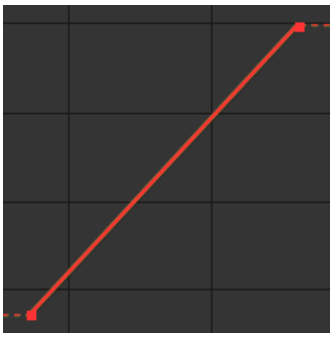
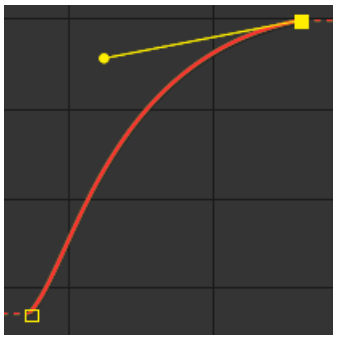
Interpolation Previewer

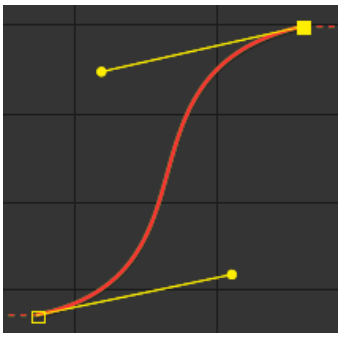
To make animation more juicy motion designers use interpolation. They apply interpolation to some transitions to make it not linear but specific. To do so designer, need to select a transition and move control points (marked yellow) on the Interpolation Graph in a specific designer environment.



Linear transition



Interpolated transition



Interpolated transition

After Interpolation specified - animation plays awesome!

But now designer needs to share the information about interpolation with developer. We need to make a tool that can help designer understand how to pass the interpolation data.

For this purpose we need to make a web application which will show similar graphic visualization of the Interpolation Graph, but also show the values that specify the line. You probably know that all those lines (orange) can be represented via cubic Bezier curve. Cubic Bezier curve can be represented with start point, end point and two control points each represented with {x, y} coordinates.

The layout of the web application should contain:

- List of selectable predefined interpolations (see list below).
- Interpolation Graph representing coordinate axis and interpolation line. (*canvas or svg can be used for implementation here*)
- Input fields representing the control point values of the interpolation currently shown on Interpolation Graph.

Use-cases:

- User should be able to select interpolation from the list, which should reflect and update the Interpolation Graph with the item selected
- User should be able to change input field values, those changes should reflect on the Interpolation Graph.
- User should be able to drag & drop control points on the canvas. Those changes should reflect on the Interpolation Graph and on the values in input fields also.

Additionally, designers would be grateful if they could run this web-app on the Android mobile phone as separate application, which can be installed with the apk.

List of predefined interpolations (format: name & control points):

- ease (0.25, 0.1, 0.25, 1)
- ease-in (0.42, 0, 1, 1)
- ease-out (0, 0, 0.58, 1)
- ease-in-out (0.42, 0, 0.58, 1)
- sine-in-out (0.45, 0.05, 0.55, 0.95)

**-Start point should always be {0, 0}, while end point should always be {1, 1}.*