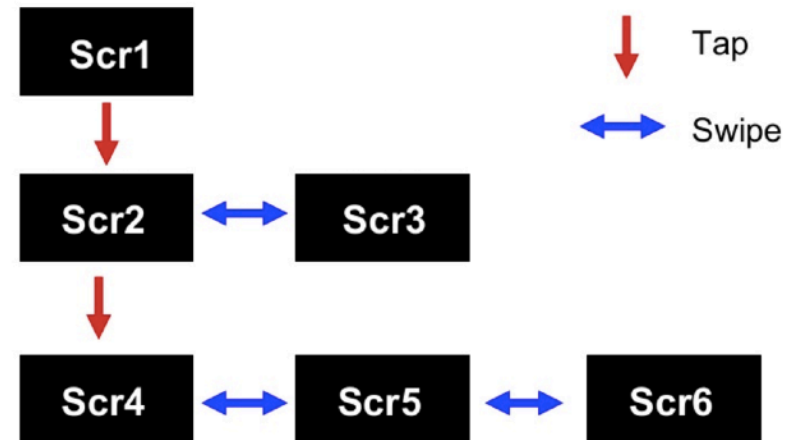


Exercises

Quick prototyping on Android (Wear) wearables:

- starts with UI design, storyboarding, wireframing:

- **glass-ui.com,**
glassware-flow-
designer.appspot.com
- **PopApp.in**
- **UXPin.com,**
Proto.io



- for prototypes that can be deployed on the wearable:

- **processing.org**
- **wearscript.com**

Exercises

Processing in 5 steps:

- download at: **processing.org** (± 200 MB, 15 minutes)
- connect an android phone via USB (10 seconds)
- on phone, enable USB debugging (20 seconds):
Settings -> Developer options -> USB Debugging
- in Processing, on top right ('Java'):
select **Add Mode**,
install **Android Mode** (20 minutes)
- ready! Time to Program ...

Exercises

Processing hello world:

```
void setup() {  
  background(0);  
  orientation(LANDSCAPE);  
  size(640, 360);  
  textAlign(CENTER, CENTER);  
  textSize(47);  
}  
  
void draw() {  
  background(0);  
  fill(255);  
  text("Hello World", 320, 180);  
}
```

Exercises

Processing drawing / interaction:

```

/* World's simplest Android App! blprnt@blprnt.com, Sept 25, 2010 */

float boxRotation = 0; //Build container to hold current rotation of box
void setup() {
  size(480,800); //Set the size of the screen
  smooth(); //Turn on smoothing to make everything pretty.
  fill(255); //Set the fill and stroke colour for the box and circle
  stroke(255);
  rectMode(CENTER); // rectangles to draw from center (default is TL corner)
};

void draw() {
  background(mouseY * (255.0/800), 100, 0); // moving down -> more bg red
  boxRotation += (float) (pmouseX - mouseX)/100; // box rotation, right-left
  line(width/2, height/2, mouseX, mouseY); // Draw ball-and-stick
  ellipse(mouseX, mouseY, 40, 40);
  pushMatrix(); //Draw the box
  translate(width/2, height/2);
  rotate(boxRotation);
  rect(0,0, 150, 150);
  popMatrix();
};

```

Exercises

Processing acquiring sensor data:

```
import ketai.sensors.*;
KetaiSensor sensor;
float accelerometerX, accelerometerY, accelerometerZ;

void setup() {
  sensor = new KetaiSensor(this);
  sensor.start();
  orientation(LANDSCAPE);
  textAlign(CENTER, CENTER); size(640,360);  textSize(12);
}

void draw() {
  background(78, 93, 75);
  text("Accelerometer: \n" +
    "x: " + nfp(accelerometerX, 1, 3) + "\n" +
    "y: " + nfp(accelerometerY, 1, 3) + "\n" +
    "z: " + nfp(accelerometerZ, 1, 3), 0, 0, width, height);
}

void onAccelerometerEvent(float x, float y, float z) {
  accelerometerX = x;
  accelerometerY = y;
  accelerometerZ = z;
}
```

Exercises

Pointers to tutorials and examples:

- <http://android.processing.org/tutorials/>
- <http://ketai.org/download/> + <http://ketai.org/examples/>

Exercises

Your Android App for Smartwatch + Smartphone + Glass:

- Jogging App that *optimally* displays progress in steps
 - Use the right accelerometer axis and a threshold:

```
if (abs(accelerometerX - x)>10.0) { stepCount++; }
```
 - See *glanceability* concept for inspiration
 - Test you design yourself