadows begin to follow me, whispering faintly as they get closer no matter how fast I run. I trip, and faceless figures surround me, leaning closer as I try to scream but can't make a sound. I felt frozen with fear, unable to breathe or move. The whispers sounded like my name, and I woke up feeling cold and disoriented.

Now, could you identify the negative elements on it?
Copy and paste the negative elements below.

Done

NIGHTMARE COUCH

Well done!
Sounds like your nightmare? Feel free to edit if necessary!

I'm walking alone in a dark forest, surrounded by towering trees and silence. Shadows begin to follow me, whispering faintly as they get closer no matter how fast I run. I trip, and faceless figures surround me, leaning closer as I try to scream but can't make a sound. I felt frozen with fear, unable to breathe or move. The whispers sounded like my name, and I woke up feeling cold and disoriented.

Now, could you identify the negative elements on it?
Copy and paste the negative elements below.

- alone
- dark
- shadows
- trip
- faceless figures
- frozen
- cold
- disoriented

Done

User actions: type negative words using bullet points

Program actions: update nightmare_narrative > process user input in a list (negative_elements_list)

NIGHTMARE COUCH

program is processing...

This takes 5 seconds

6. Replace with positive elements

NIGHTMARE COUCH

Magic!
Based on the list of negative elements, I've removed all the unpleasant parts from your nightmare. Don't worry if something is left out—you have the power to reshape your dream anytime you want.

Now, let's fill it with positive elements to create your favourite dream!
Remember, in dreams, anything is possible, and you're in control.
Feel free to add or remove details as you wish.

I'm walking ____ in a ____ forest, surrounded by towering trees and ____ ____ begin to follow me, whispering faintly as they get closer no matter how fast I run. I ____ and ____ figures surround me, leaning closer as I try to scream but can't make a sound. I felt ____ with fear, unable to breathe or move. The whispers sounded like my name, and I woke up feeling ____ and ____.

Done

NIGHTMARE COUCH

Magic!
Based on the list of negative elements, I've removed all the unpleasant parts from your nightmare. Don't worry if something is left out—you have the power to reshape your dream anytime you want.

Now, let's fill it with positive elements to create your favourite dream!
Remember, in dreams, anything is possible, and you're in control.
Feel free to add or remove details as you wish.

I'm walking calmly in a sunlit forest, surrounded by towering trees and soft, golden light. Butterflies begin to follow me, whispering faintly as they drift closer no matter how fast I walk. I pause and friendly figures surround me, leaning closer as I try to laugh but can't stop smiling. I felt peaceful, able to breathe deeply and move freely. The whispers sounded like my name, and I woke up feeling warm and refreshed.

Done

User action: type and edit the text > press the done button

Program actions: process and save the text as a new string (variable_name = [new_storyline])

7. Write a positive ending

NIGHTMARE COUCH

Great job!
 You're almost there—just one last step to complete your dream!
 Read it again and add some final touches. Make sure it have a happy ending!

I'm walking calmly in a sunlit forest, surrounded by towering trees and soft, golden light. Butterflies begin to follow me, whispering faintly as they drift closer no matter how fast I walk. I pause and friendly figures surround me, leaning closer as I try to laugh but can't stop smiling. I felt peaceful, able to breathe deeply and move freely. The whispers sounded like my name, and I woke up feeling warm and refreshed.

Finish

NIGHTMARE COUCH

Great job!
 You're almost there—just one last step to complete your dream!
 Read it again and add some final touches. Make sure it have a happy ending!

I'm walking calmly in a sunlit forest, surrounded by towering trees and soft, golden light. Butterflies begin to follow me, whispering faintly as they drift closer no matter how fast I walk. I pause and friendly figures surround me, leaning closer as I try to laugh but can't stop smiling. I felt peaceful, able to breathe deeply and move freely. The whispers sounded like my name, and I woke up feeling warm and refreshed, as though the forest's magic stayed with me.

Finish

User action: type and edit the text > press the finish button
 Program actions: update new_storyline

8. Post-session survey

NIGHTMARE COUCH

On a scale of 0-10, how scared does your dream make you feel?

0 - Neutral
 10 - Highest possible

Continue

NIGHTMARE COUCH

On a scale of 0-10, how scared does your dream make you feel?

0 - Neutral
 10 - Highest possible

Continue

User actions: choose a number on the slider
 Program actions: process and store the value as integer within the same .py file

9. End (image + sound here)



User action: None; or eventually exit
 Program actions: compile everything in a txt.file

Library Classes: The **Library** class is abstract and meant for handling collections of images. It provides navigation functionality and has two subclasses: **StandardLibrary** and **PersonalLibrary**.

Collage Class: The **Collage** class handles interactions between the user and the collage exercise, allowing users to add pictures from the library. It also provides functionality for managing metadata (title, keywords, author, and date) and supports features like reordering images.

5.Data structures (excluding the library/collage features for now)

| Program component | Data Structure |
|----------------------------------|----------------|
| survey results | List |
| user text inputs | String |
| narrative inputs | Dictionary |
| negative elements | Set |
| positive elements in the library | List |

6.Files and file formats

The program would need to be able to handle picture (.png) and sound files (.mp3).

7.Algorithms (excluding the library/collage features for now)

A) Text Processing Algorithms

Algorithms include 1)processing user input (str), 2)identifying and removing negative content based on user-provided words, and 3)processing the negative words.

| Component | Algorithms/Techniques | Why/How |
|---|--|---|
| 1) processing user input (str) main narrative | String Concatenation, Text Formatting (Capitalization, Punctuation Addition) | Combine user inputs (sentences) into a single string. |

| | | |
|--|---|---|
| 2) identifying and removing negative content | Search, replace, rebuild String Tokenization | -Search: Compare each word in the list against the negative words list. -Replace: Replace matched words with blanks (____). -Rebuild: Reconstruct the modified list into a single string. |
| 3) processing the negative words | From string to list, then from list to set | It first converts the string into a list and then into a set. |

B) Data Handling & Report Generation

| Component | Algorithms/Techniques | Why/How |
|-----------------------------------|----------------------------------|--|
| 1) comparing survey results (int) | Simple difference calculation | -Collect pre-session and post-session survey results. -Perform: pre-session score - post-session score = improvement(int) |
| 2) generating the Final Report | String Formatting & File Writing | Create structured .txt files summarizing the session. |

8. Testing plan

I will write tests throughout the development for each of the classes implemented to ensure the program is error-free at each testing stage. These tests will focus on operations and functions that directly impact the program's results, for example verifying that

user inputs are processed as expected or contents are removed correctly from the text based on the list of negative words provided.

The testing plan will fulfill the requirements of the project topic (having unit tests for at least one part of the program) and will ensure that the core functionality works as intended. Additionally, I will add unit tests to verify the additional features I plan to implement.

9. Libraries and other tools

- PyQt6 for GUI
- Unittest for testing

10. Schedule

Since I started this project last year, I decided to start from where I left off. Keeping in mind the deadlines:

(Information copied and pasted from the instructor's slides)

| Task | Deadline | Notes |
|--|--|---|
| Submit your project plan to GitLab | 21.2. at 14:00 | Your TA arranges a time with you to demo the plan |
| Demo of the plan (to your assigned TA) | 28.2 (15 mins demo) | You start coding |
| Submit Checkpoint 1 (must contain code) | 21.3 at 14:00 (send to GitLab + email your TA) | You continue working based on the feedback |
| Submit Checkpoint 2 (must contain tests) | 11.4 at 14:00 (send to GitLab + email your TA) | You continue working based on the feedback |
| FINAL submission + demo time reservation (NO EXTENSIONS) | 9.5 at 14:00 (submission to GitLab and A+) | You show present your work in arranged demo session |
| Project demos | 16.5. (according to TA's | You continue writing awesome programs :) |

| | | |
|--|------------|--|
| | timetable) | |
|--|------------|--|

- ☒ ~~Week 1: ui and main.py~~
- ☒ ~~Week 2: functions in the data.manager file and main.py~~
- ☒ ~~Week 3: week2 (writing & reading file) / unittest~~
- ☒ ~~Week 4: start library/collage~~
- ☒ ~~Week 5: improve library/collage~~
- ☒ ~~Week 6: finalise library/collage~~
- ☒ ~~Week 7: finalise library/collage + QProgressBar~~
- ☒ ~~Week 8: review everything + adding description for my codes~~
- ☒ ~~Week 9: project documentation~~
- ☐ Week 10: submission to GitLab and A+ + prepare for the demo
- ☐ Week 11: demo session

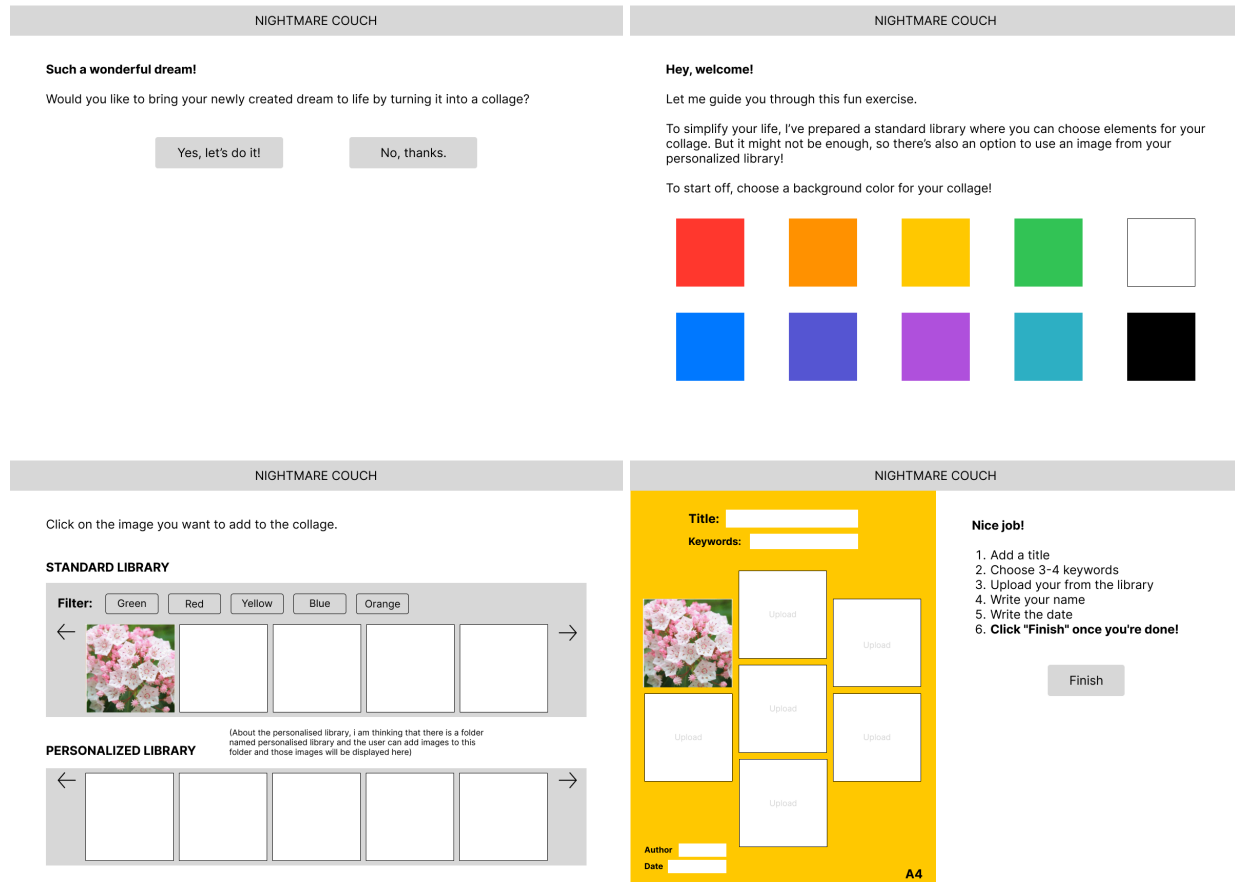
I'll be writing unit tests throughout the process. I've already started implementing a few features with the GUI, and I'm currently refining them. (12.02.2025)

- ☒ ~~Image Rehearsal Therapy session~~
- ☒ ~~PreSurvey and EndSurvey~~
- ☒ ~~Reading file in the beginning~~
- ☒ ~~Writing file in the end~~
- ☒ ~~End screen: image + background music~~

Once I've completed everything above (expected by March), I'll focus on how to visualize the **library/collage feature** and present it at the first checkpoint. After that, I'll start implementing the feature, aiming to finish before the second checkpoint, with possible refinements over the following two weeks. Any extra time will go toward polishing the documentation and preparing for the final project demo.

I would like to discuss the library/collage feature with my project assistant to make it feasible and relevant to the course. Here's the link to the Figma prototype of how I'm imagining it: [link](#).

Some screenshots:



11.Literature references and links

A brief guide to IRT:

<https://wichitasac.com/wp-content/uploads/2018/10/Reflections-Aw-Brief-Guide-to-Imagery-Rehearsal-Therapy-for-Nightmares.pdf>

Some real-life examples of platforms that incorporate IRT:

<https://ux-design-awards.com/winners/2023-2-otherworld>

https://s18798.pcdn.co/northstar/wp-content/uploads/sites/20115/2021/01/Dream-EZ-_NORTHSTAR_4.15.2019.pdf

<https://nightware.com/>

Python documentation: <https://docs.python.org/3/>

Unittest: <https://docs.python.org/3/library/unittest.html>

PyQt6: <https://www.riverbankcomputing.com/static/Docs/PyQt6/>

Round 5, Reading files:

https://plus.cs.aalto.fi/y2/2025/materials_k05/k05_fileinput/

QMediaPlayer:

<https://stackoverflow.com/questions/69415713/playing-sounds-with-pyqt6>

PyQt QMainWindow

<https://www.pythontutorial.net/pyqt/pyqt-qmainwindow/>

Datetime

<https://docs.python.org/3/library/datetime.html>

<https://www.programiz.com/python-programming/datetime/strftime>

12. Attachments

UML diagram,

One UI class equals one window, and key UI classes are shown in this diagram.

UI classes

