

Snapsible

**Data Visualization exploration
for Foldable Screen displays.**

**Mehul Shah
Olga Khvan**

Overview



Fig 1: Snapsible concept

Snapsible provides effective communication of complex quantitative ideas, information for the Foldable Screen displays. Just as Microsoft Surface Duo/Samsung Galaxy Fold can provide a larger screen real estate for more possibilities and interactions, Snapsible visually displays measured quantities by means of the combined use of points, lines, a coordinate system, numbers, symbols, words, shadings, and colors [1] with foldable interaction capabilities. Snapsible makes this visualization more compelling by seeking advantage of the dual-screen displays. Moreover, an extended set of gestures helps to create a more intuitive and easy way of visualized data exploration.

This system addresses an ongoing need for professionals that work on the move and try to get more done. This need is ever-growing for professionals - such as engineers, academicians, product managers, executive officers, and working professionals.

Our target audience is users—mainly working professionals that have access to dual-screen displays and who have a need for interacting with information and visualizations. Data visualization has become an inherent part of our daily lives - whether it is viewing the latest information on a map, current company stock price, or the current site-analytics. As people shift to smartphones as their primary source of information, we are faced with exploring new visuals and interfaces that are optimized for small screens and constrained input modalities. [2] Still the size of a regular smartphone screen and the existing gestures are not enough to fully explore the presented data, especially, when a user wants to compare multiple graphs or communicate selection between multiple graphs. We propose foldable displays as one way to solve this problem.

It is important to notice that Snapsible's main goal—make users interact with data in a more intuitive and easy way. So Snapsible does not provide them with any tools for making visualizations or processing the data.

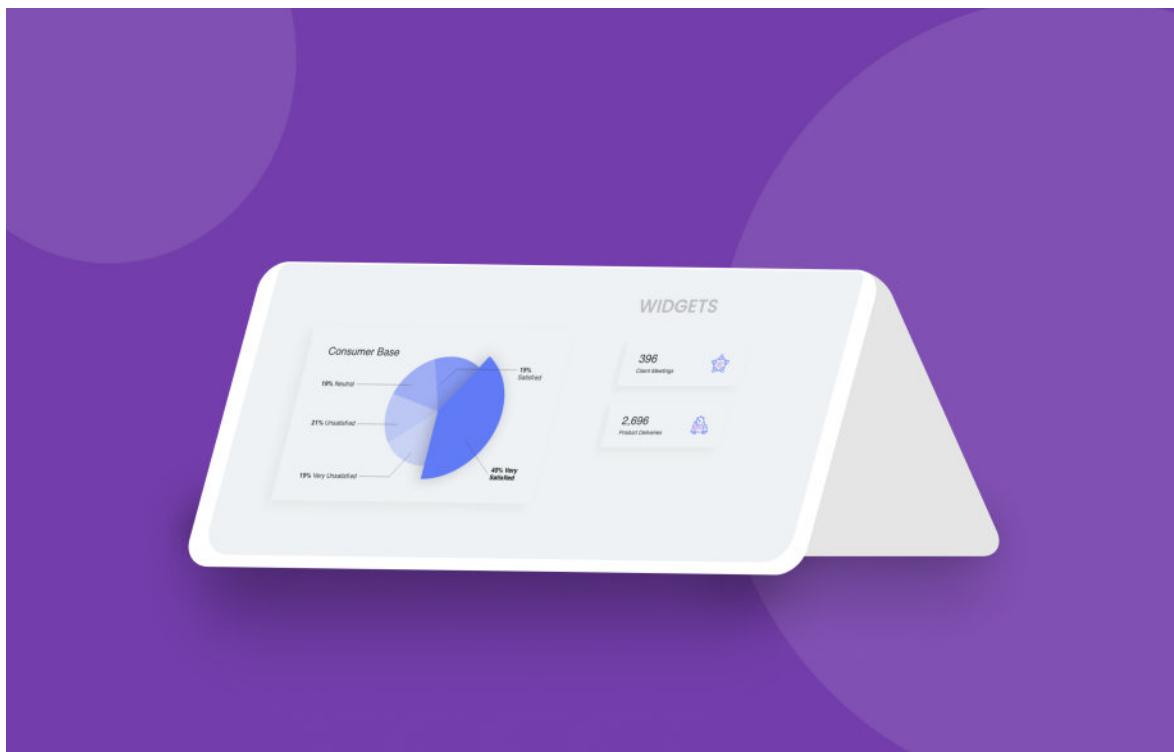


Fig 2: Snapsible **[dock mode]** that allows widgets to interact with another person with two rear screens on a single device.

Experience Design Details

User Interface

Snapable leverages the dual-screen smartphone displays which have the capability of folding into two equal halves. This interface allows users to interact with either one of their rear screens or the entire unfolded display. This device can be used in one hand much like how they currently use their smartphones when it's in the folded state and with both their hands when they seek to perform more actions by unfolding the display to gain a larger screen real-estate. These multiple states of the display allow users to interact with visualized data in different ways. Each state supports certain types of interaction with both graphs and other users.



Fig 3 and 4: Folded and Unfolded states of the device

Characteristics of effective graphical displays

Before describing certain Snapsible interactions and gestures, it is necessary to understand the main qualities of effective graphical displays. Professor Edward Tufte explained that users of information displays are executing particular analytical tasks such as making comparisons. The design principle of the information graphic should support the analytical task.[1] In his 1983 book *The Visual Display of Quantitative Information*, Edward Tufte defines 'graphical displays' and principles for effective graphical display in the following passage: "Excellence in statistical graphics consists of complex ideas communicated with clarity, precision, and efficiency. Graphical displays should:

- Show the data
- Make the viewer think and focus on the substance, rather than on methodology, graphic design or technology
- Avoid distorting what data really reflects
- Present a lot of numbers in a limited space
- Make even large data sets coherent
- Encourage exploration and comparison of different data pieces
- Reveal the data at numerous levels of details, give multiple view perspective
- Have fine and structure
- Serve a clear and reasonable purpose (description, exploration, decoration, etc.)
- Integrate closely with both statistical and verbal description of data

Graphics reveal data. Indeed graphics can be more precise and revealing than conventional statistical computations."[1] So these characteristics were taken into account while designing Snapsible.

Device Orientations

Unfolding the device to provide a larger screen can have a positive impact on users:

- A larger screen real-estate usually means a more immersive experience.
- With multi-windows the user can perform multiple tasks at the same time.
- With docking mode users can interact with another user with two rear screens.
- Folding and unfolding can change the screen size, density, or ratio.

This is not a new problem and already happens in these non-folding cases:

- Phones: switching between portrait and landscape mode.
- Chrome OS running in desktop mode: resizing Android apps.
- Devices with multiple or additional screens.

Continuity



When exploring data on a foldable device, Snapsible can transition from one screen to another automatically.[3] To provide great user experience, it is very important that the current task continues seamlessly after the transition and resumes in the same state and location. Following are the different states a foldable device can fold, such as in or out:

Fig 5: Folding states of the device

New Screen Ratios

Android 10 (API level 29) and higher support a wider range of aspect ratios.[3] With Snapsible, form factors can vary from super high long and thin screens (such as 21:9 for a folded device) all the way down to 1:1.

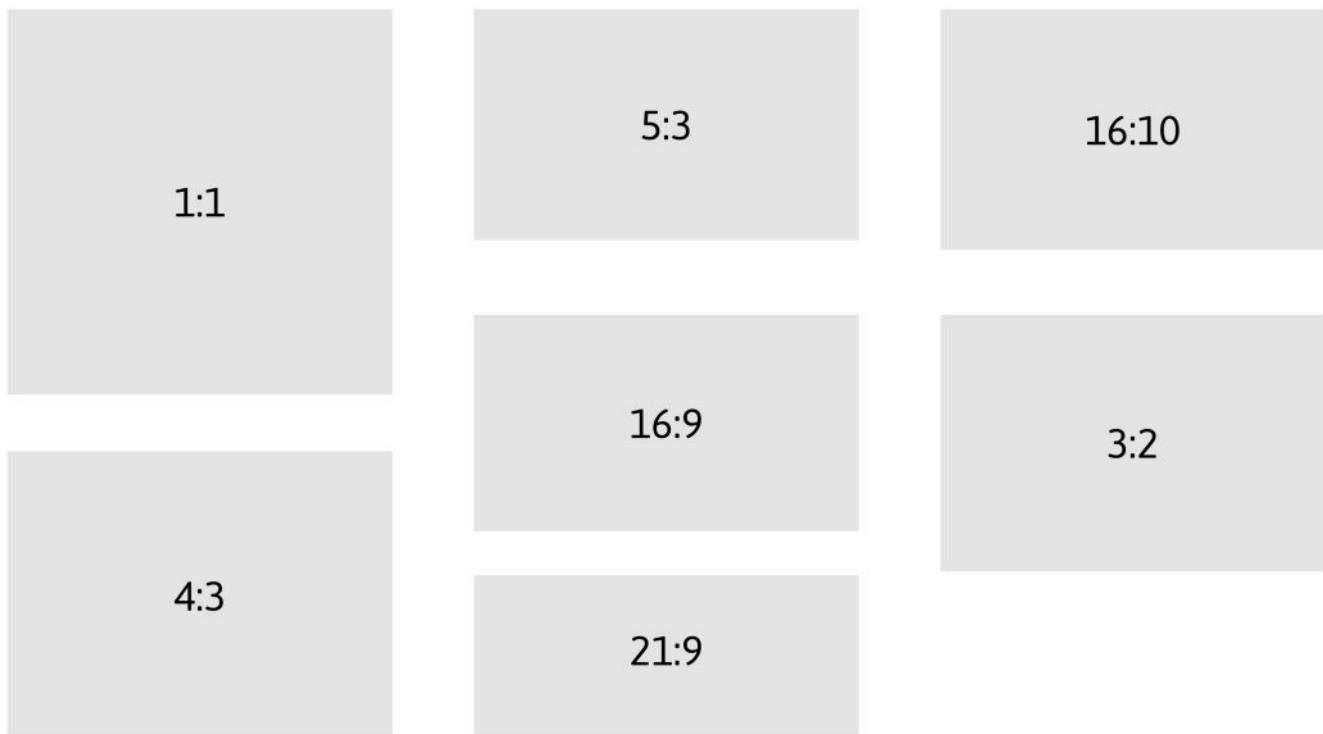


Fig 6: Device screen ratios

Multi-window

Being able to run multiple windows is one of the benefits of larger screens. In the past, having two apps side by side was common in some devices. With the advancements in technology; three or more apps can run on the screen at the same time, and also share content between themselves:

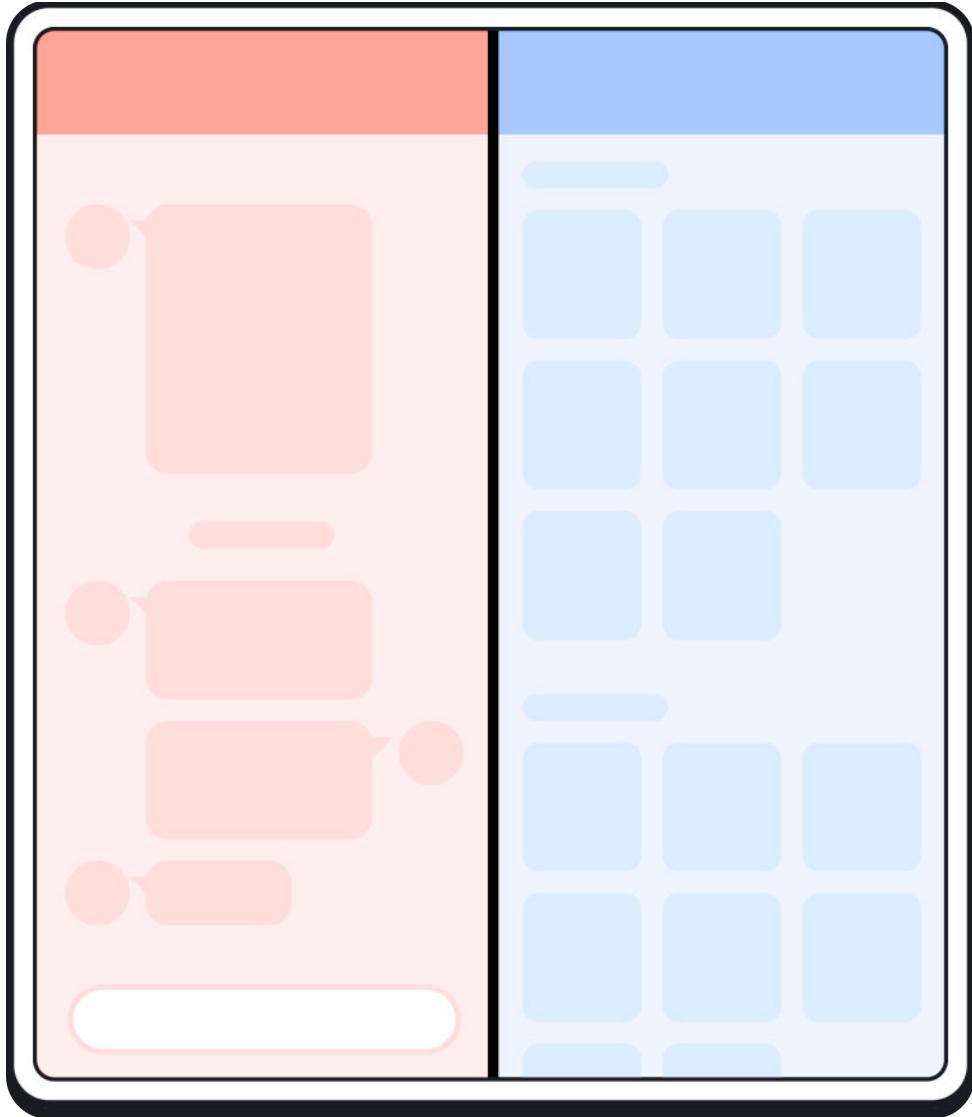


Fig 7: Multi-window screen

Resizing Windows

Snapsible ensures that it works in multi-window mode and dynamic resizing.[3] This is achieved by setting `resizeableActivity=true` which provides maximum compatibility with whatever form factors and environments it might encounter (like foldables, desktop mode, or freeform windows).

With AOSP (Android Open Source Project) emulator providing support for folding devices. To provide the best experience we test Snapsible in their specified folding scenarios.[3] Such as the mentioned 7.3" and 8" emulators below:

7.3" emulator

7.3"	Display	Resolution		Logical Display	
	Size	X	Y	DensityDpi	Size
Unfolded	7.3	1536	2152	420	Large
Folded	4.6	840	1960	420	Normal

8" emulator

8"	Display	Resolution		Logical Display	
	Size	X	Y	DensityDpi	Size
Unfolded	8.03	2200	2480	420	Large
Folded	6.62	1148	2480	420	Normal

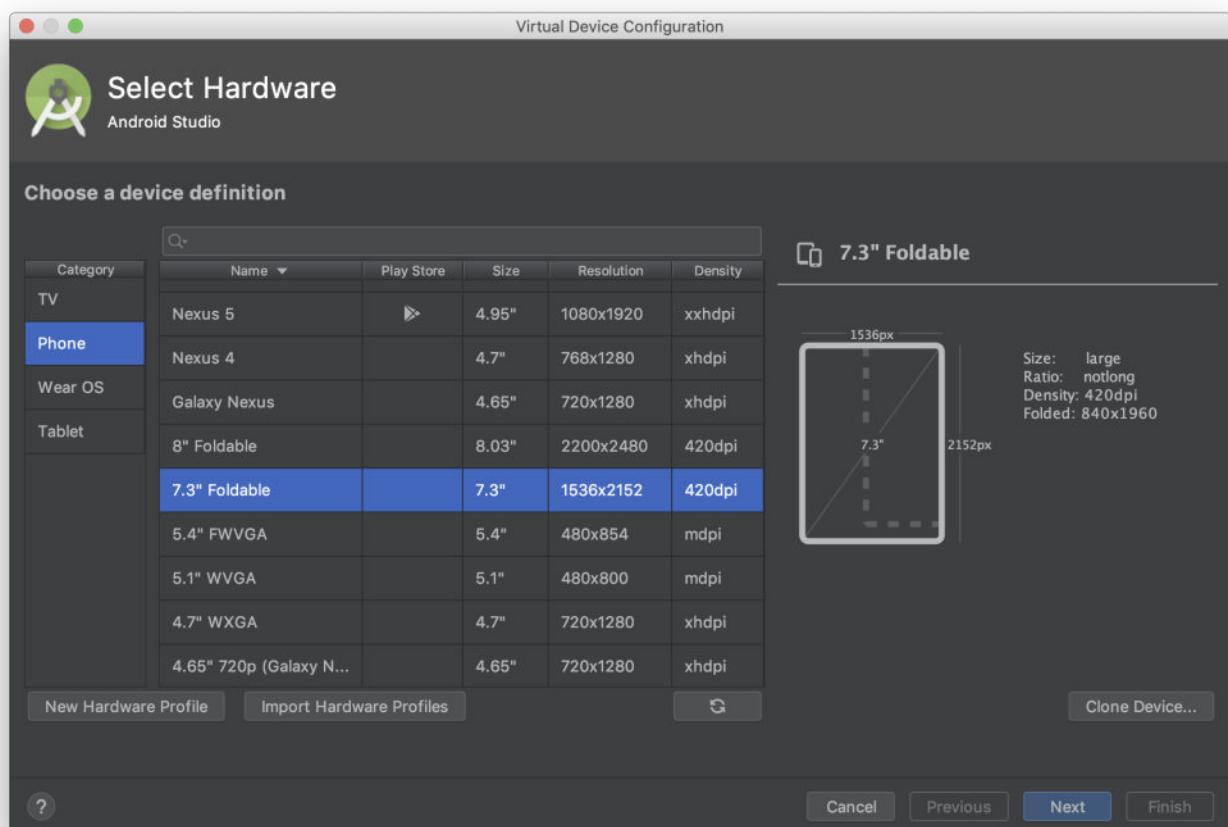


Fig 8: Screenshot of Android Virtual Device selection on Android Studio

If we set **resizeableActivity=false**, this tells the platform it doesn't support multi-window. Then Snapsible may still resize your app or put it in multi-window mode, but compatibility is implemented by applying the same configuration to all the components in the app (including all of its Activities, Services, and more). In some cases, major changes (like a display size change) might restart the process rather than change the configuration.

Once **resizeableActivity=false** is set along with a MaxAspectRatio and the device running Snapsible is unfolded, the activity configuration, size, and aspect ratio are maintained by putting Snapsible in compatibility mode.

If we do not set `resizeableActivity`, or set it to true, the system assumes Snapsible fully supports multi-window and is resizable.

Interaction Modes

Description

Interaction for visualization is characterized as a goal-oriented activity that contains semantics, that is sequential, incremental, and iterative, and preserves the following properties[8]:

Granularity of interaction

Micro-level: screen touch,
Macro-level: hypothesis generation
In-between: sort, filter

Continuous vs. Discrete

Continuous interactions involve a sequence of intermediate visualization states between an initial state and a goal state
Discrete interactions mean that action and reaction occur in a distinct manner

Direct vs Indirect	Direct interaction with data itself
	Interaction with data using widgets or queries as a medium

Granularity-wise, Snapsible provides users with in-between and micro-levels of interactions that can be both continuous/discrete and direct/indirect. Whereas this categorization provides a more high-level understanding of interaction types for Snapsible, there is also a more specific interaction taxonomy based on one of the data-visualization tools, Vega-Lite. Its taxonomy for data visualization interaction identifies seven categories of techniques[4]:

- Select, to mark items of interest;
- Explore to examine subsets of the data;
- Connect to highlight related items within and across views;
- Abstract/elaborate to vary the level of detail;
- Reconfigure to show different arrangements of the data;
- Filter to show elements conditionally;
- Encode, to change the visual representations used.

Snapsible addresses these goals and meets requirements for data graphical display listed above through the 5 main interaction modes. Every mode has its own features, gestures, and ways of how Snapsible users can interact with visualized data.

Snapsible Modes [9]

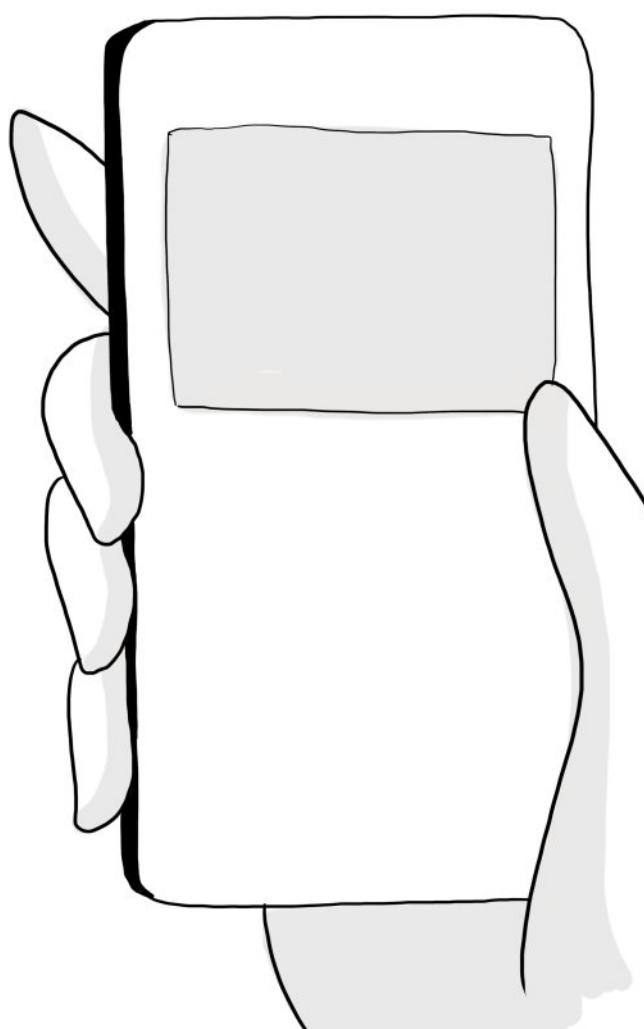
Closed	Closed Mode			
Opened	Portrait Compact	Portrait Notebook	Landscape Notebook	Landscape Compact

The proposed modes will automatically adjust the location and size of visualized data objects based on the location of the folding area (hinge mechanism). In addition to the original architecture, the proposed interface also contains other components, including an Auto-Adjusting Graphs Placement, a thread to detect the value from the bending sensor, a driver for the bending sensor and a Java Native Interface (JNI) library. [5]

Snapable interactive modes will be demonstrated based on the visualization of Best Artworks Of All Times dataset from Kaggle [6] using Altair[7] and Vega Lite[4].

Modes Description

Closed Mode

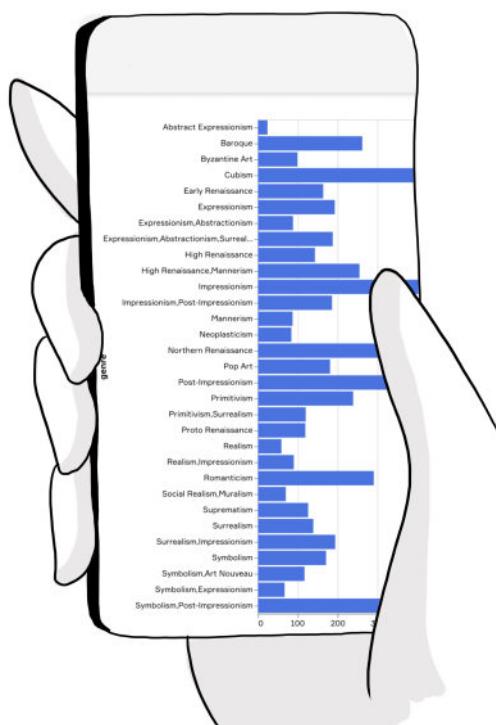


That is standard standby mode for the Snapable when the user can see only important notifications and time on a small front display.

Fig 9: Closed Mode

Open modes

Portrait Compact Mode

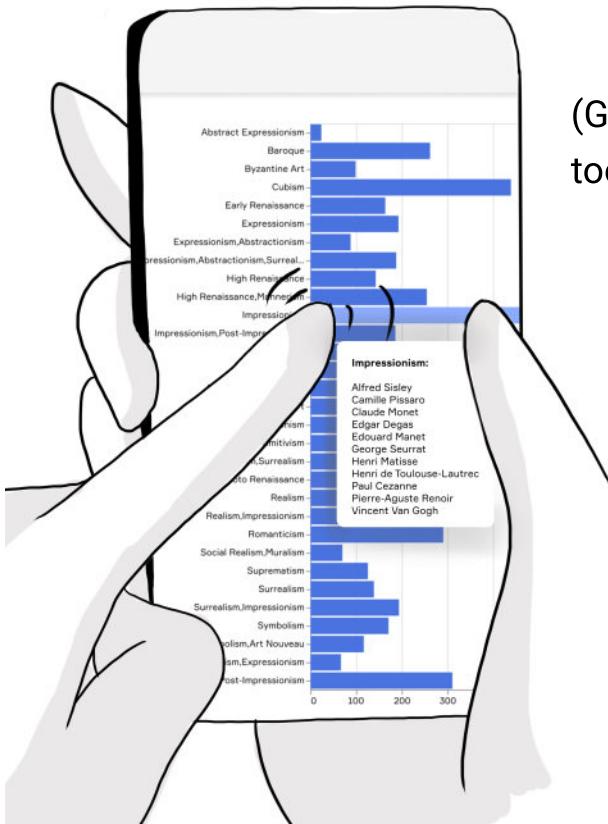


For this mode, Snapsible is folded along the hinge mechanism axis and placed vertically in hands. This position of foldable displays may remind users of the standard form of modern touch-screen smartphones. Based on the gyroscope, the device identifies which side of Snapsible faces the user and activates it, the opposite side switches off and becomes inactive. Mainly, in this mode users cannot connect to highlight related items across views or reconfigure to show different data arrangements, but they can do standard graphs exploration and variation of details using specific gestures.

Fig 10 a,b,c: Portraitit Compact Mode

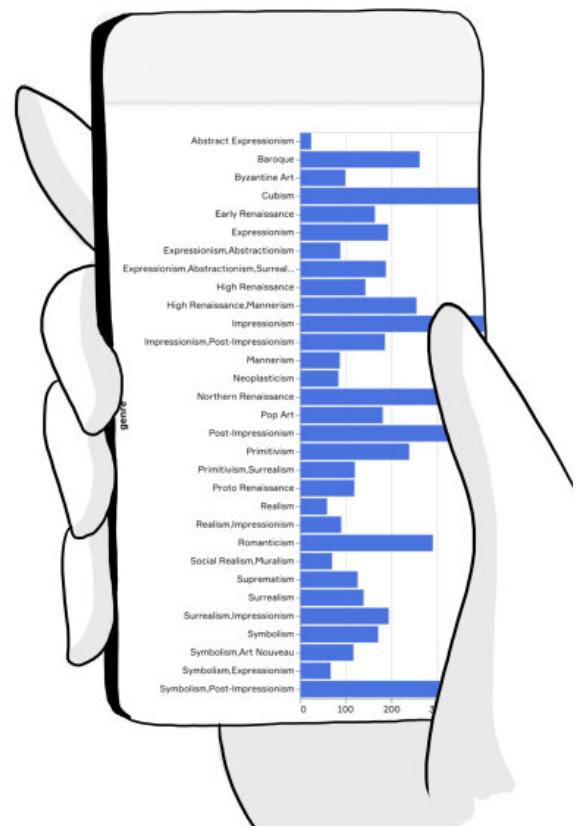
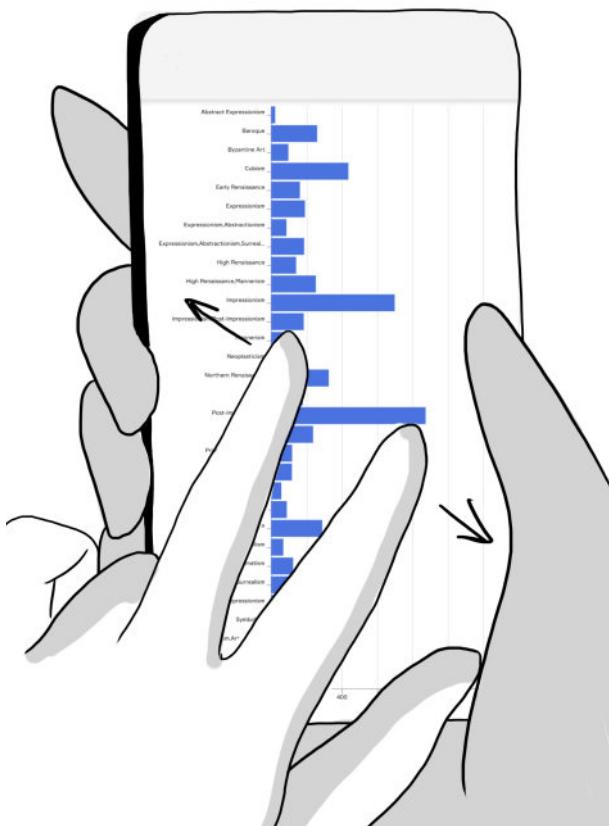


(G1) Double finger touch and dragging up/down will help the user to scroll through the graph.



(G2) Force-touch on one of the bars will evoke a tooltip with detailed information.

Fig 10 d,e,f: Portrait Compact Mode



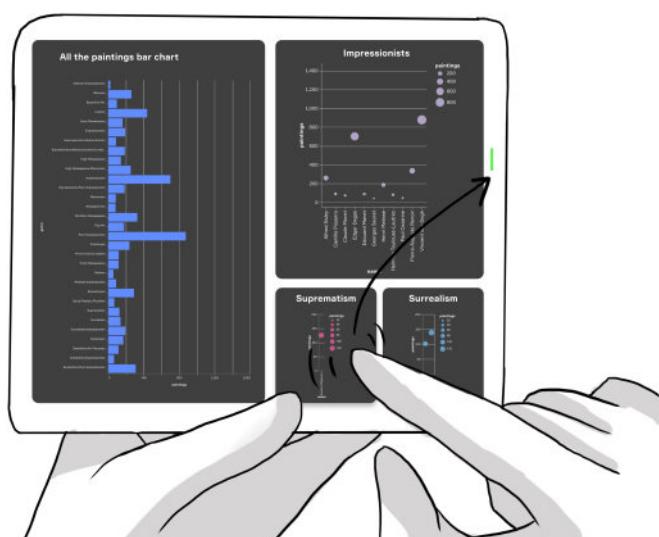
(G3) Pinch gesture enables the user to zoom in/out the graph.

Portrait Notebook

For this mode, Snapsible is unfolded and placed vertically. Besides G1–G3 gestures, the user can also communicate between multiple views and rearrange them. For communication between views, the user can touch one of the visual encodings on the graph. Other views will be adjusted to the selections made.



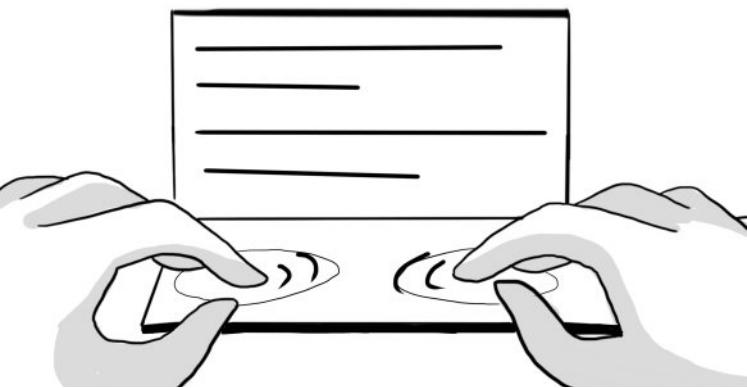
Another way to communicate between different graphs is by using the “brushing” mode. To do this, the user makes single-touch and drag (G4). By doing this they will select a certain area on one graph and other views will be adjusted based on the selection made.



To rearrange views user drags graphs by a top bar (G5). The view can be dragged to the highlighted places.

Fig 11 a,b: Portrait Notebook Mode

Landscape Notebook



To activate this mode, Snapsible is folded along the hinge mechanism axis by 90-130 degrees and is placed horizontally. The interaction with the graphs is similar to portrait compact (G1–G3 gestures), and one of the main differences is the usage of the horizontal screen as an input affordance (e.g. keyboard).

Fig 12: Landscape Notebook Mode

Landscape compact

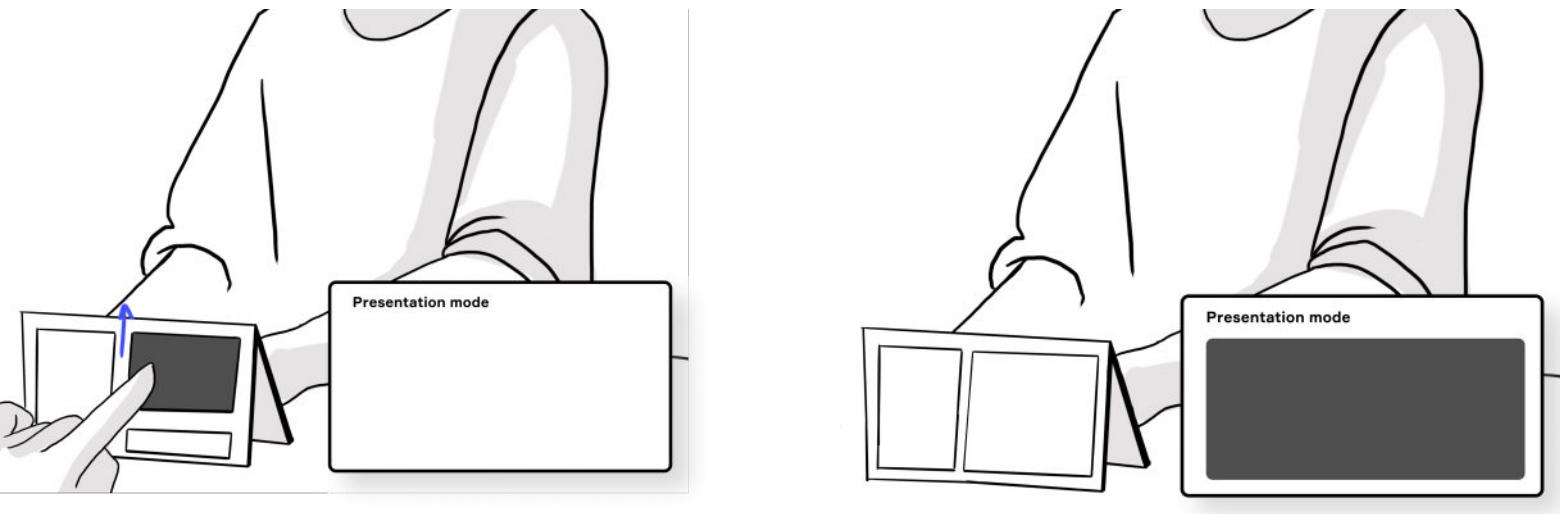


Fig 13: Landscape Compact Mode

For this mode, Snapsible is folded along the hinge mechanism axis for more than 130 degrees and is placed horizontally. The core value of this mode is to let users present certain visualizations by leveraging communication between two Snapsible screens.

By dragging the graph (G5 gesture) beyond the top part of the screen, the user can “send” this view to the opposite screen. Thus, users can select what kind of views they want to show.

Also, views can be erased by clicking the close button.

Engineering Details

Snapable folds to a smaller screen and unfolds to a larger screen supported by the device's hinge mechanism along an axis. The hinge assembly can provide several features that facilitate the use of a single flexible display across both portions. During rotation of the first and second portions, the hinge assembly can change the effective length of the device that lies beneath the flexible display to reduce stresses imparted on the flexible display.



Fig 14: The hinge mechanism

This aspect can be achieved with a cord that connects the first portion to the hinge assembly. A length of a pathway of the cord can change during the rotation so that the cord draws the first portion toward the hinge assembly and/or allows the first portion to be biased away from the hinge assembly depending on the orientation.

Another aspect relates to deployable bridge support that can support the flexible display in some orientations. The deployment of the bridge support can be controlled by a cam that can operate independently of the cord.

The display includes two layers -- one is the flexible OLED (a.k.a FOLED) panel and the second layer is a plastic outer shell. There is a polymer bond that holds the display panel, touch-sensitive digitizer, and the outer layer together. This adhesive allows more degree of freedom while folding and unfolding the device without leaving any crease, fractures, and spiderwebs that make the device unusable.

Future Development

Snapsible can partner with firms offering analytics, data visualization, and data monitoring for the foldable screen displays to further extend the possibilities the device has to offer. Please explore android developer documentation and features include multi-resume, resources access, and multi-display to expand Snapsible in these directions.

Citations

- [1] Edward R. Tufte, *The Visual Display of Quantitative Information*, 2nd edition, 1983.
- [2] Benjamin Watson and Vidya Setlur. 2015. Emerging Research in Mobile Visualization. In Tutorial Proceedings of the International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI). 883–887. DOI: <http://dx.doi.org/10.1145/2786567.2786571>
- [3] Android Developer Guidelines and Docs.
<https://developer.android.com/guide/topics/ui/foldables>
- [4] Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer. 2017. Vega-Lite: A Grammar of Interactive Graphics. *IEEE Transactions on visualization and computer graphics* (Vol. 23, No. 1, January 2017)
- [5] Tseng-Yi Chen, Shuo-Han Chen, Heng-Yin Chen, Wei-Kuan Shih. 2014. An Enhanced User Interface Design with Auto-Adjusting Icon Placement on Foldable Devices. *2014 IEEE International Conference on Systems, Man, and Cybernetics* October 5-8, 2014, San Diego, CA, USA.
- [6] Best artworks of all times dataset
<https://www.kaggle.com/ikarus777/best-artworks-of-all-time>
- [7] Altair: Declarative Visualizations In Python.
<https://altair-viz.github.io/index.html>
- [8] Emanthia Dimara and Charles Perin. 2020. What is Interaction for Data Visualization? *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS*, VOL. 26, NO. 1, JANUARY 2020
- [9] Yoshiyuki Kaneko, Muneaki Yamaguchi, Hitoshi Matsuya and Toshihisa Tsukada. 1996. Foldable-display Systems As A Standard Platform For Multimedia Use. *IEEE, Central Research Laboratory, Hitachi, Ltd. Kokubunji, Tokyo 185, Japan.*

Inspirations:

<https://www.microsoft.com/en-us/research/video/touchwave-kinetic-multi-touch-manipulation-hierarchical-stacked-graphs-2/>

Peter S. Games, Alark Joshi, "An evaluation-guided approach for effective data visualization on tablets," Proc. SPIE 9397, Visualization and Data Analysis 2015, 939704 (8 February 2015); <https://doi.org/10.1117/12.2076523>