


# S2 2019 workshop

-  Because the special situation of a pandemic , physical installation in not possible
- However, Alex mentioned in Apr20 note that he want your installations to be feasible
- Only providing options here, **not** must use, pick what you feel comfortable
- Example: [Example.odt](#)

## System Diagrams



- System Diagram: Shows how different parts of the installation interact
  - Also relevant: how embeded part is connected with screen interaction
    - Hardware
      - Wired
        - USB-Serial
        - LAN
        - Etc
      - Wireless
        - Wifi
        - Bluetooth
        - LoRa
        - Etc
    - Protocol
      - Self defined (Usually for simple projects)
      - TCP/IP
      - Fieldbus
      - Etc.
- Interaction Diagram: Shows how user uses the installation and how the installation reacts
  - [A bit simple, but still shows how user operates it | Also a bit simple](#)
  - [Using multiple pages](#)
  - [Has more text](#)

## Screen Interaction Part



- The would be the part that you would be focus on as it's possible to prototype it
  - Usually all the Visual/Audio/Gesture element goes here
- Tools used from last years experience:
  - More User Friendly, Less Customizable:
    - [Adobe XD](#) and [Origami studio](#) is what you like for screen interaction also [Sketch](#)
    - [Construct2](#) free for small scale project, exports HTML
  - Less User Friendly, More Customizable (If you need not only point and click, but interactions like drawing, dynamic content):
    - [Processing](#) and [P5JS](#) ( seems some of you already very good at it )
    - Game Engine:
      - [Unity](#) ( All of you should already be familiar with it )
      - [Unreal](#) , [CryEngine](#) , [Godot](#) all also Free to Use
  - More complex screen interaction will require you go about the hardway

```
float mxl, myl;
boolean startdr = false;

void setup() {
  size( 400, 300 );
  background(255);
}

void draw() {
  // clear button
  stroke(0); noFill();
  strokeWeight(2);
  rect(1, 1, 398, 298 );
  noStroke(); fill(128, 128, 128);
  rect(0, 0, 20, 20);
}

void mousePressed(){
  if( mouseX <= 20 && mouseY <= 20 ){
    background(255);
  }else{
    startdr = true;
    mxl = mouseX; myl = mouseY;
  }
}

void mouseDragged(){
  if( startdr ){
    noFill(); stroke(0);
    strokeWeight(3);
    line( mxl, myl, mouseX, mouseY );
    mxl = mouseX; myl = mouseY;
  }
}

void mouseReleased(){
  startdr = false;
}
```

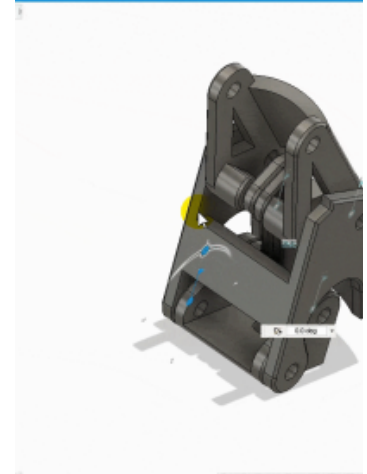
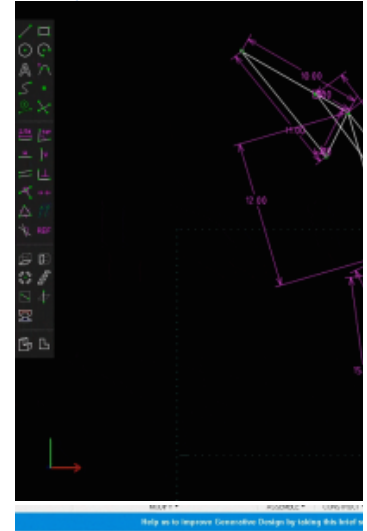
- Screen based interaction can also have computer attached more complex sensors:
  - Screen
  - Projection
  - Touch ( eg: [Smart mirror](#) )
  - Motion capture ([Kinect/Leapmotion](#))
  - others...

## Physical Installation



- Structures:
  - Overall arrangement of the **structure / layout**
    - Hand sketch
    - software like: SketchUp
- Mechanics:
  - **Looks-like versus Works-Like**
  - For this, as there's no trial and error, work-like is goal, however, there would be caveat for real construction always
    - Tools that can do simulation

- [Solvespace](#)



- [Linkage](#)
- Also CAD/game engine/blender if you want to

- Electronics:
  - For any
    - sensors:
      - Sound
      - Distance
      - Gas
      - Temperature/Humidity
      - etc...
    - actuators:
      - Motor
        - Stepper/servo/linear
      - Heater
      - [Hydraulics](#)
      - etc...

## System Diagrams



- Budgeting (estimation)
  - [Bill Of Materials](#)



- Video to use:
  - Identify things to use: [https://www.youtube.com/watch?v=b5\\_pkiJv8\\_k](https://www.youtube.com/watch?v=b5_pkiJv8_k)
  - Really need to seriously think about feasibility:
    - <https://www.youtube.com/watch?v=nseRCJl9gKY>
    - [https://www.youtube.com/watch?v=XkFC\\_UyBh-8](https://www.youtube.com/watch?v=XkFC_UyBh-8)
  - Mechanical mockup vs functional: <https://www.youtube.com/watch?v=Kdsl5ibR9Pk>