How to use Mutter

What is “px”?

The measurement of px stands for pixels. The reason for this is that the layout has fixed widths, meaning input is limited by the amount of pixels available.

Key mappings

Enter/Typing	Input text into chat
Ctrl	Toggle quick help screen
Scroll mouse wheel OR Up/Down Arrows	Scroll chat
Left/Right Arrows	Volume control

Setting a timer

Numbers must be in a two-digit format, for example 2 will be 02, 5 will be 05. The timer is valid only when the text fits the 335px limit and the format is followed. The limit shows on the left of the input when you type.

Timers are made by typing them at the end of a text

“Drawing some concept sketches 10m”

“Coding 01h30m”

Add lowercase g to set a global (or group) timer

“Let’s meet at the group room 03h20mg”

“Deadline for submission 02hg”

Manipulating timers

Change your timer by setting it again as needed. Or you may use the buttons found on the bottom right. You may only add to the timer a certain amount of times.

00m	Removes your current timer
30m	Sets a 30m break OR adds 30m to the current timer
04h	Sets a 04h snooze OR adds 04h to the current timer

Syncing

Mutter automatically sends out sync data every 8 seconds. If the user who set the timer is not online, data will not be synced. Clicking the global timer sets you as the global timer sync.

The taskbar/program title

The idle count shows how many users have no timer active out of how many are online.

The mutter text will update to the global timer if there is a global timer active.

The mutter icon turns red after a global timer ends and an alarm sounds.

Table of Contents

[1.0] Game Jam Checklist

These are suggestions and may help as a sanity check if not used as an actual checklist.

[1.1] Computer/laptop access

1. Program installations (Game Creation tools, IDEs, Audio, Graphics)
2. Extensions, USB drives/storage, headphones, adapters, cables, fans
3. Tablets, microphones, extra equipment like controllers
4. Mobile devices, tablets or consoles as desired
5. Either your laptop's webcam or mobile phone for photos

[1.2] Logistics

1. Place to work
    a. Wireless internet access
    b. Whiteboard access (or your own, or a large sketchbook)
    c. Where to keep things safe
    d. Access to printers and scanners (for posters or assets)
2. Place to rest
    a. Sleeping materials (pillows, blanket, ear protection)
    b. Toiletries (toothbrush, toothpaste, towel, deodorant)
    c. Extra clothing
3. Food
    a. Immediate snacks & drinks
    b. Water, fruit
    c. Know places to get food
4. Where/How to get help
    a. Contact information, maps, support
    b. Know the schedule/activities

[1.3] Ideation/Prototyping kit (optional yet recommended)

1. Cardstock, cardboard
2. Papers, post-its
3. Drawing materials
4. Scissors, cutting materials
5. Tape, paper & universal glue
6. Dice
7. Playing cards
8. Tokens if desired, otherwise be creative

[1.4] Game specifics

1. (Recommended) A minimum resolution of 1024x768
2. Player controls, mechanics and gameplay work
3. Game challenge and game flow are balanced
4. Menus for starting and ending the game, and a help screen if needed
5. If the game ends, you should be able to easily start/retry the game
6. Audio (especially music) add to the game and presentation experience
7. External user testing or feedback has been conducted

[1.5] The deliverable (optimally a zipped file for ease of download)

1. A readme with a short description, credits/contacts, game help, links to dependencies
2. Executables and libraries
3. Source files
4. Screenshots (3 are optimal)
5. (Optional) Presentation, Gameplay Video, Gamelog, Design document, Documentation

[2.0] Physical voting methods

These are suggestions and may be used to inspire other methods.

[2.1] Pulled from a hat

Participants write their votes or topics into a hat, which later is shaken up and drawn from.

- ➔ May be used to decide on an idea randomly
- ➔ May have multiple hats which represent the ideas, and can place one vote per participant into them
- ➔ May be used to decide role or task assignments

[2.2] Drawing straws

Varying lengths of paper, sticks, or straws are drawn from a person's hand

[2.3] Rock, paper, scissors

A hand game that decides a winner through a number of rounds

[2.4] Flipping a coin, rolling dice, guessing closest to a written number

- ➔ May be used to decide on an unwanted/desired task
- ➔ May be used to pick out a participant for a task

[2.5] Ball throwing

Participants take turns passing the ball to each other once before getting a second turn, where the ball allows speech

- ➔ Suitable for situations where many opinions and votes have to be heard with less criticism

[2.6] Evaluation tables, pros/cons list

- ➔ Suitable for situations where a whiteboard is available and a complicated decision needs to be voted for
- ➔ Evaluation tables may be simplified so that certain features/suggestions may be evaluated as positive (+), neutral (o) or negative (-)

[3.0] Game Jam Practical Troubleshooting tips

A quick rundown of things to look out for during the course of the game jam.

[3.1] Division of different tasks

(not to be confused as roles, everyone may contribute)

- ➔ Coding/programming/version control
- ➔ Artwork/assets including character design
- ➔ Game Mechanics
- ➔ Level/map Design
- ➔ Music/sound and audio
- ➔ Quality Assurance/feedback and user testing
- ➔ Documentation/copywriter

[3.2] When stuck or struggling

1. Check if there are other priorities, the detail may be minor and disregarded.
2. Reassess if making it technically work is worth it, and how much time it will take.
3. Attempt to divide the problem into smaller ones; among team members.
4. Look for external feedback and accept critique to change the design.
5. Look for creative alternative, easier solutions and playtest them.
6. Take a short break and revisit the issue later. Napping, sleeping or eating may help.

[3.3] Recommended habits

1. A shared folder to collaborate together on.
2. A form of version control (Mercurial, or a script, or manually duplicating the project).
3. Put up a schedule where everyone can easily see it. Preferably on a whiteboard.
4. Shared working area, and a quiet (and dark) isolated area.
5. Save early and often, playtest early and often. Take breaks early and often.
6. Rotate working hours so the work process and flow is more natural (example: when one coder sleeps another can take their place to avoid the need to merge code).
7. Assets and graphics rotate working hours only to keep sanity and sleep deprivation at bay.
8. Keep a positive vibe and remember the time dedicated to this is for fun.

[3.4] Quick overview checklist

1. (Philosophical) What defines a game?
2. Do core mechanics work?
3. Does it relate to or challenge the theme?
4. Is this a game of quality?
5. Is the gameplay fun?
6. Does it have good game flow, challenge and feel?
7. Is the game compelling or intriguing?
8. Has there been playtesting in at least three different sessions?
    a. How long does gameplay last?
    b. Is it understandable, can players get it?
9. Does the game have music or audio?
10. Have you added juice?