

Gaia UX **System**

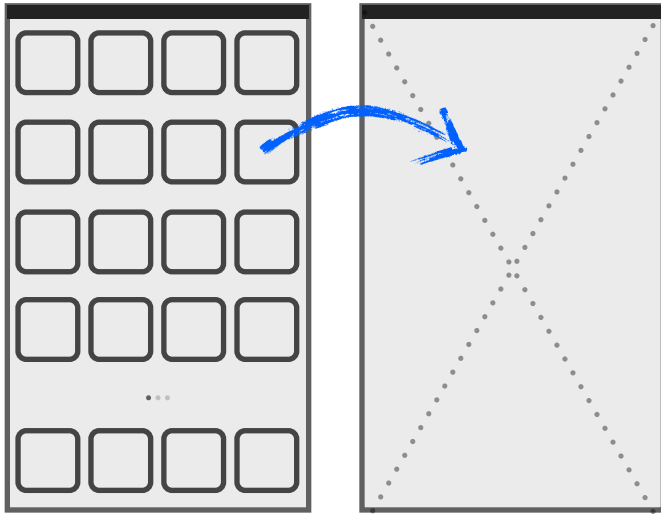
April 23, 2012

Home Screen & Task Switching Concept

Home Screen Concept

Can we marry the “Cards” home screen approach of webOS and RIM Playbook with the ease of use of iOS’s Grid view?

iOS

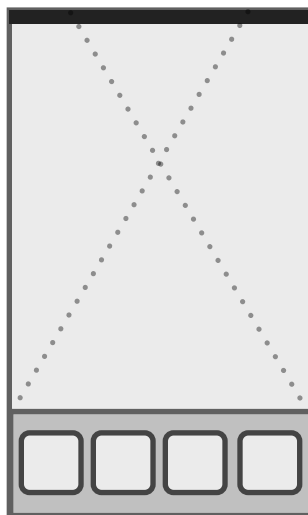


1

2

Two primary levels:

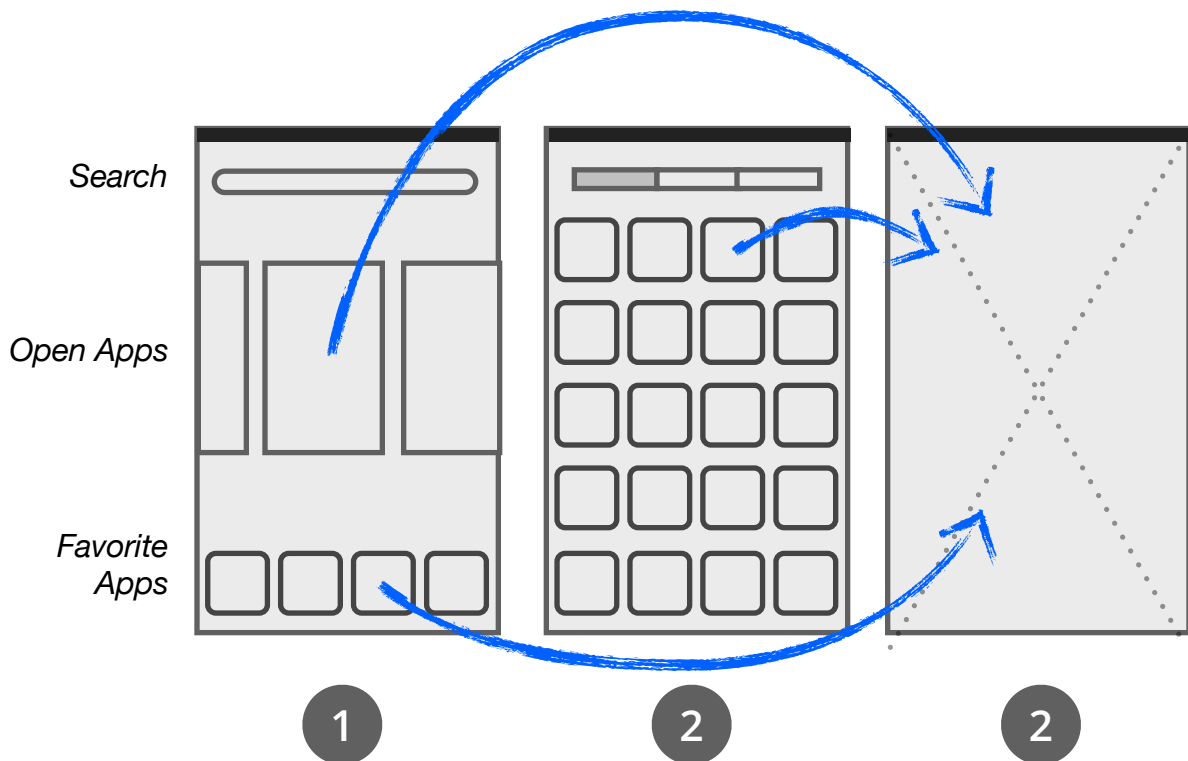
- Home screen grid (1)
- Full screen app (2)



3

There *is* a third level (3) for lateral (app to app) task switching, but it was a late addition (iOS 4), and unnecessary in day-to-day use. You never *need* to use it. Large majority of iOS users probably don't it exists (it's hidden behind a home-button double press).

Playbook / webOS



Three primary levels

- Task switching (“Card” view) (1)
- Apps browsing (“Grid” view) (2)
- Full screen app (3)

(1) presents open apps as “cards” showing screen captures of their current states, and limits the available apps to 4-5 favorites in a tray along the bottom of the screen. User can manipulate the open apps by dragging, throwing off-screen, etc.

To access the full list of available apps, the user must click or swipe to access a separate Grid view (2).

Pressing “home” button from open app returns user to the Cards view.

Pros and Cons

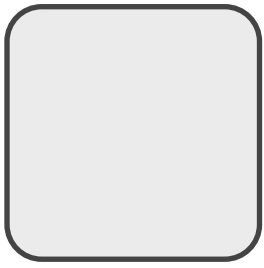
RIM and webOS models present a steeper learning curve than iOS simple grid of apps. Users live in an app-manipulation mode that is fun and effective for some purposes (eg: task switching), but opaque to first time users. It's not clear, for example, where apps live. With webOS the user must discover the Launcher icon (lower right), or with Playbook, swipe up from the bottom bezel.

iOS, on the other hand, with it's elegant Grid/App dichotomy, is immediately intuitive for everyone from a toddler to a senior.

With Gaia, can we combine the best of both approaches? Incorporate a card-based task switching view, but tweak it slightly to ease new users into model, and make it easier to find the Grid of apps?

Here are some ideas:

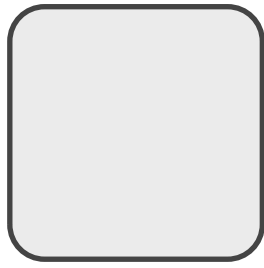
1. **Present the Grid first.**
2. Once user selects app, use a **transition animation** to introduce users to the Cards view, briefly showing it fly by.
3. Once user **exits app** (presses Home hardware button), take them to **Cards view**.
4. If users want to return to Grid view, the transition animations have taught them where it is, and they can **press or swipe** to get there.



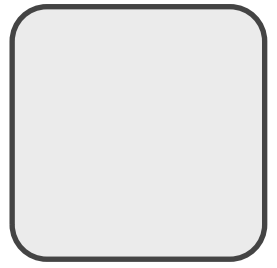
Phone



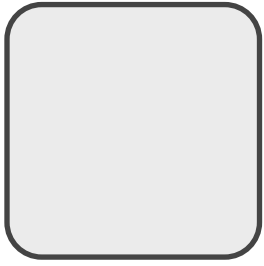
Messages



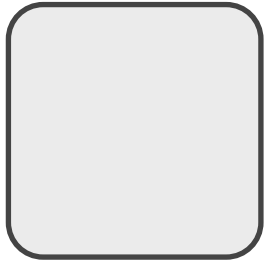
Firefox



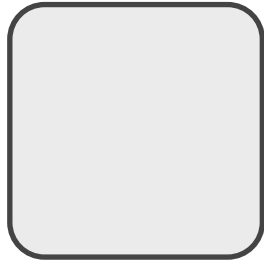
Firefox



Marketplace



Contacts

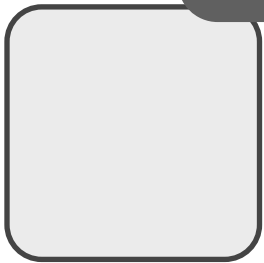


Videos

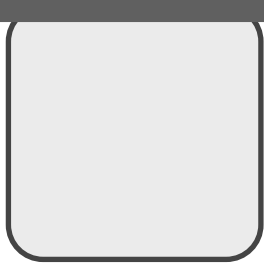
Settings



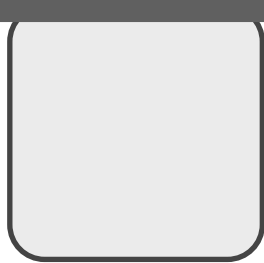
On first-run, the user starts in the Grid view. It is familiar and easily understood. We don't make the mistake of webOS and Playbook (dropping the new user into the unfamiliar and empty Cards view).



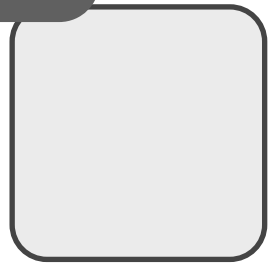
Videos



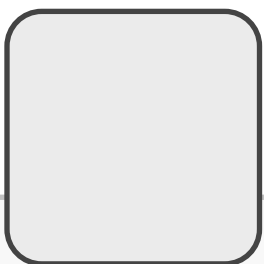
Pictures



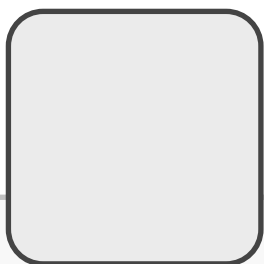
Settings



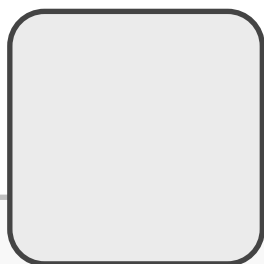
Settings



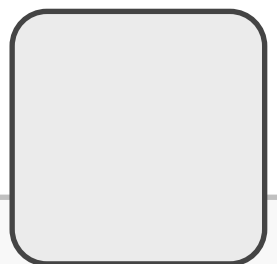
Videos



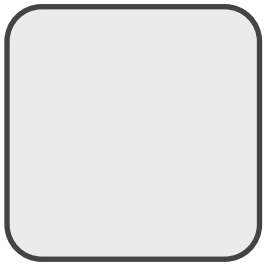
Pictures



Settings



Settings

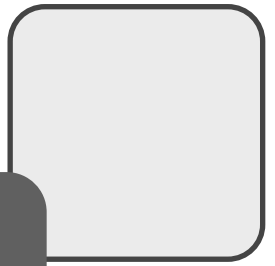
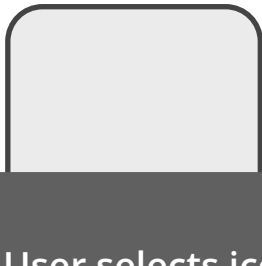


Phone

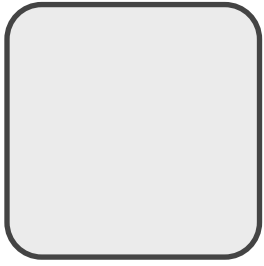


Messages

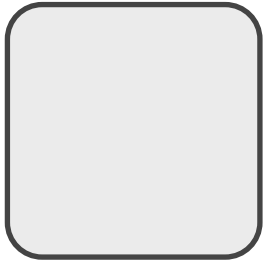
User selects icon



Firefox



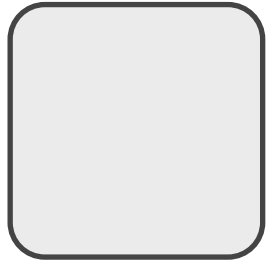
Marketplace



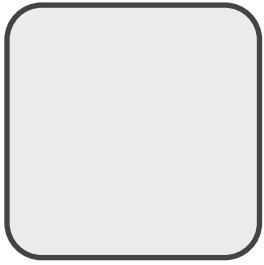
Facebook



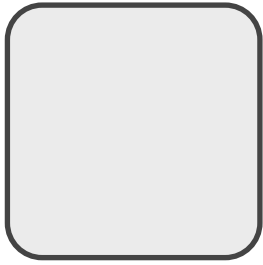
Contacts



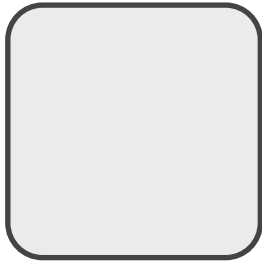
Contacts



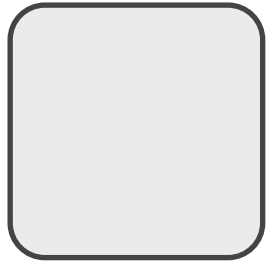
Videos



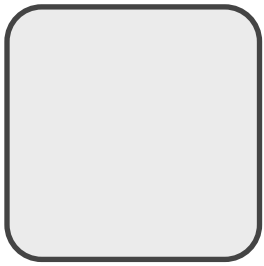
Pictures



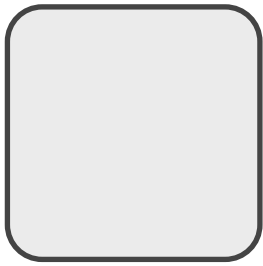
Settings



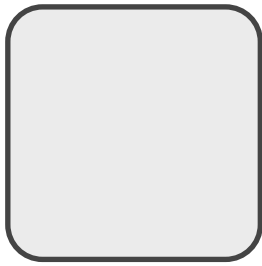
Settings



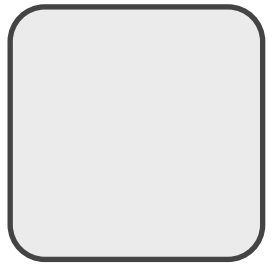
Videos



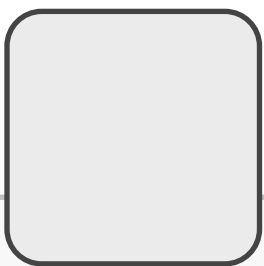
Pictures



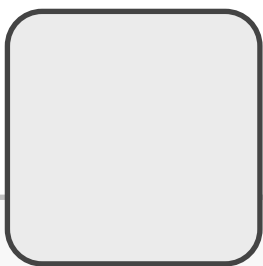
Settings



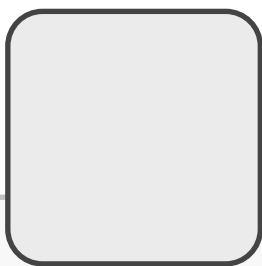
Settings



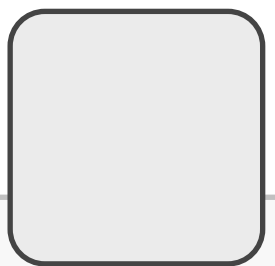
Videos



Pictures

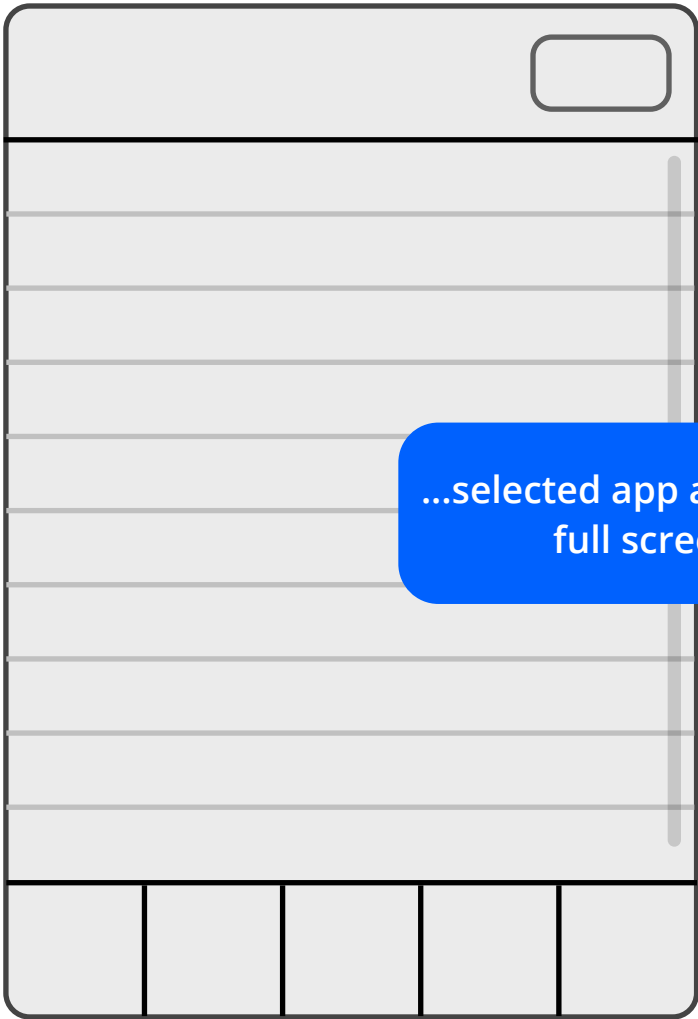


Settings

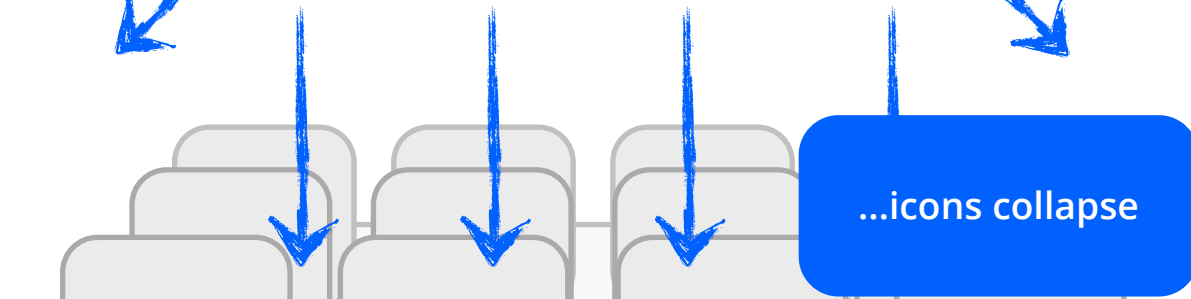


Settings

Animated transition...



...selected app animates to full screen...



...icons collapse

The transition hints at the rules of the interface, and indicates where apps collapse to.

Videos

Pictures

Settings

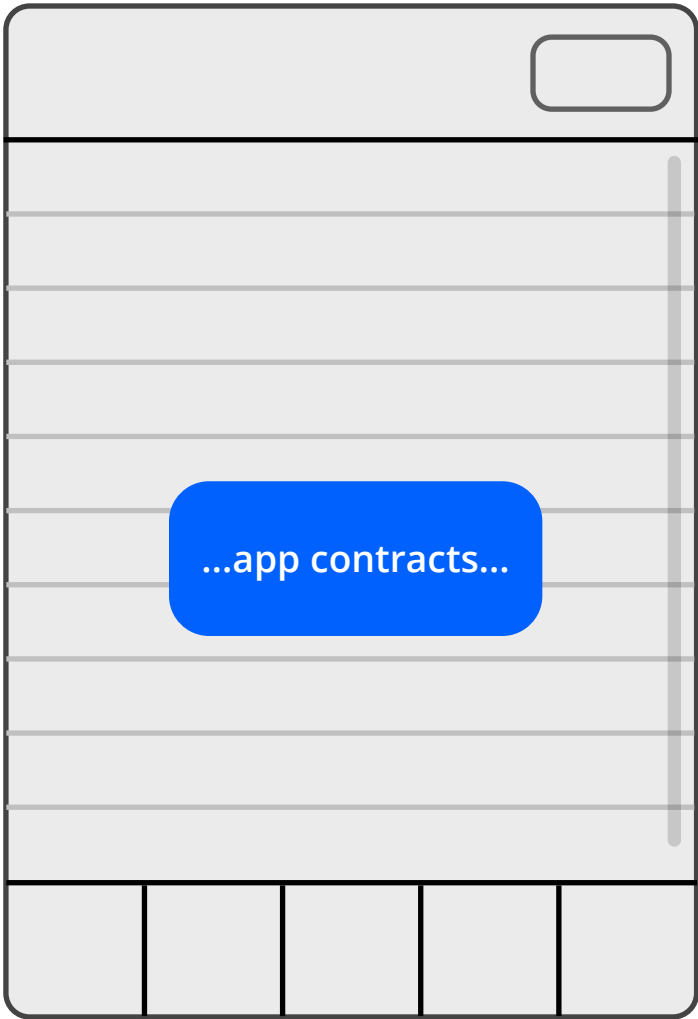
Settings



User uses app for a few minutes,
then wants to switch.

Presses hardware home button.

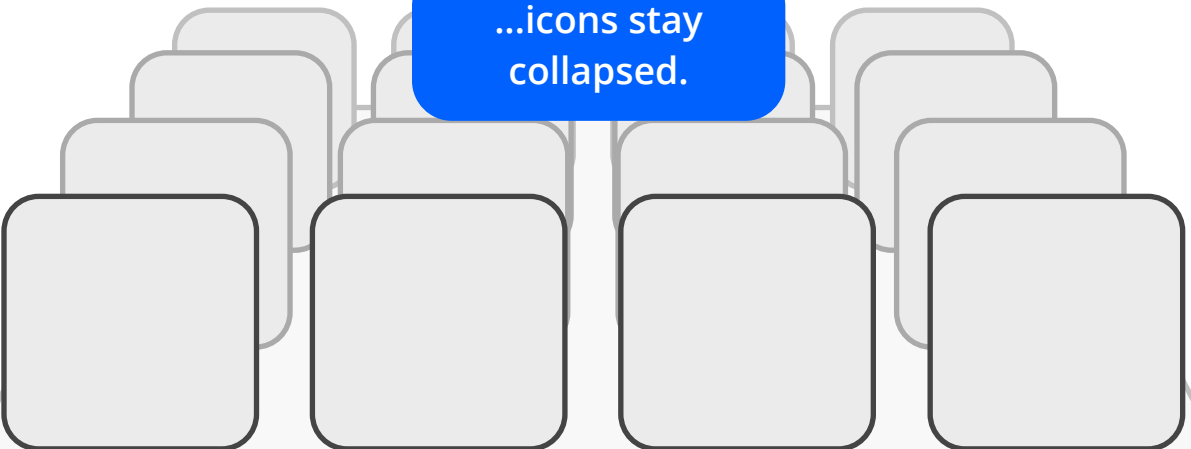
Animated transition...



...app contracts...



...icons stay collapsed.



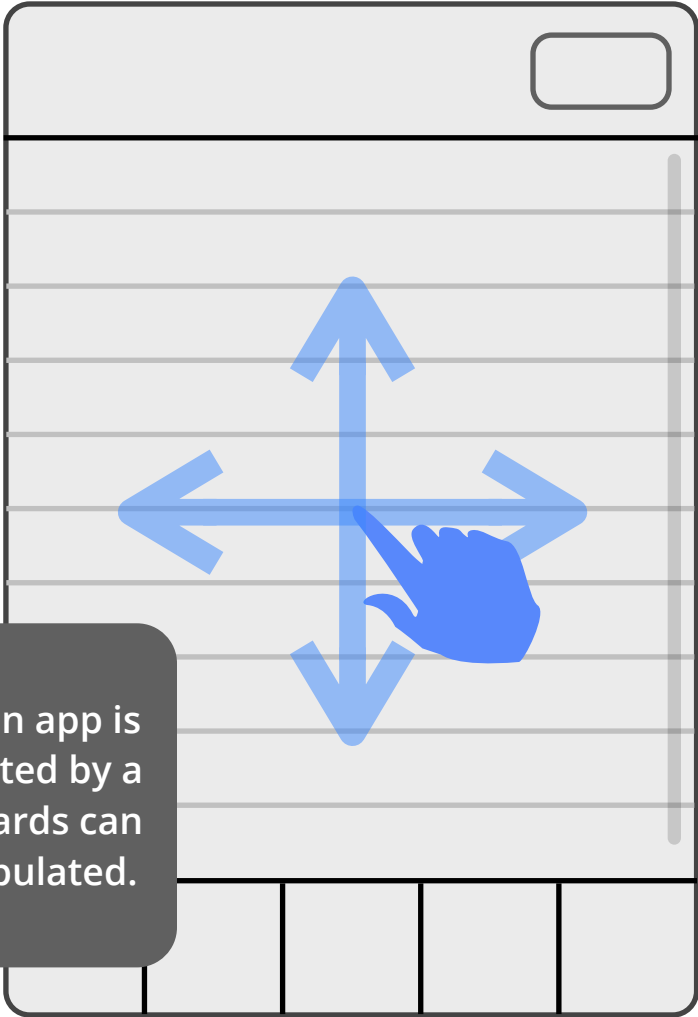
Videos

Pictures

Settings

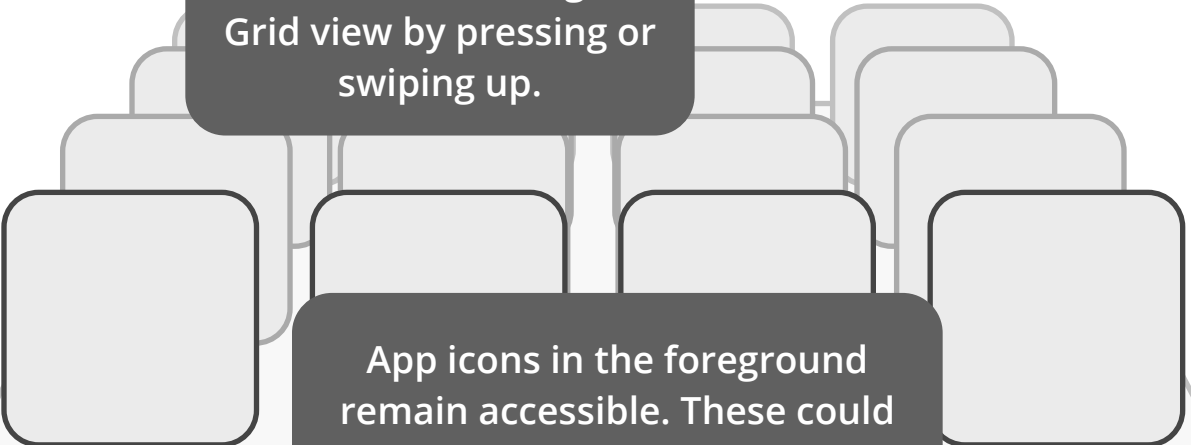
Settings

We're in our "Task Management" view



Each open app is represented by a "card". Cards can be manipulated.

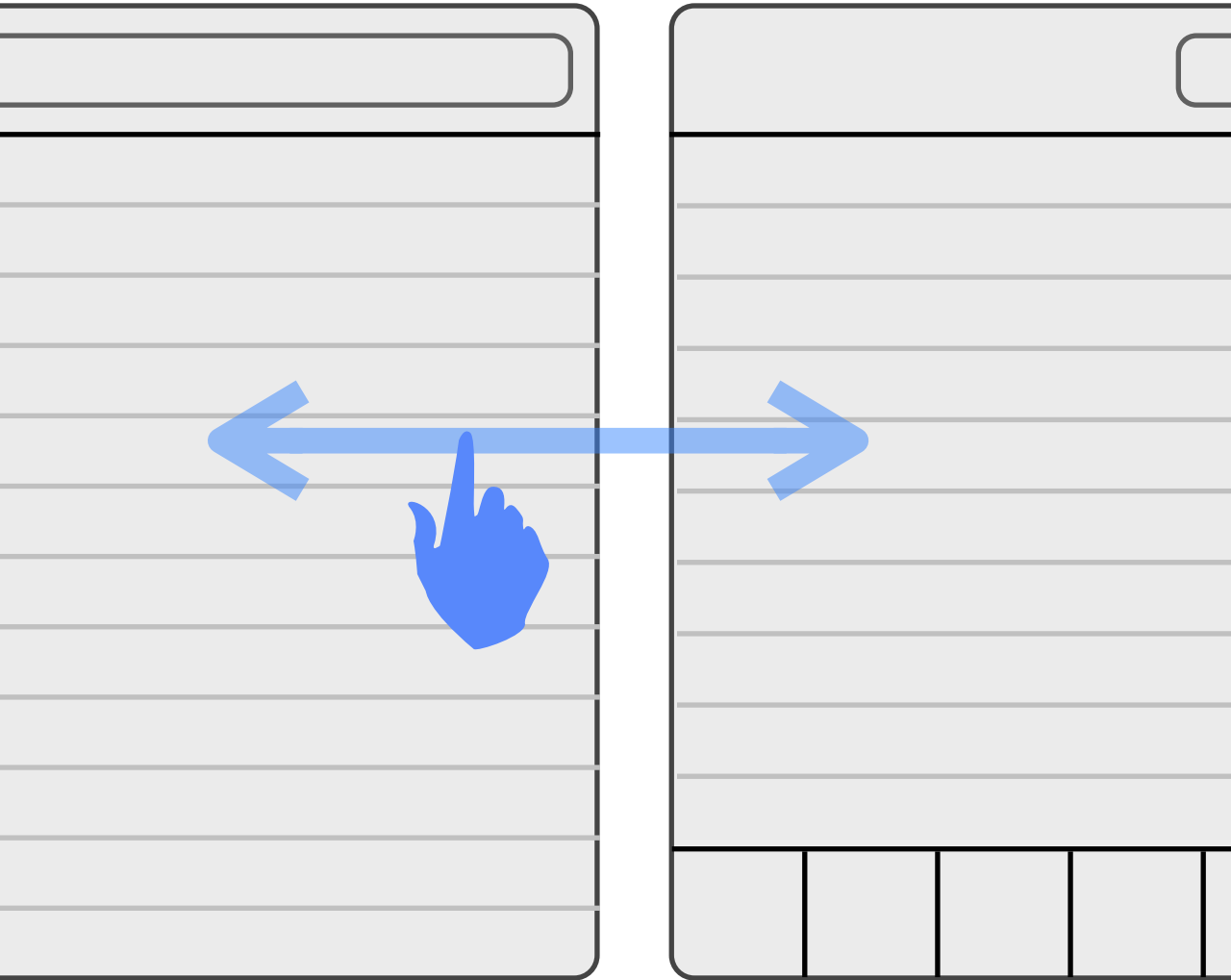
Apps remain collapsed. User can returning to Grid view by pressing or swiping up.



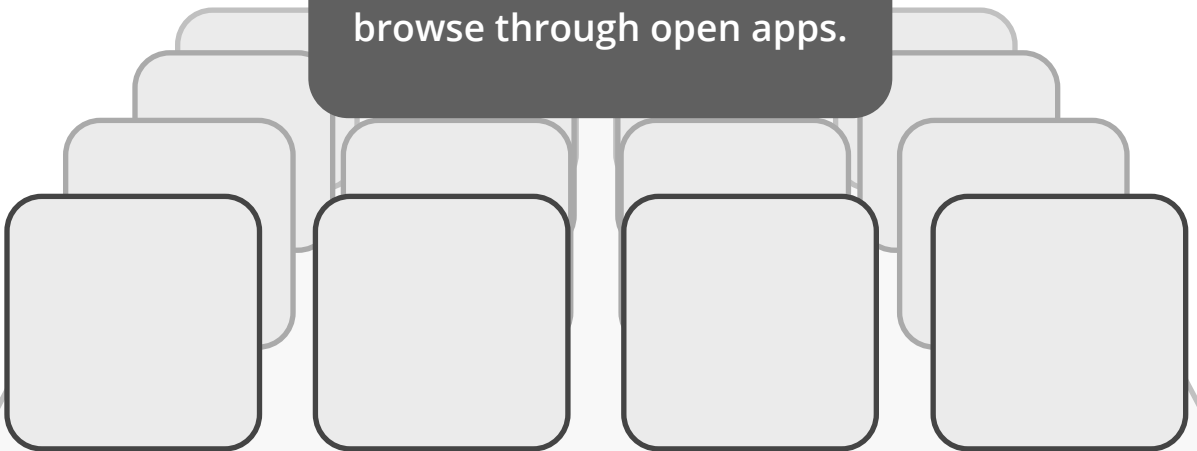
App icons in the foreground remain accessible. These could be favorites.

Videos

Settings



Swiping left and right to browse through open apps.

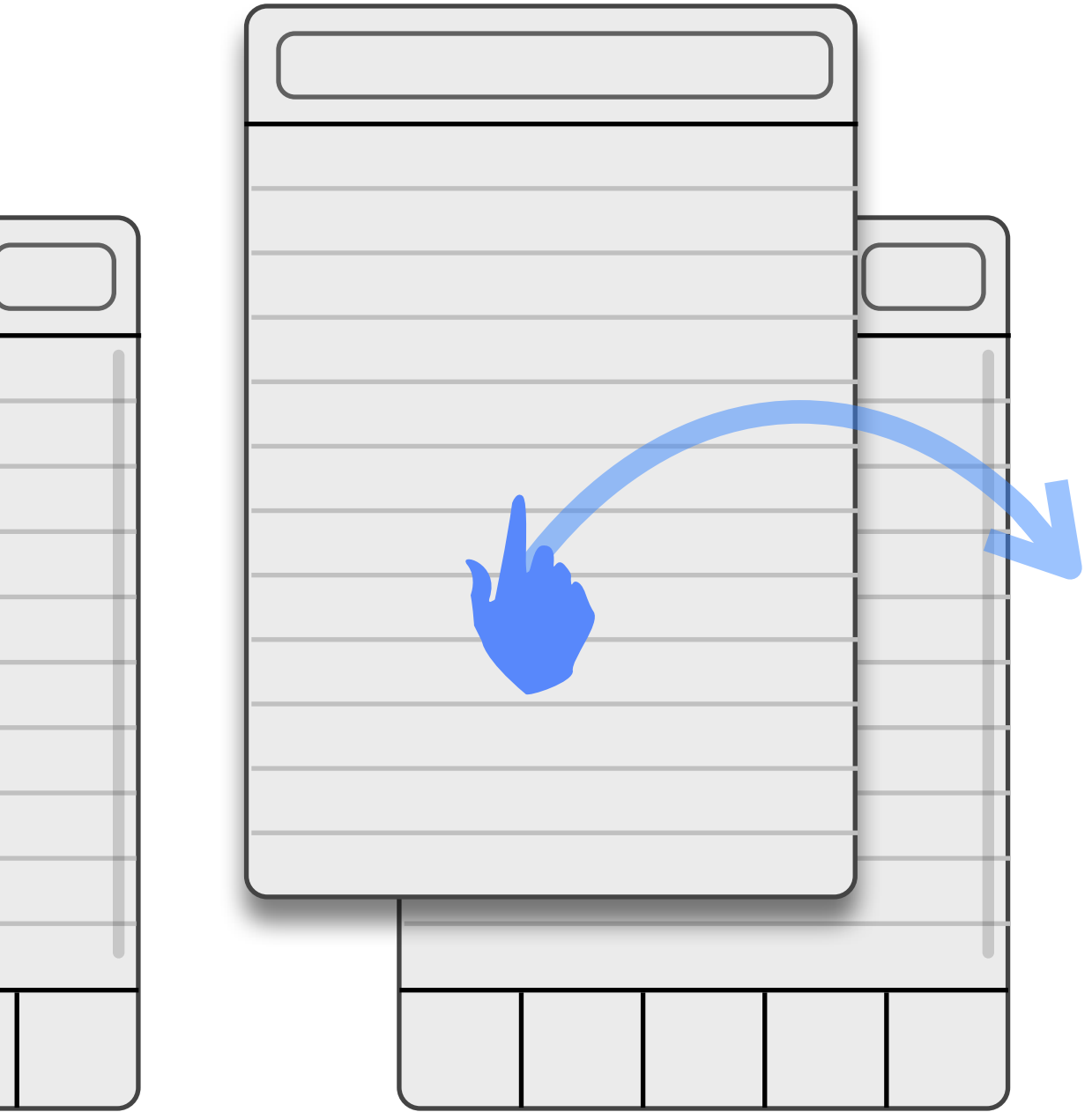


Videos

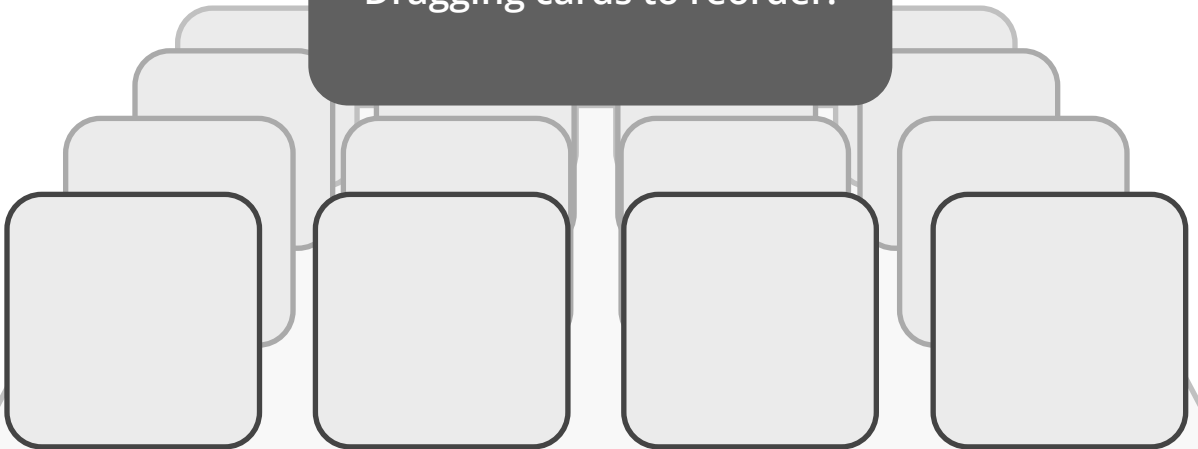
Pictures

Settings

Settings



Dragging cards to reorder.

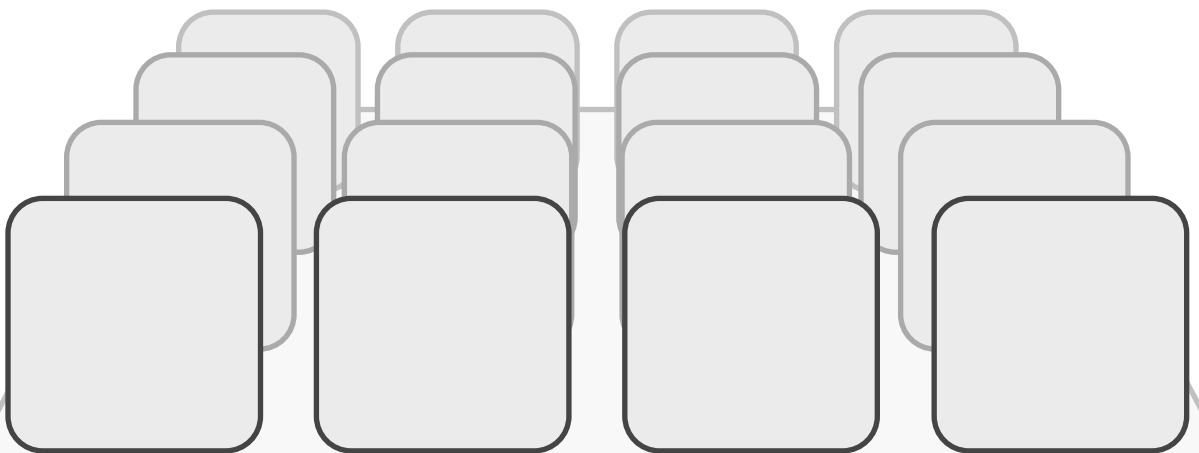
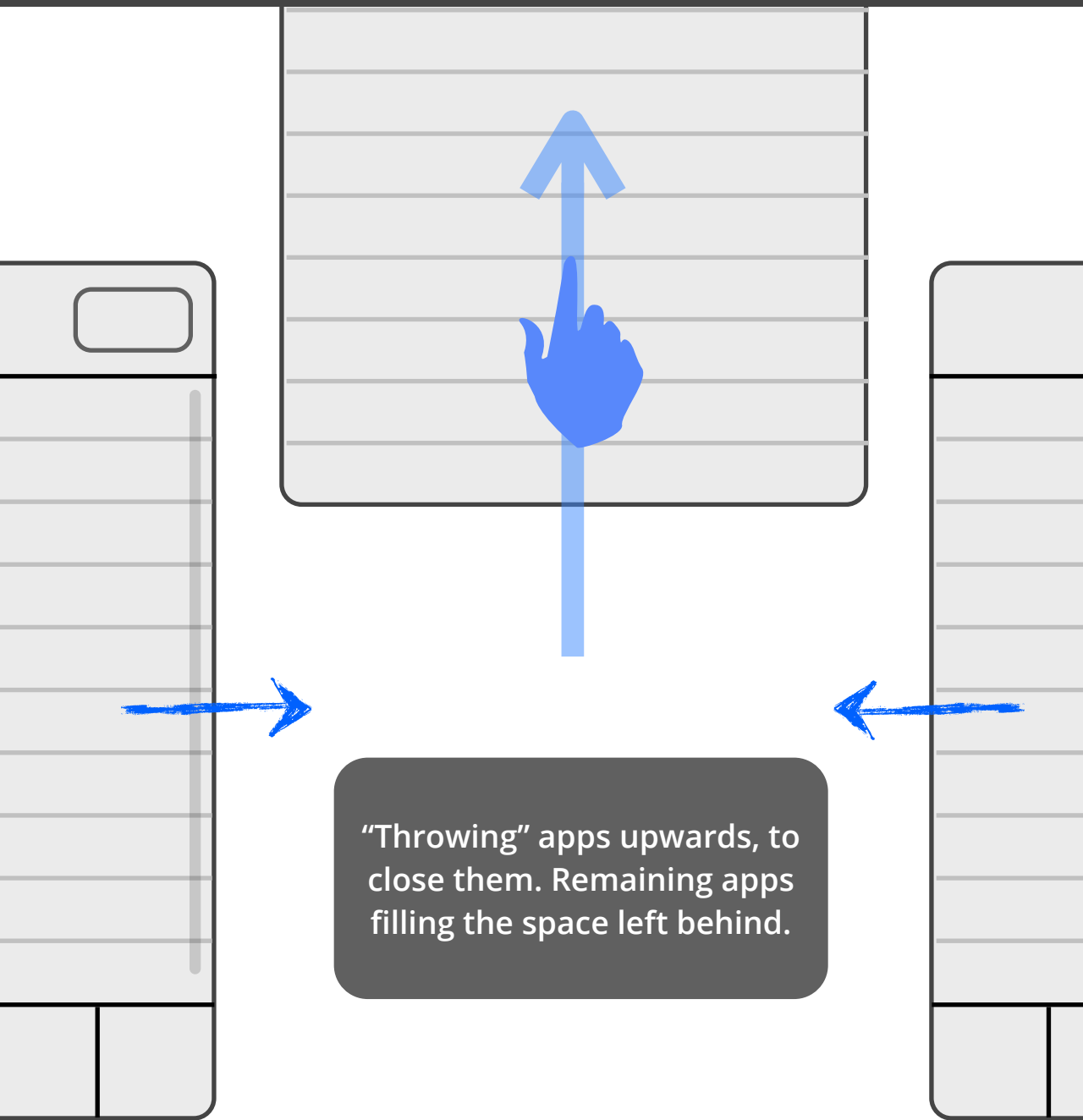


Videos

Pictures

Settings

Settings



Videos

Pictures

Settings

Settings



...or pressing the collapsed icons.

User can return to Grid view by swiping up from bottom bezel...

Videos

Pictures

Settings

Settings

Animated transition...

...cards recede from view
(possibly by flying off-screen,
or fading out, etc)...

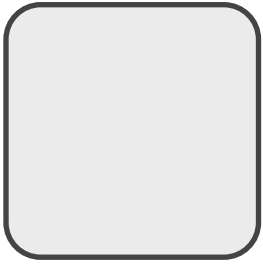
...icons expand.

Videos

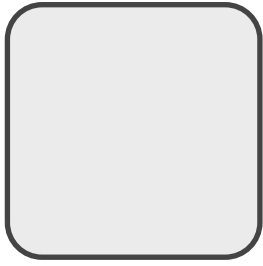
Pictures

Settings

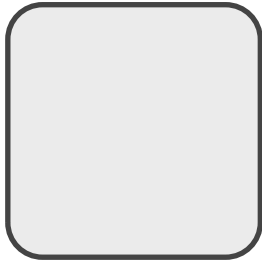
Settings



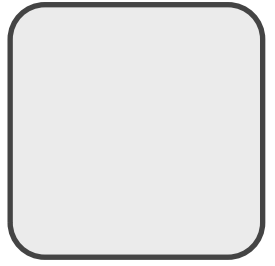
Phone



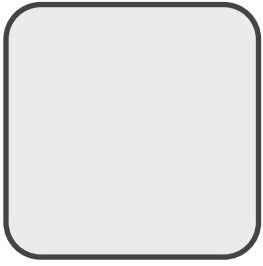
Messages



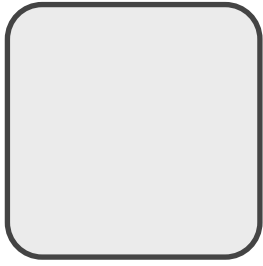
Firefox



Firefox



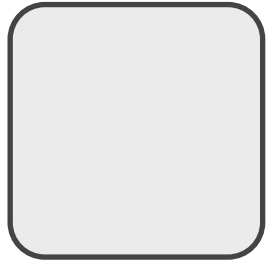
Marketplace



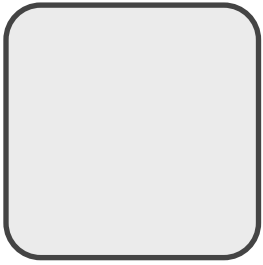
Facebook



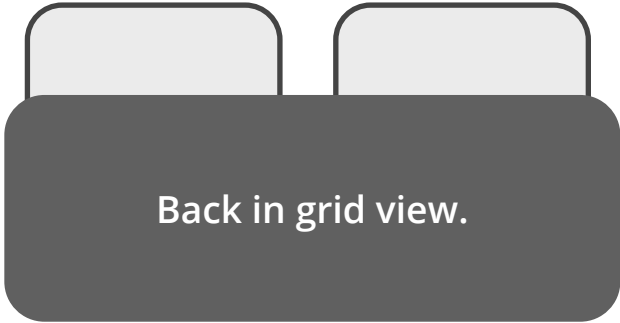
Contacts



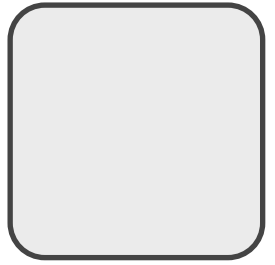
Contacts



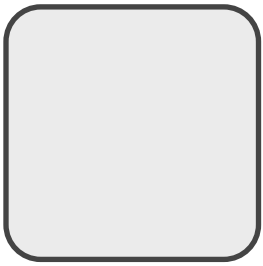
Videos



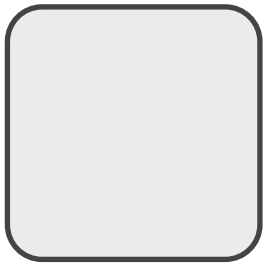
Back in grid view.



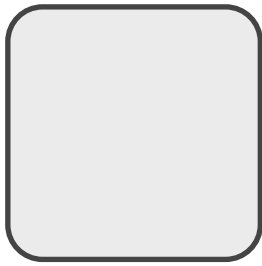
Settings



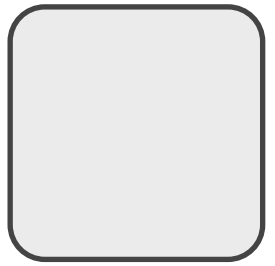
Videos



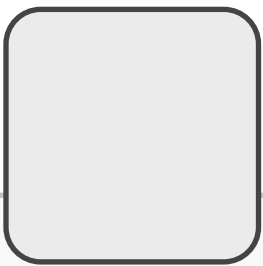
Pictures



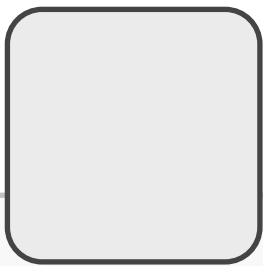
Settings



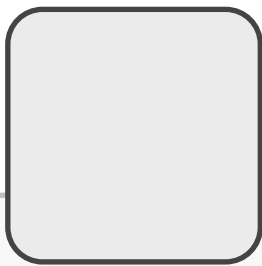
Settings



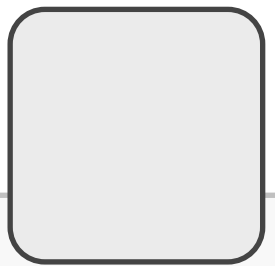
Videos



Pictures



Settings



Settings

Slight alternative

In the preceding model, the user jumps between Card view and App view, and only occasionally pops into Grid view.

Alternatively, we could favor the iOS / Android approach: the user jumps between Grid view and App view, and only occasionally pops into Card view (to switch tasks).

If we accept that the Grid is more intuitive to the average user, then promoting it over the Card view would be a win for usability. This approach would also enable a staggered release. We could release 1.0 with just Grid and App views, and then add the nice-to-have Card view in later versions, without breaking any UX rules. This was the iOS approach. Task switching only arrived in iOS 4.0

About Search...

A quick note on **search**, which isn't referenced in the preceding. Every mobile OS includes some form of system-level search. We're undecided on UX for Gaia, but are exploring some different options:

iOS: tucked into a dedicated home screen panel to the far left

Android / webOS: included on home screen main panel, at the top.