UI Design for Our Project - Gizelle Mendoza
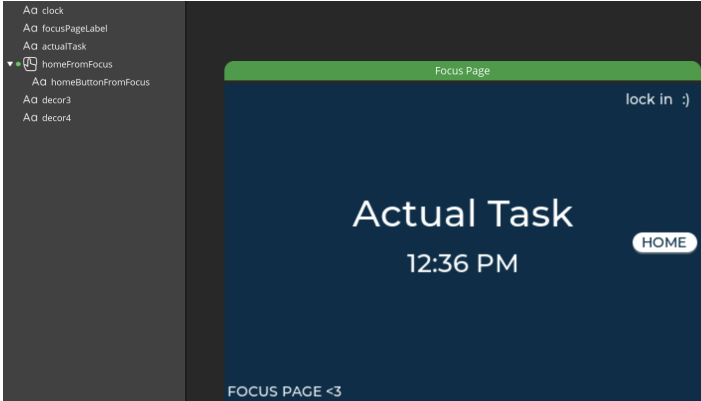
Our **Focus Page** is designed to help users stay on track with their routines. It displays the current task alongside the current time, providing a distraction-free interface to help users stay accountable for their work.
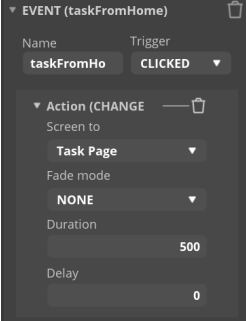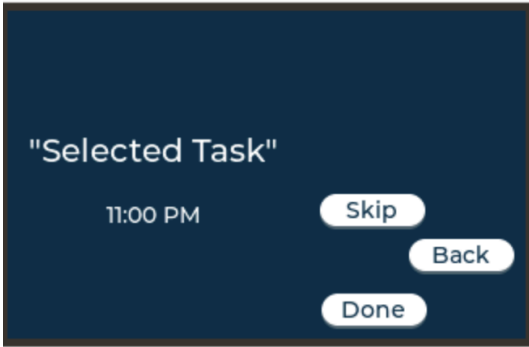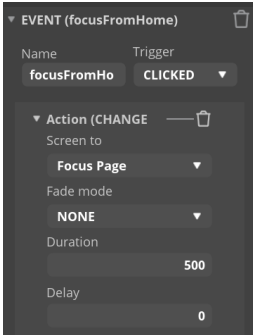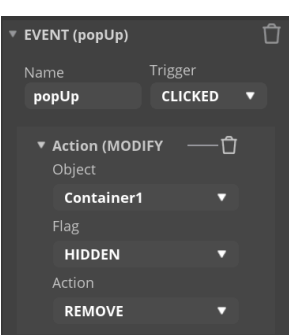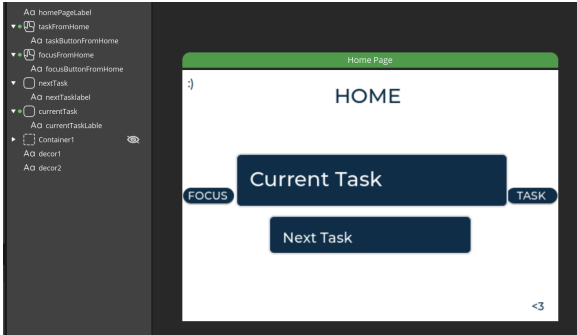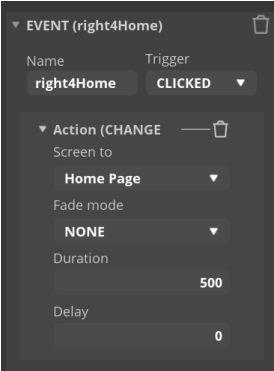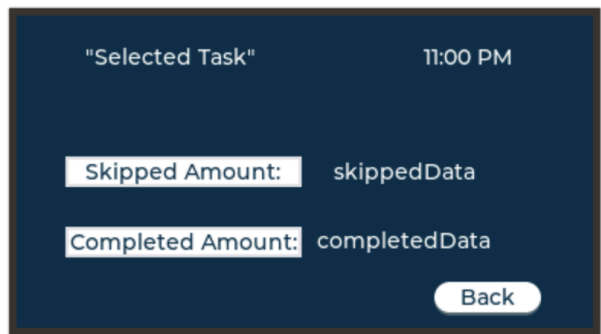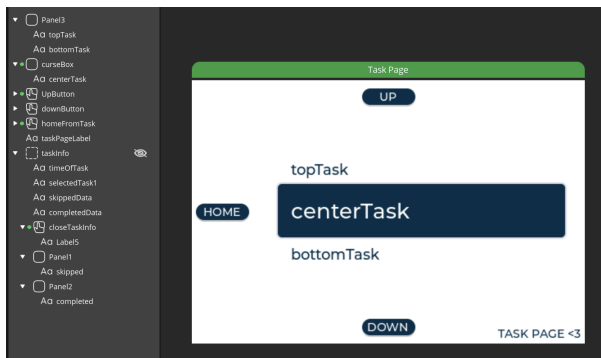

The **Home Page** serves as a central hub where users can view both their current task and the

next one in their routine. This page also includes functionality to skip or complete tasks, empowering users to manage their schedule efficiently.

The **Task Page** offers a comprehensive view of the user's entire schedule. Here, users can track their progress by reviewing how many times they've completed or skipped specific tasks, providing valuable insights into their habits.

All pages are made using **LVGL widgets**:

Labels: Used to display default text, such as task names and times, dynamically updated to reflect the user's schedule.

Buttons: Customized through code to display task-specific information. These buttons are
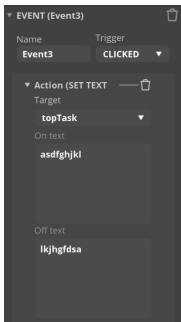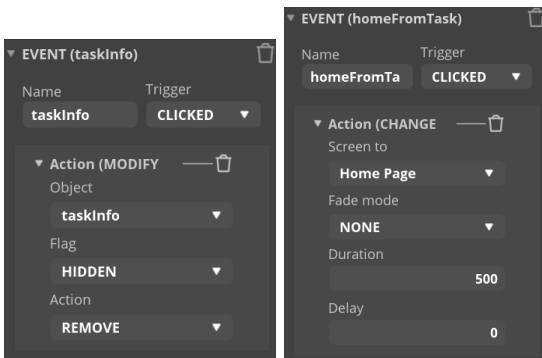
equipped with **LVGL events**, triggered by custom functions in our project file. They integrate seamlessly with hardware button states using interrupts, ensuring smooth user interaction.

Containers: Used for pop-up pages to deliver additional feedback on the users routine. These containers, like the buttons, are programmed with custom and LVGL functions to handle events efficiently.

Our UI combines functionality, clarity, and responsiveness to create an intuitive experience that aligns with our project's goals.

Our focus page is where the user can display the current task from their routine as well as the current time. It's meant for them to be free of distractions and stay accountable for what they should be working on.
Our home page is where the user can see the current task and next task, here they are able to skip or complete tasks.
Our Task page is where the user is able to look throught their entire schedule. Here they will be able to see the amount of times they have completed or skipped certain taks.

All the pages are made from several LVGL widgets such as labels which allows us to have a default text as shown in the image above. We also used lvgl buttons, both of which we changed with the code to accurately display the text according to the users scheduled tasks. The buttons also had LVGL events which we were able to trigger by making custom functions in our project file so that they would interact with our hardware button states using interpts when the user pressed them.

Other LVGL elements used were containers, which we used to serve as a popup page for the user to have more feedback with their routines. These also had events which we programmed through our project file using custom functions and LVGL functions.