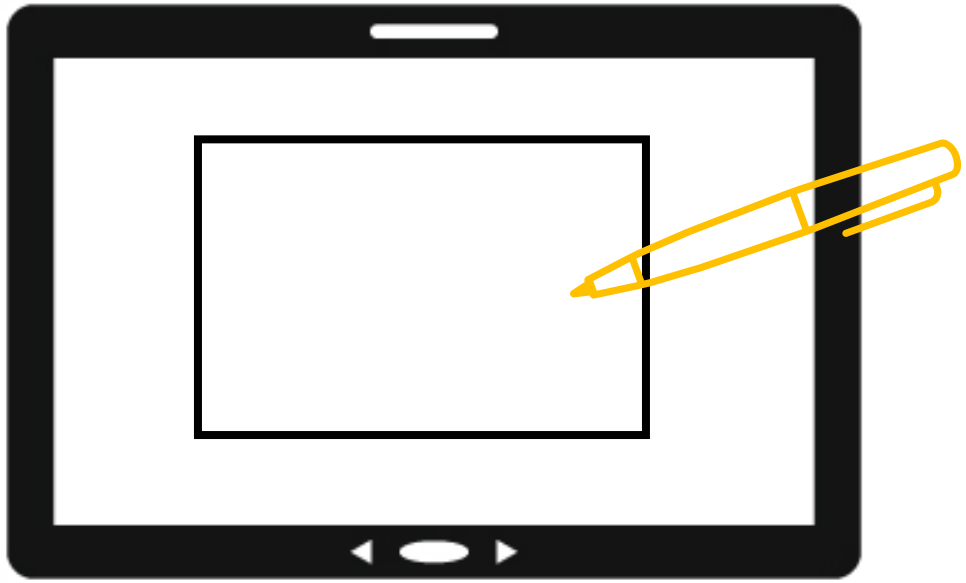
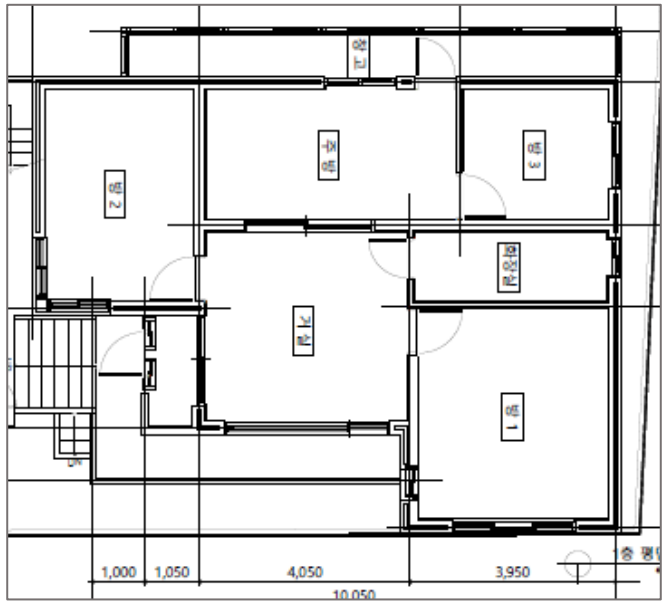
# Sketch2Cad

Software 201735892 □ □ □
Software 201735879 □ □ □
Software 201835442 □ □ □
Software 201935047 □ □ □

# Contents

# Introduction






**.dwg**

**Interview who work in architecture office**

**Time to turn the floor plan(82.5$m^2$) to CAD**

**more than 6 hours**

**Interview who work in architecture office**

**Why?**

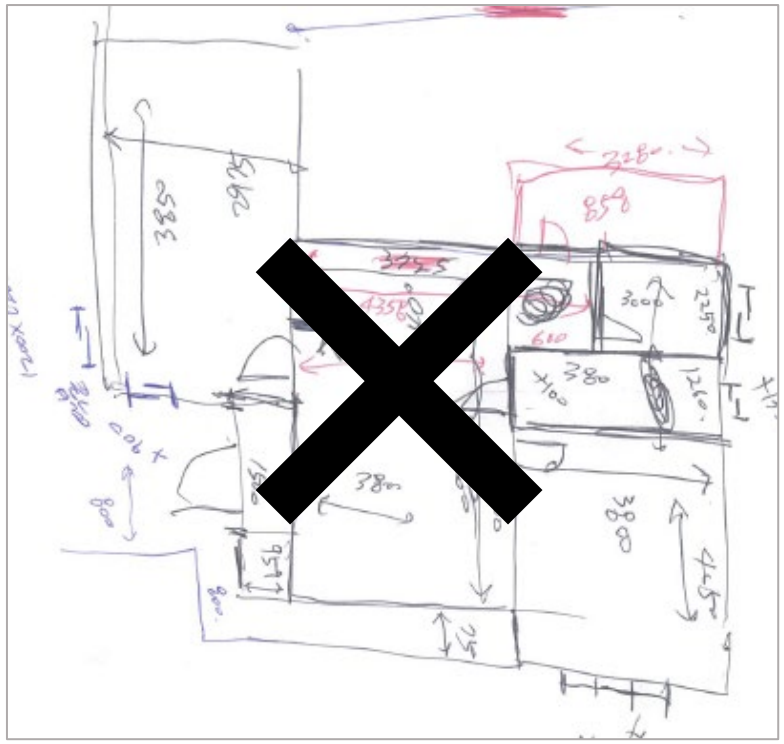
1. Hard to understand except drawer
2. Shape rate is not correct
3. **Time is not enough**

Interview who work in architecture office

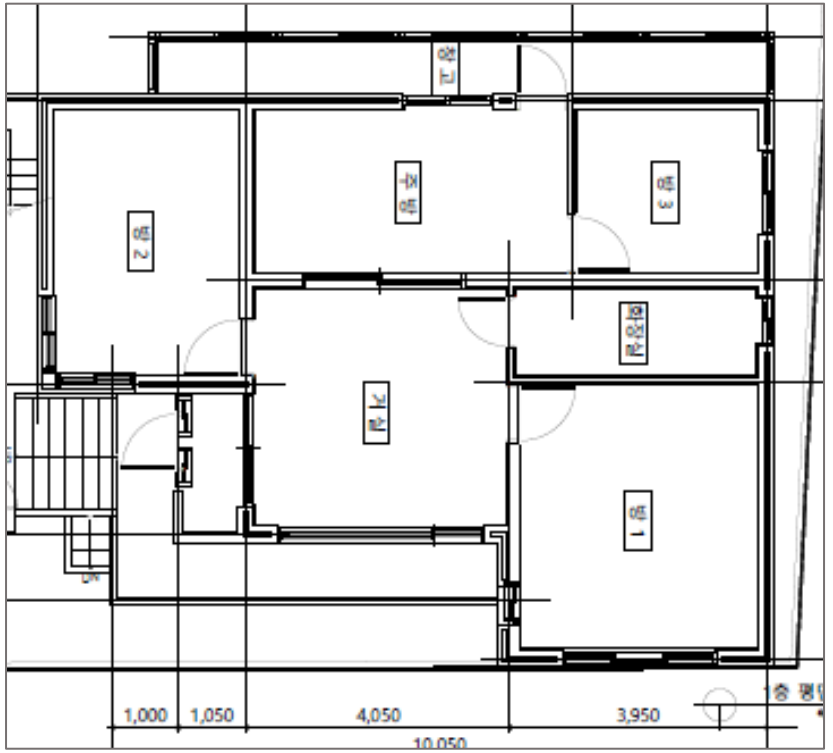
**Conclusion:**
**Too much time to turn the floor plan into a CAD file.**

**Want a program to create a CAD file automatically**

# Analyze Problem

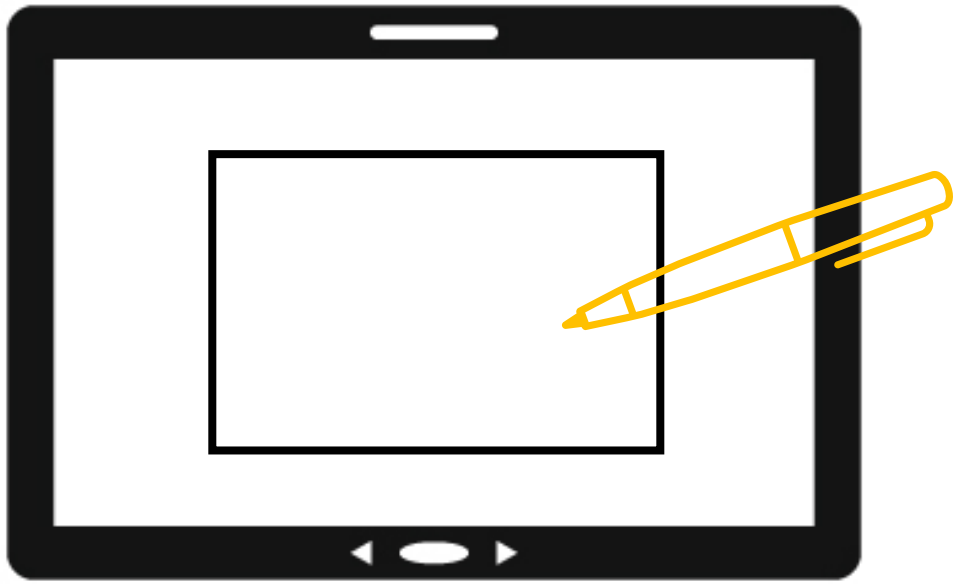


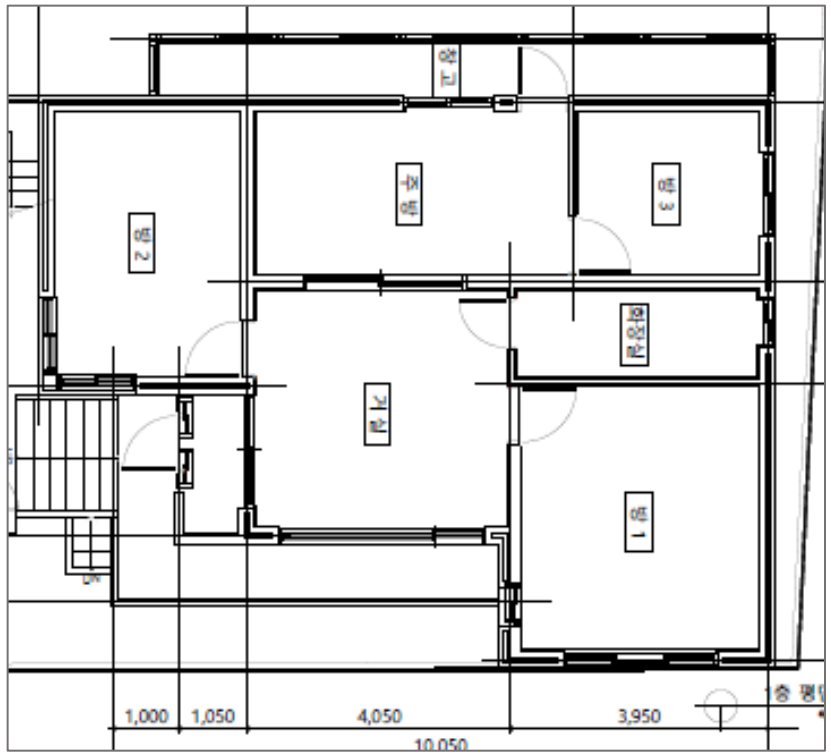
<Image of floor plan>




<CAD>

# Analyze Problem






**Instead of picture, Draw on tablet**

→ To improve recognition

→ To improve preprocess accuracy

## 도면 제작사


**Strength**

- To draw exact floor plan
- Possible to use various architecture element

**Why it is <u>not popular</u> with expert?**

- Takes too long time
- Do not export CAD file

## Floor Plan Creator

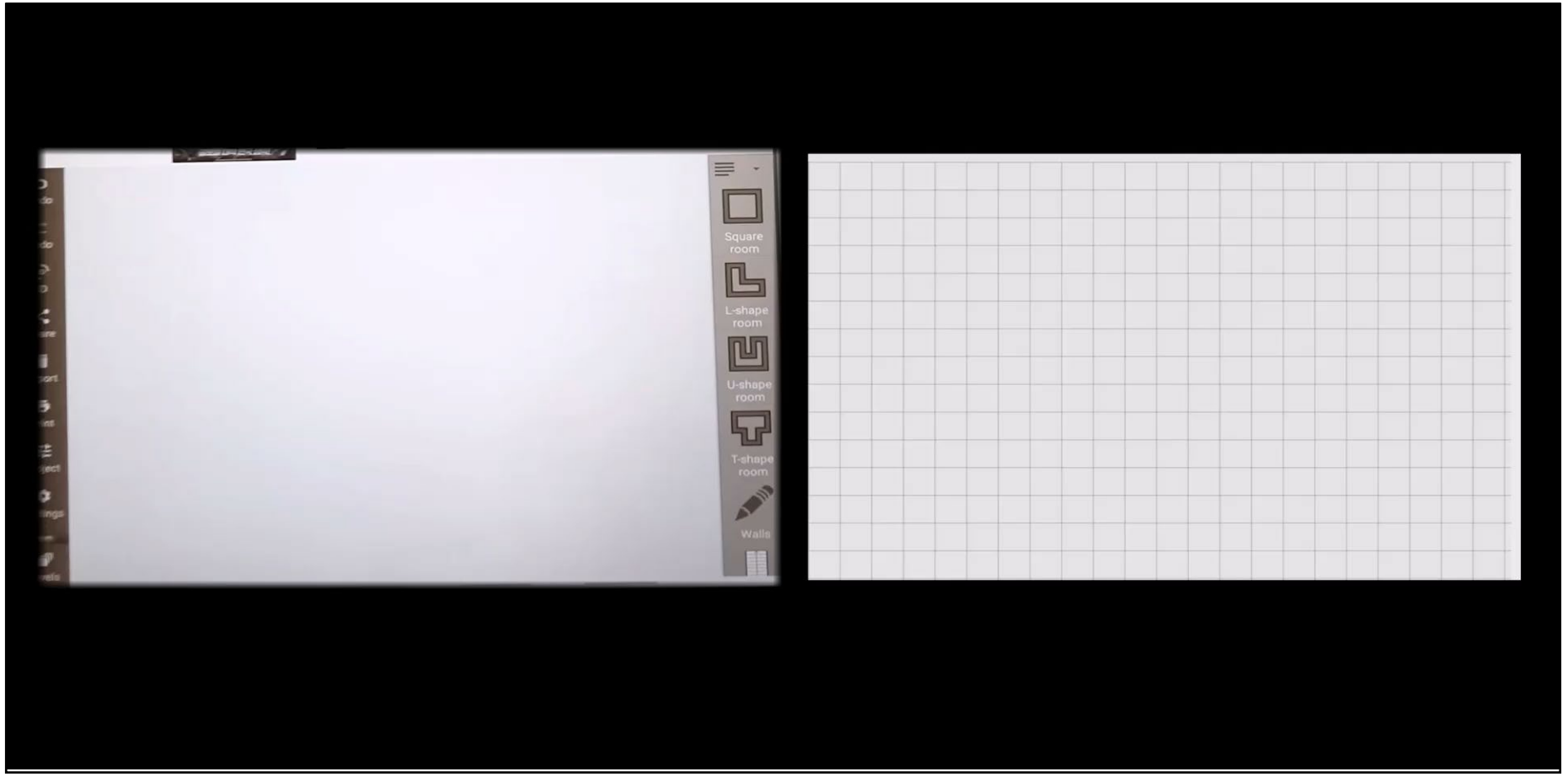

### Strength

- CAD export in paid app
- Easy to use

### Why it is <u>not popular</u> with expert?

- Takes more time than draw on paper
- Hard to draw detailed

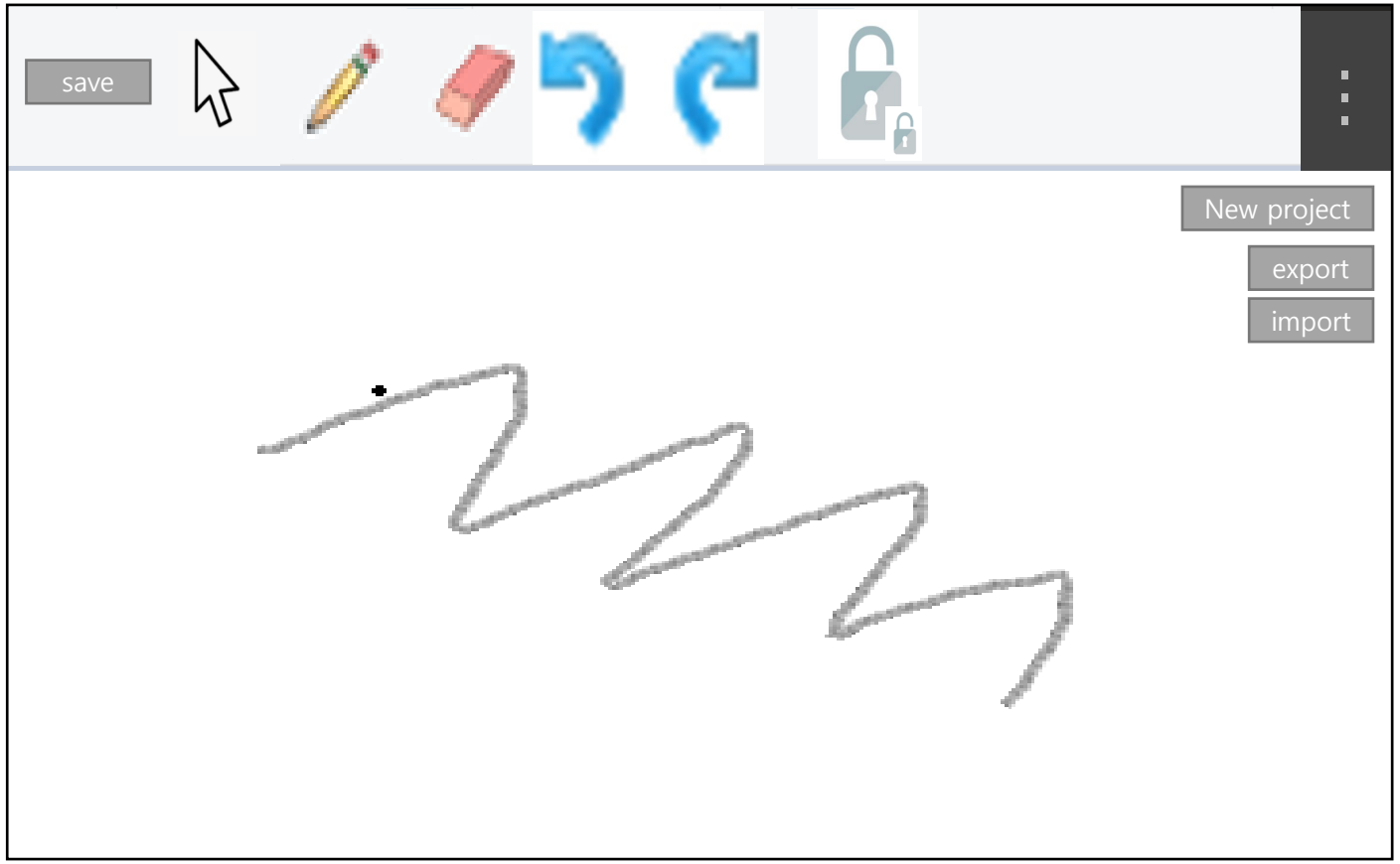
## Floor Plan Creator vs Our program expected



- **Architecture Experts**
- **Similar method draw on paper**
- **More obvious than draw on paper**
- **Less than 3minutes draw  82.5m² floor plan**
- **Export CAD file**

# Key features

# Key features




## Create a Paint that allows users to draw pictures in the tablet app

### S Pen Remote SDK

#### Overview

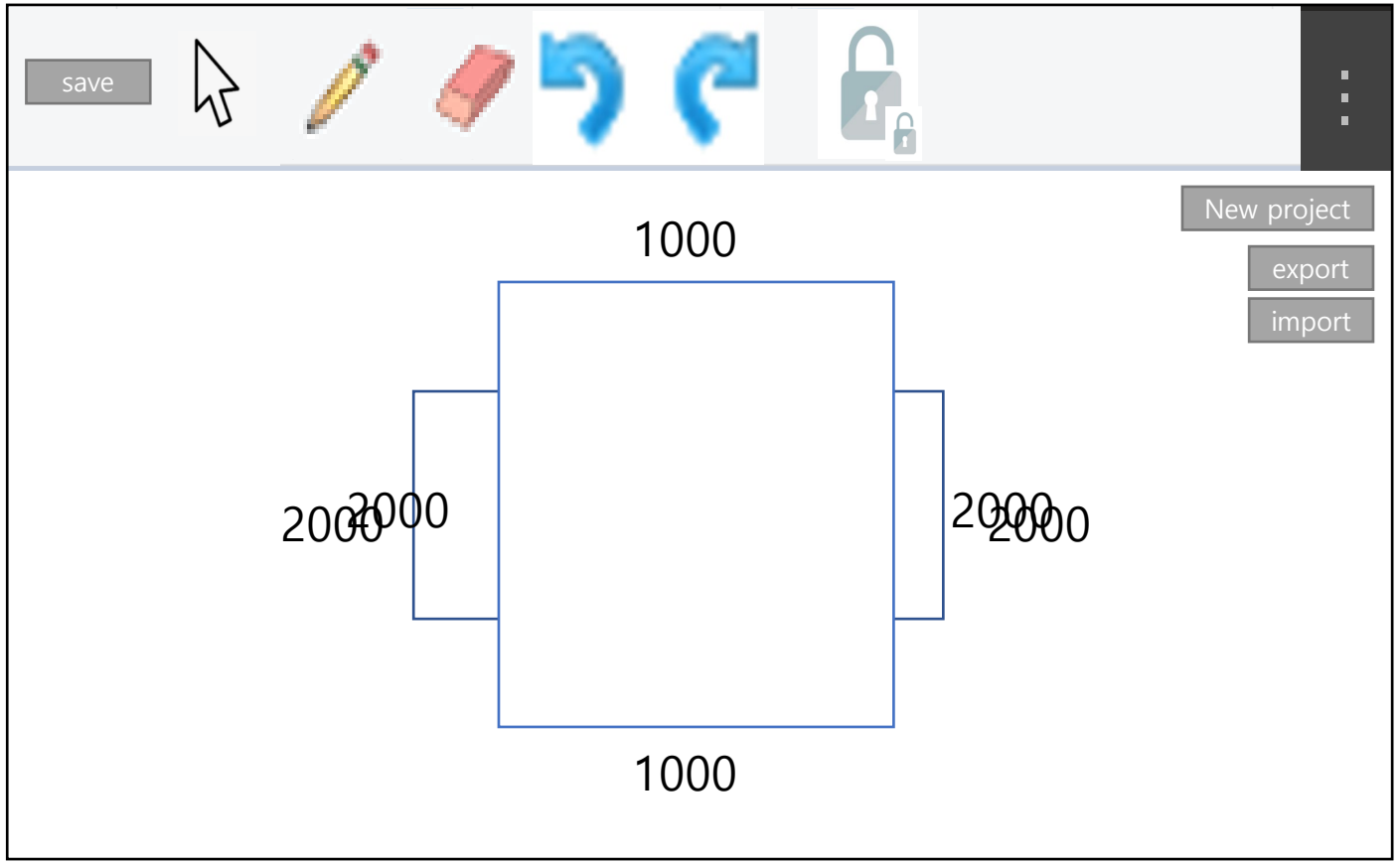
S Pen Remote SDK allows you to develop applications that use S Pen Remote generated events. S Pen includes buttons and motion sensor, and events from these units can be delivered to the application via the S Pen Framework. Simple gestures such as Button Click or 4-directional Swipe can be easily recognized via the S Pen Framework. But to implement a more powerful application using the raw data of units, you must use the S Pen Remote SDK.

#### Supported Platform

The package uses a static Java library that depends on internal Android framework modules. It means that this package only runs on devices that support those modules. Some Samsung Smart Devices do not support S Pen Remote as it requires devices of the Note series.

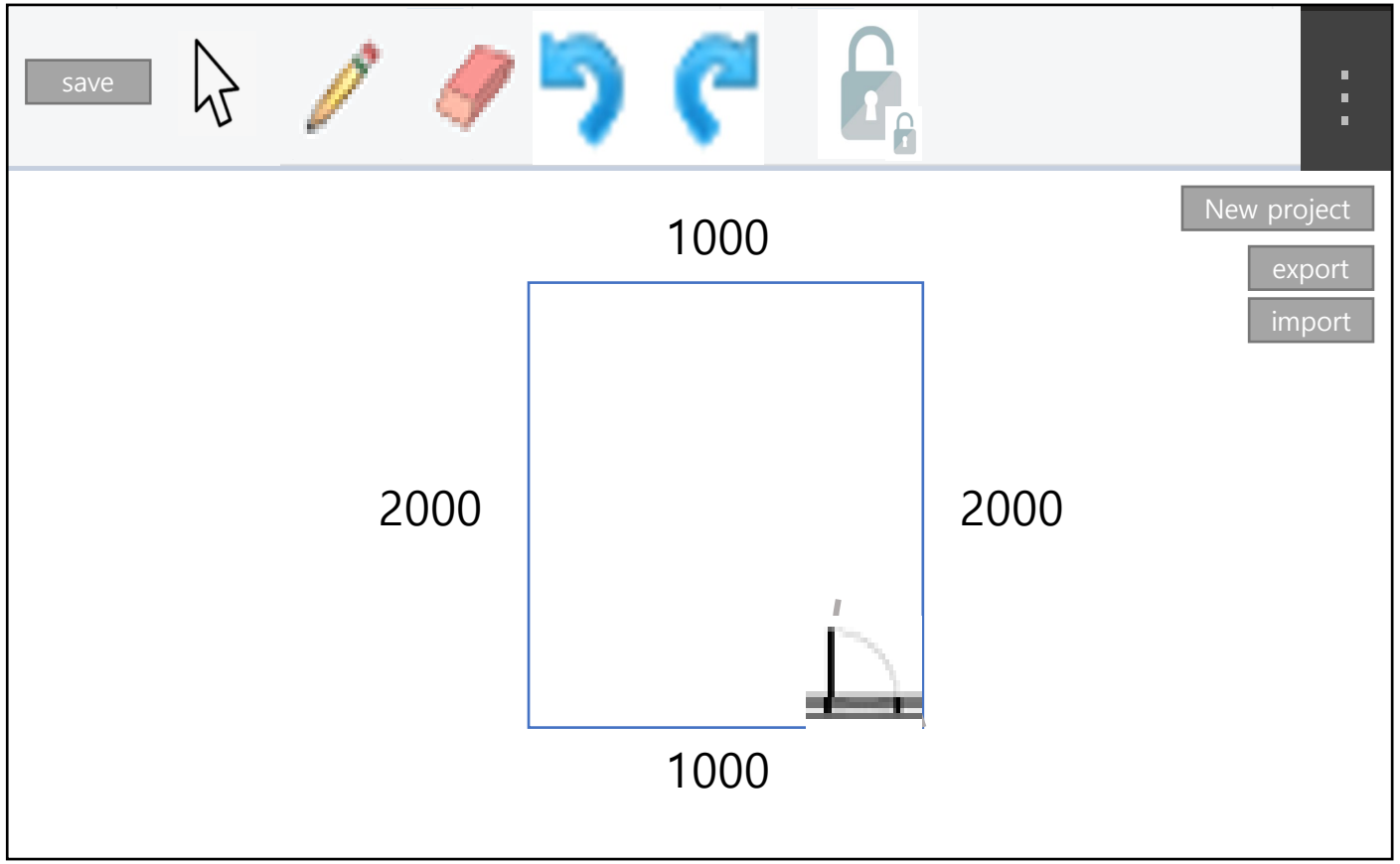
*https://developer.samsung.com/galaxy-spen-remote/s-pen-remote-sdk.html*

# Key features




The user modifies the shape according to the

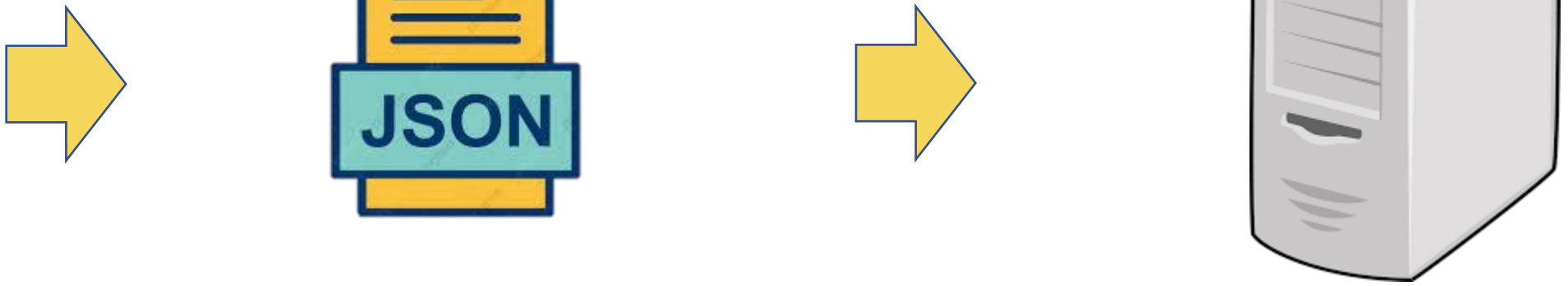
# Key features




The program recognizes symbols for doors, windows, stairs, etc.

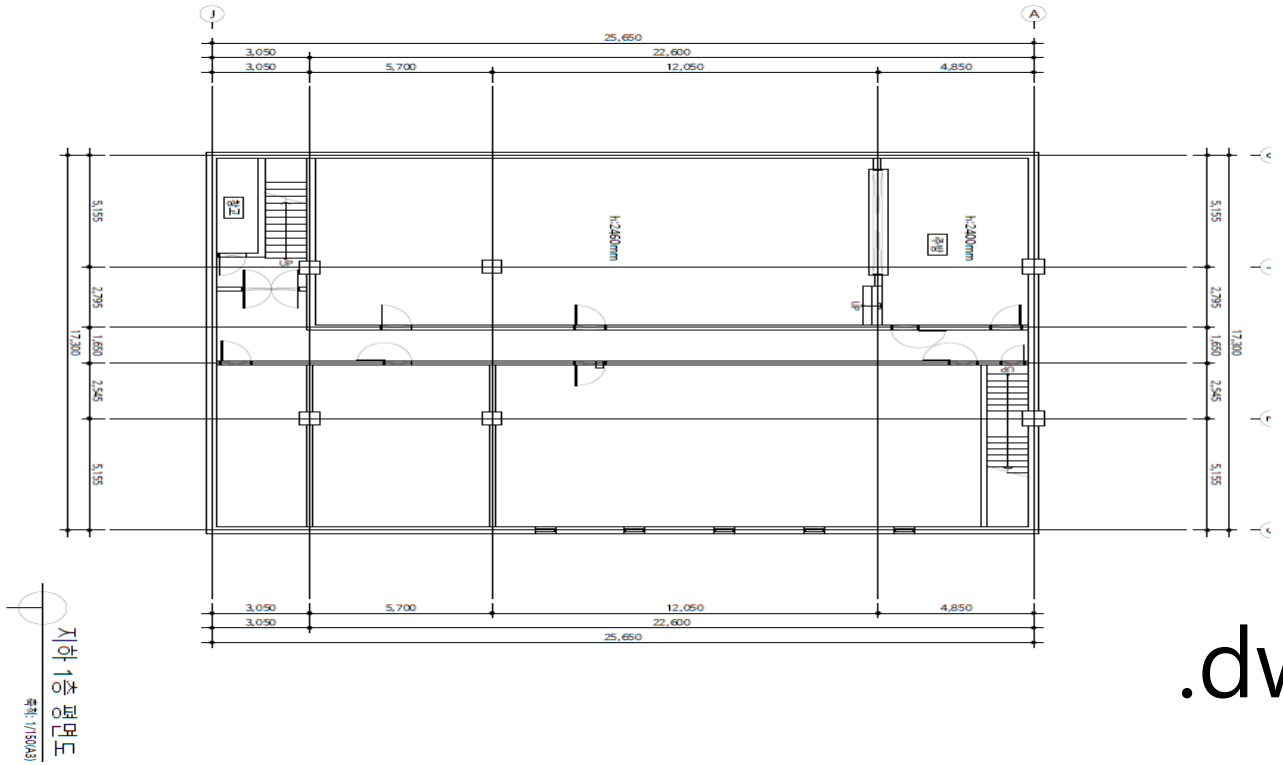
# Key features

server




The program sends the required part of the dwg file to the server in a json file.

# Key features

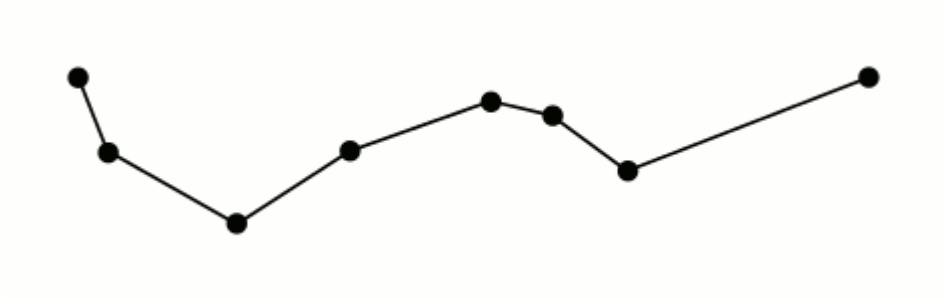



.dwg

The server converts to a dwg file and sends it back.

# Technology used

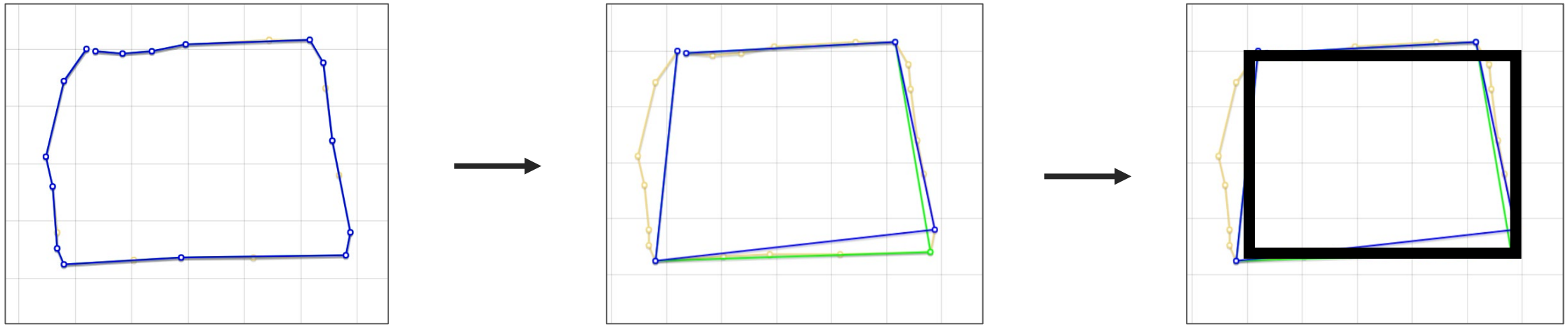
## Line to Straight



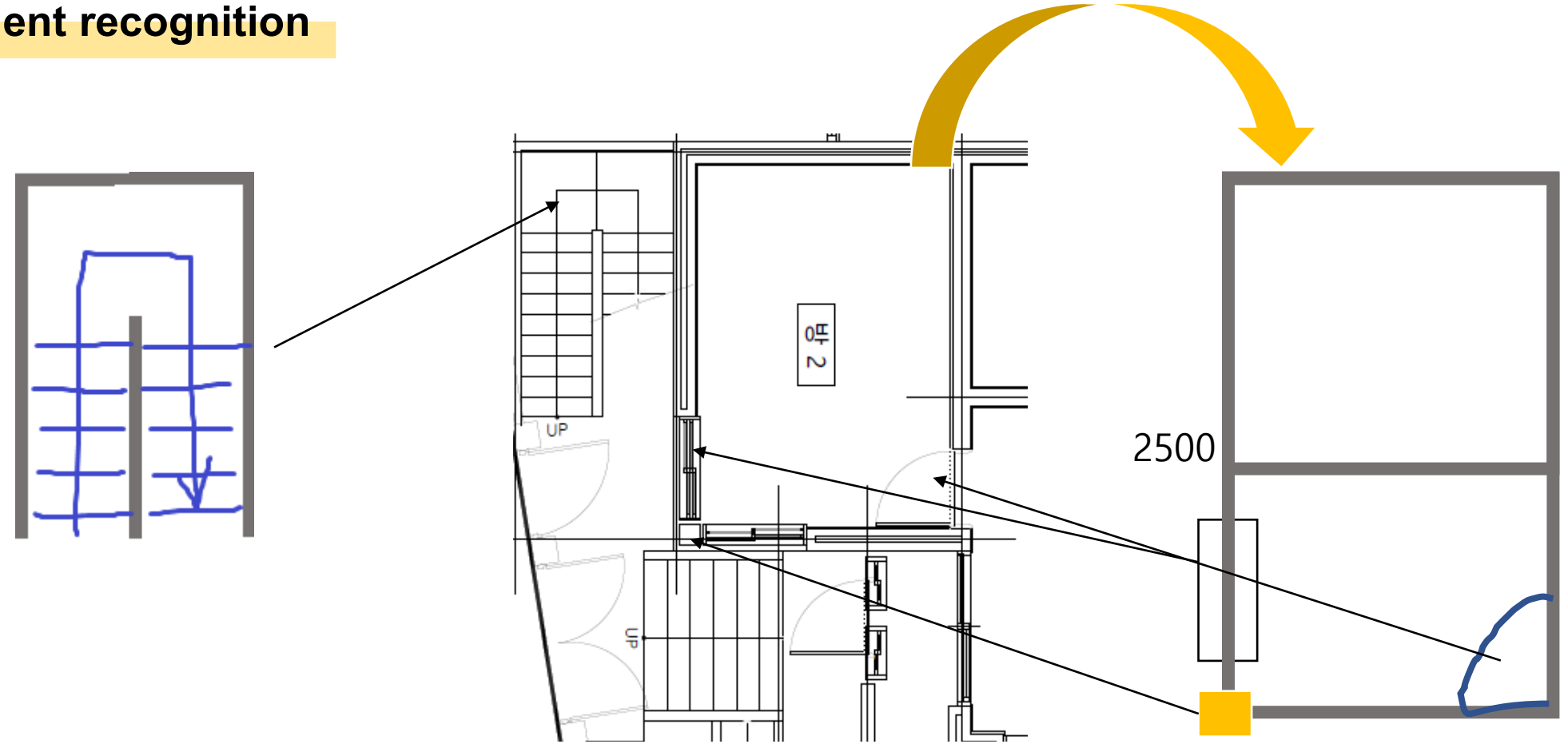
**Douglas Peucker Algorithm**

**→ Iterative end-point algorithms**

**→ Tuning to algorithm angle to 90degree**



## Element recognition



## Number recognition

- **How?**

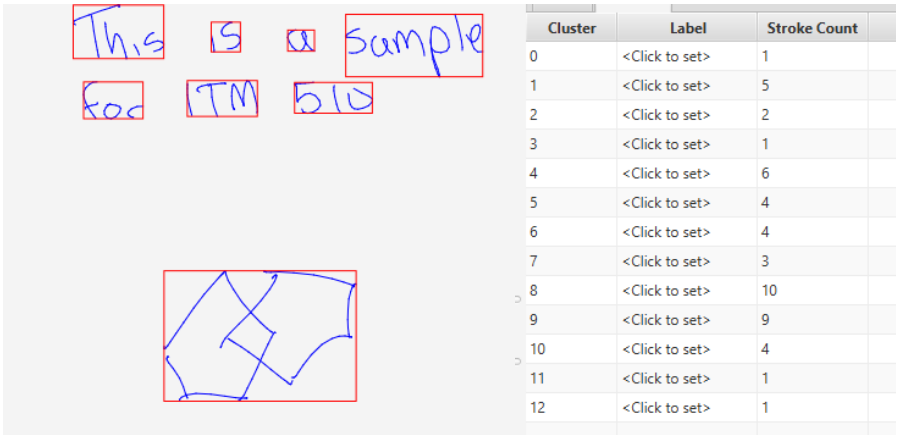
Recognize numbers drawn on tablet->
Using Deep Learning(Tensorflow/mnist data set)

**2000 recognition**

## Point

Because the accuracy of the model is not 100%,
Users should write down the numbers so that the app can recognize them.

## Grouping



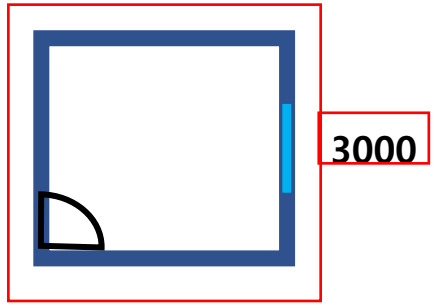

| Cluster | Label | Stroke Count |
|---|---|---|
| 0 | <Click to set> | 1 |
| 1 | <Click to set> | 5 |
| 2 | <Click to set> | 2 |
| 3 | <Click to set> | 1 |
| 4 | <Click to set> | 6 |
| 5 | <Click to set> | 4 |
| 6 | <Click to set> | 4 |
| 7 | <Click to set> | 3 |
| 8 | <Click to set> | 10 |
| 9 | <Click to set> | 9 |
| 10 | <Click to set> | 4 |
| 11 | <Click to set> | 1 |
| 12 | <Click to set> | 1 |

https://github.com/ttruty/SmartInk

**Cluster with 3 feature**
**Time, Distance, Type**

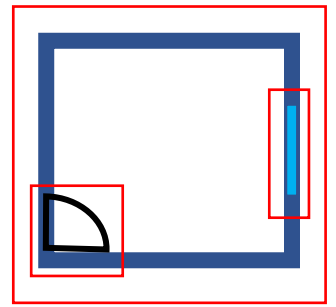
## Mapping

**Boundary meet**
**Element & Digit**



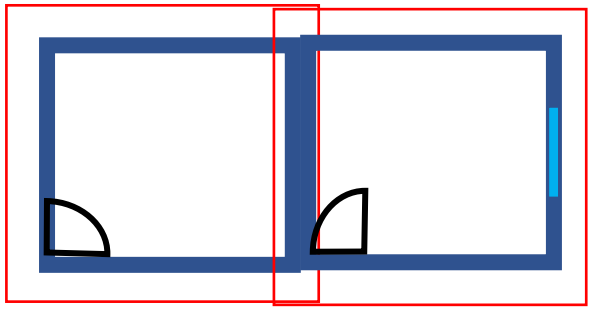

## Child

**Boundary contain**
**Element & Element**

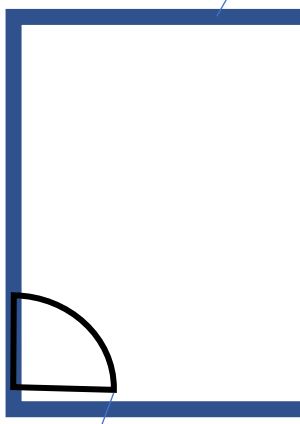


## Neighbor

**Boundary meet**
**Element & Element**



## Processing cad



Wall
Width: 2000
Height: 3000
High: 2300

Door
Width: 900
Height: 2100

Window
Width: 1000
Height: 1000

**Json**

```
Room :{
        X,Y : (0, 0)
        Wall1 : {
        X,Y : (0, 0)
        direction: vertical
        length: 3000
        width: 200
        Door : {      X,Y : (0, 200)
                      type: normal
                      direction: right-up
                      width: 900
                      height: 2100}
        }
        Wall3 : {
        .....
        Window: {  X,Y : (3000, 1000)
                      type: fixed
                      width: 100
                      length: 1000 }
        }
}
```

**Server**

## Convert to cad

**Json**

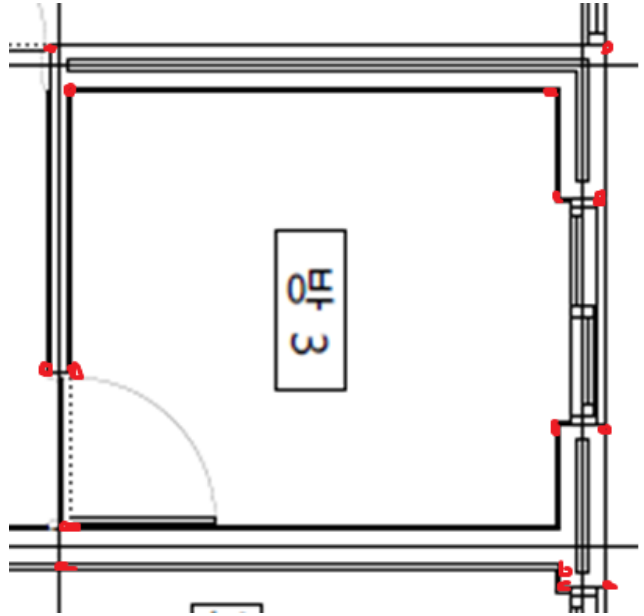
```
Room :{
        X,Y : (0, 0)
        Wall1 : {
        X,Y : (0, 0)
        direction: vertical
        length: 3000
        width: 200
        Door : {    X,Y : (0, 200)
                    type: normal
                    direction: right-up
                    width: 900
                    height: 2100}
        }
        ..............
}
```

Parsing to **Python**

**Python**

```
Wall1 = [(x,y),(0,0),(1000,0)....]
Wall2 = [ ..... ]
Door = { x = 0, y = 200,
         type = normal,
         width = 900, direction: 1 }
Window = { x = 3000, y = 1000,
           width = 1000,
           length = 1000}
```

Use **pyautocad**



## Used

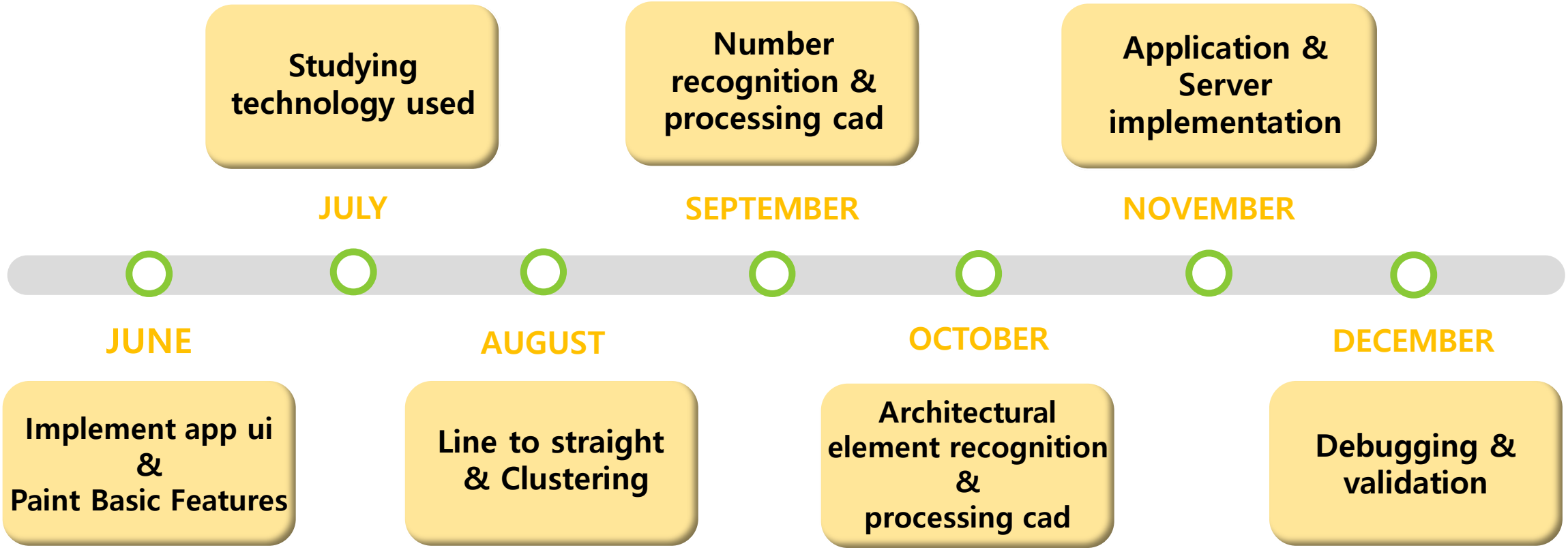
1. Pyautocad libaray
2. S pen remote SDK
3. Tensorflow input Mnist

## To contribute

1. Json2CAD

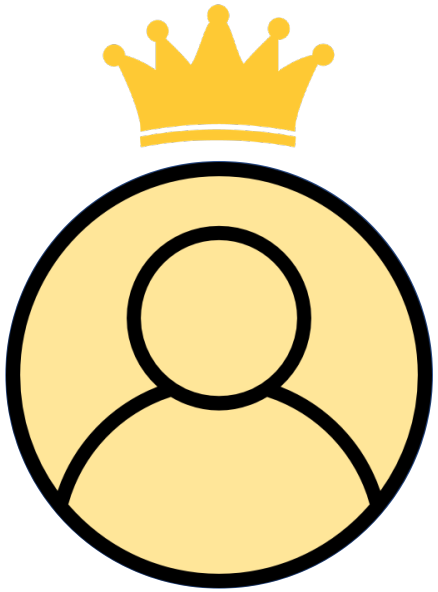
# Time Table

# Time Table


Studying technology used
JULY
Number recognition & processing cad
SEPTEMBER
Application & Server implementation
NOVEMBER
JUNE
Implement app ui & Paint Basic Features
AUGUST
Line to straight & Clustering
OCTOBER
Architectural element recognition & processing cad
DECEMBER
Debugging & validation

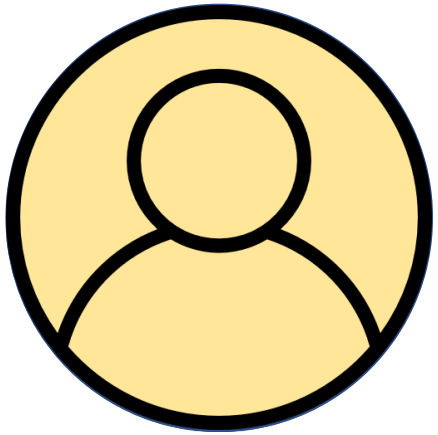
# Member Role

# Member Role



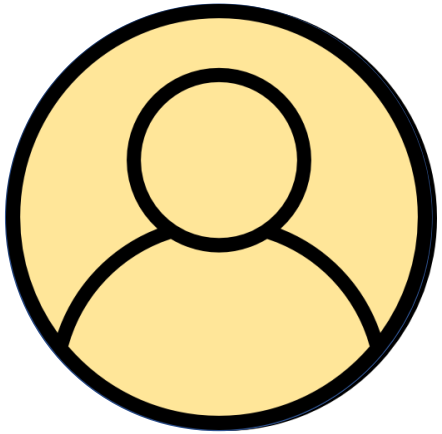
**최찬영**

**App ui,
Processing cad &
Server implementation**



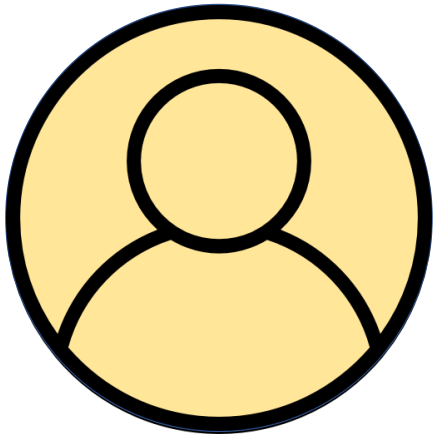
**김현민**

**Paint Basic Features,
Processing cad &
Server implementation**



**정호진**

**Paint Basic Features
Line to straight,
Numeric recognition**



**박승민**

**App ui,
Element recognition,
Clustering**

# THANK YOU