



TEAM

Xingyu Li / Jiaying Wu / Yanyun Qi

ROLE

Industrial design / design ui&ux / user researcher /
 programmer / make prototype / Mechanical design
 / IOT designer / Design service flow

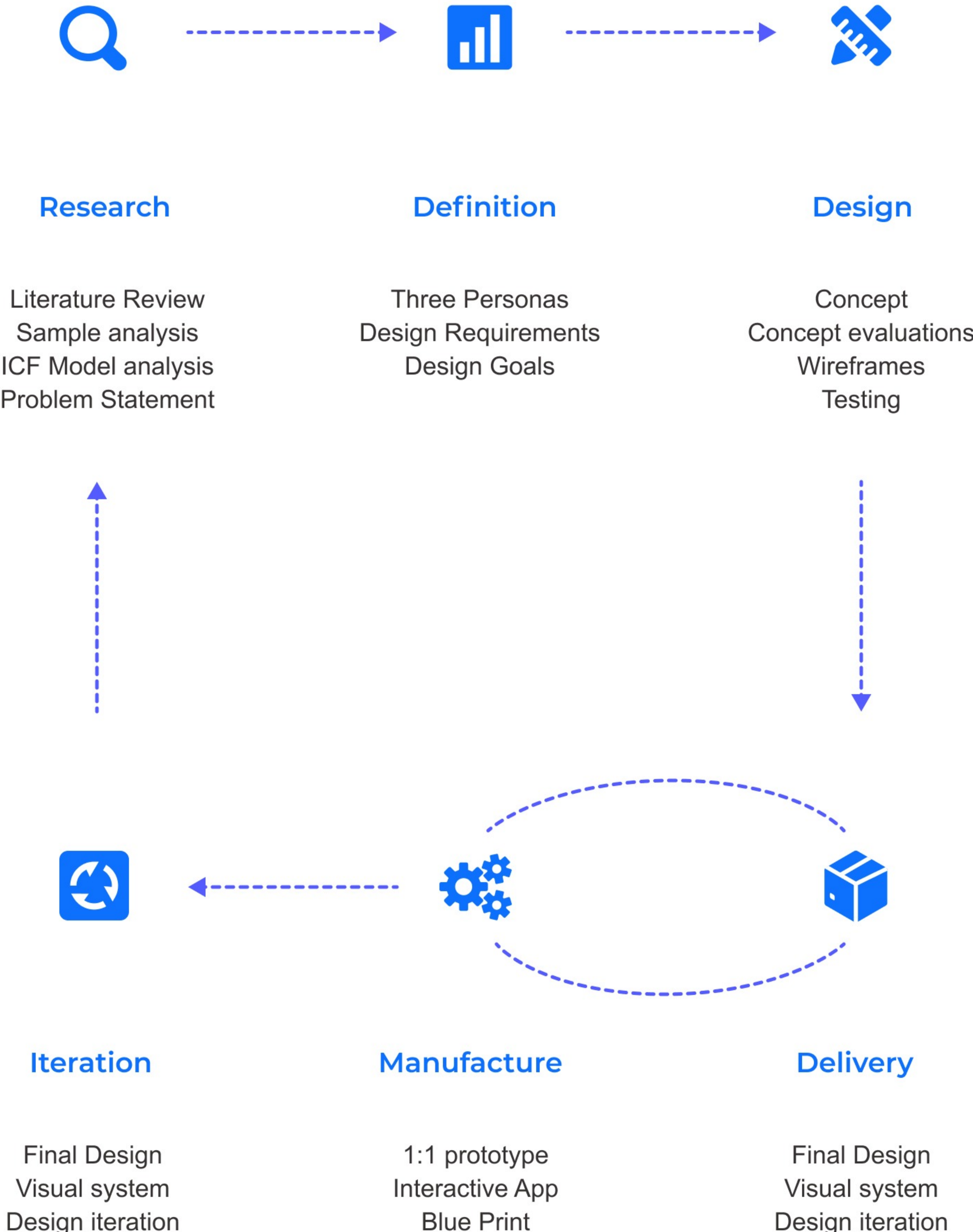
2021.3 - 2021.5

Tilling Goodness

**Product Design / UI&UX Design / Service Design / IOT /
 Accessible Design / Universal Design**

In this project, we focus on the elderly who are facing loneliness problems because of senile diseases or mobility limitations. we design a service system, including an interactive device and an application, for adults above 65 years old with mobility limitations. The system works in a retirement community that tries to solve the lonely problems. We focus on two high-risk diseases of the elderly- MCI and stroke. I design an interactive device, an App, and a service flow to provide a guideline to our service system. Impact by Covid-19, I cannot test the prototypes in person. I will do it later and refine the product based on the users' feedback.

PROJECT OVERVIEW

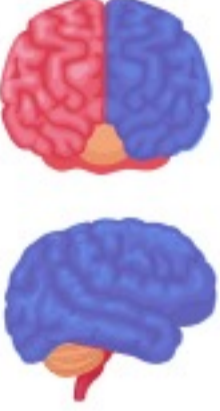


BACKGROUND

Mild cognitive impairment (MCI) and Stroke are two diseases that mainly lead to disability of the elderly over 60 years old.

Stroke

Take the right stroke as an example



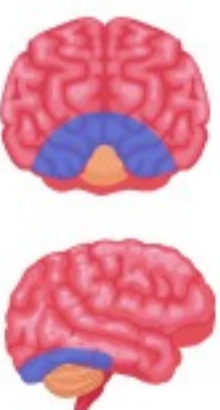
A stroke, sometimes called a brain attack, occurs when something blocks blood supply to part of the brain or when a blood vessel in the brain bursts.

Stroke Statistics

- High mortality rate**
1 in 6 deaths from cardiovascular disease [1]
- High Incidence**
Each year, more than 795,000 people in the United States. [2]
- High cost**
Stroke-related costs in the United States came to nearly \$53 billion each year. [2]
- High Risk**
62% of people hospitalized for stroke were over than 65 years old. [2]

MCI

Mild Cognitive Impairment



An early stage of memory loss or other cognitive ability loss (such as language or visual/spatial perception) in individuals who maintain the ability to independently perform most activities of daily living.

MCI Statistics

- High risk**
About one-third of them develop dementia.
- A lot of people**
Approximately 12% to 18% of people age 60 or older are living with MCI

Stroke Effects

- Left-sided weakness or paralysis and sensory impairment
- Denial of paralysis or impairment (Commonly referred to as "the left neglect")
- 👁️ Visual problems, including an inability to see the left visual field of each eye
- 💡 Spatial problems such as depth perception or directions, such as up or down and front or back
- 🧠 Memory problems
- 🚶 Behavioral changes, such as lack of concern about situations, impulsivity, inappropriateness, and depression

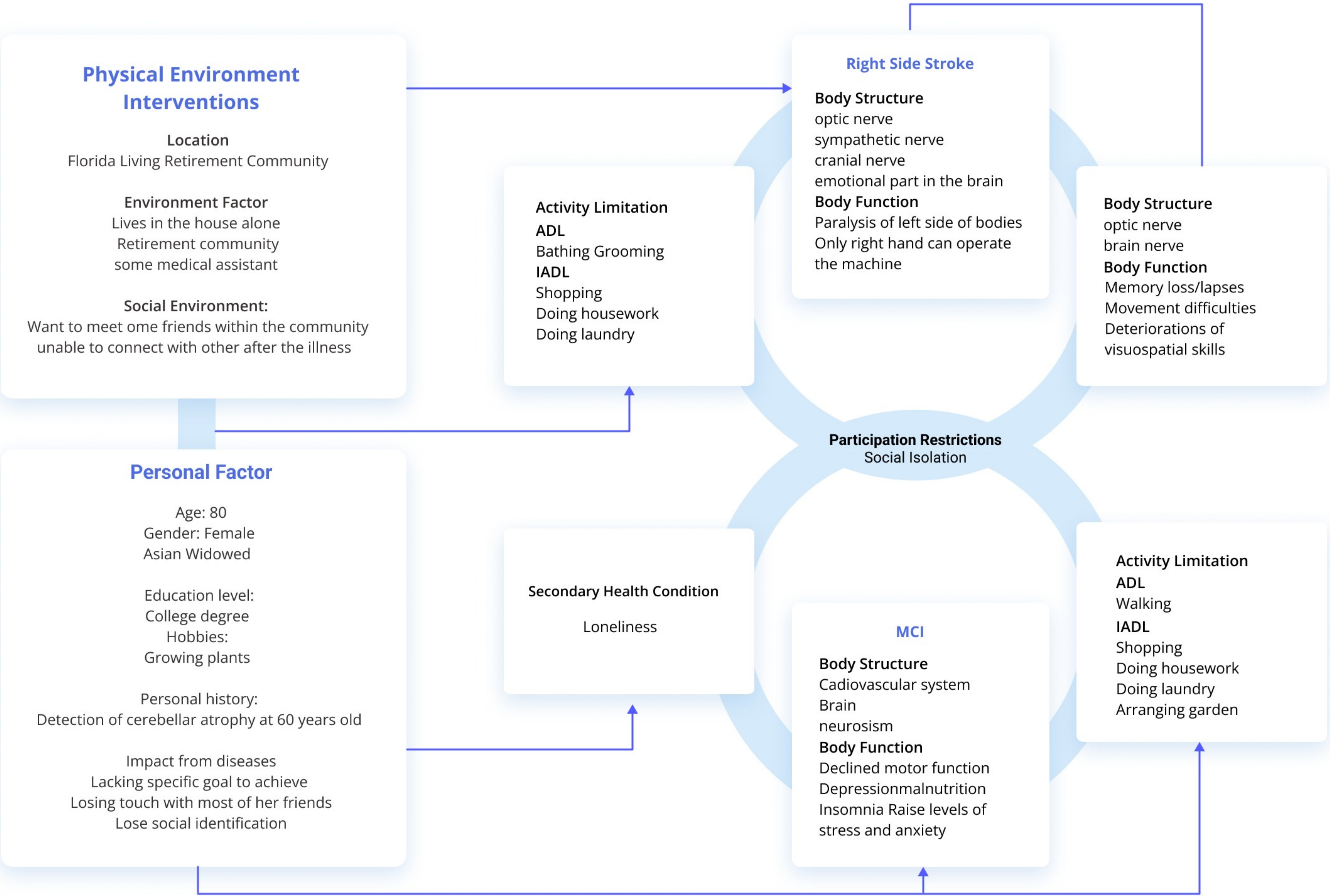
MCI Effects

There are two main types of MCI: **Amnestic MCI** and **Nonamnestic MCI**

Amnestic MCI: This type of MCI primarily primarily affects memory. A person may start to forget important information that he or she would previously have recalled easily, such as appointments and recent events.

Nonamnestic MCI: This type of MCI affects thinking skills, including the ability to make sound decisions, determine the time or sequence of steps needed to complete a complex task, and visual perception.

ICF MODEL



PROBLEM STATEMENT

People with right side stroke or MCI often experience difficulties communicating with others for their mobility impairments. Apart from that, they cannot maintain complex social interaction. To better investigate this problem, we developed a persona to help us learn target user. Paralysis of the left face will impact user's pronunciation which leads to the vague expression, and memory loss can lead to loss of information when she communicates with others. The above symptoms will not only limit her opportunities to engage in face-to-face communication with others, but also greatly reduce her time to interact with potential friends. These can in turn result in the feeling of loneliness and thus lead to mental health problems like depression and anxiety. Eventually, it will lead to the loss of motivation to finish the tasks in life, which lead to the lack of purpose in life.

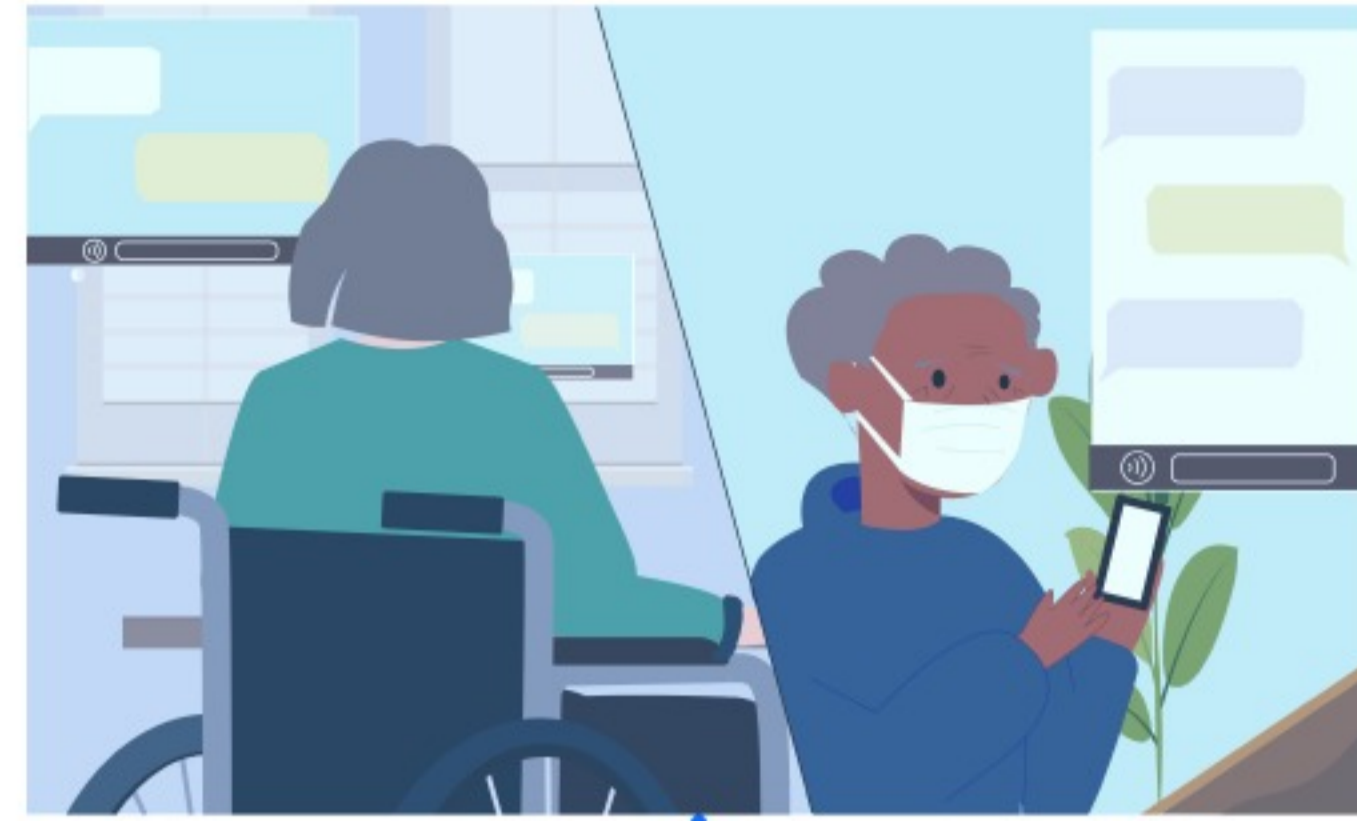
Design Criteria

- Universal Design Requirements**
- Equitable use:** Accommodate standing and sitting operation, Adjustable, Encourage natural interaction between her and the device/app
- Simple and intuitive:** Tiers of information needs to follow people's recognition, Icon guidance, Readable and understandable interface, Short learning curve
- Flexibility:** Both hands can be used, Can use voice control, No extensive gripping or extensive fine hand use
- Perceptible Information:** Image, icon, and voice functions are considered, Size fonts, Obvious feedbacks
- Tolerance & Errors** Avoid accident touching, Clear guidance
- Low physical efforts** Can be operated by one hand, Modularized component can be easily taken off, Easy to assemble, Easy to maintain, Do not require high precision and force operation, Portability, Easy to transfer planting pods
- Other Design Requirements Function achievement:** Establishing her social identification and self-esteem, Encourage interaction between the persona and others of her community (care staff, neighbors, etc.), Give her a drive/focus for her everyday life, Provide topics between her and her friends
- Memorability:** Does not require repeated learning, Material, shape and form, Use safety materials (non-toxic, no shards, etc.), Avoid sharp edge, Parts are easily to be replaced (avoid loss due to amnesia)

Service Blueprint

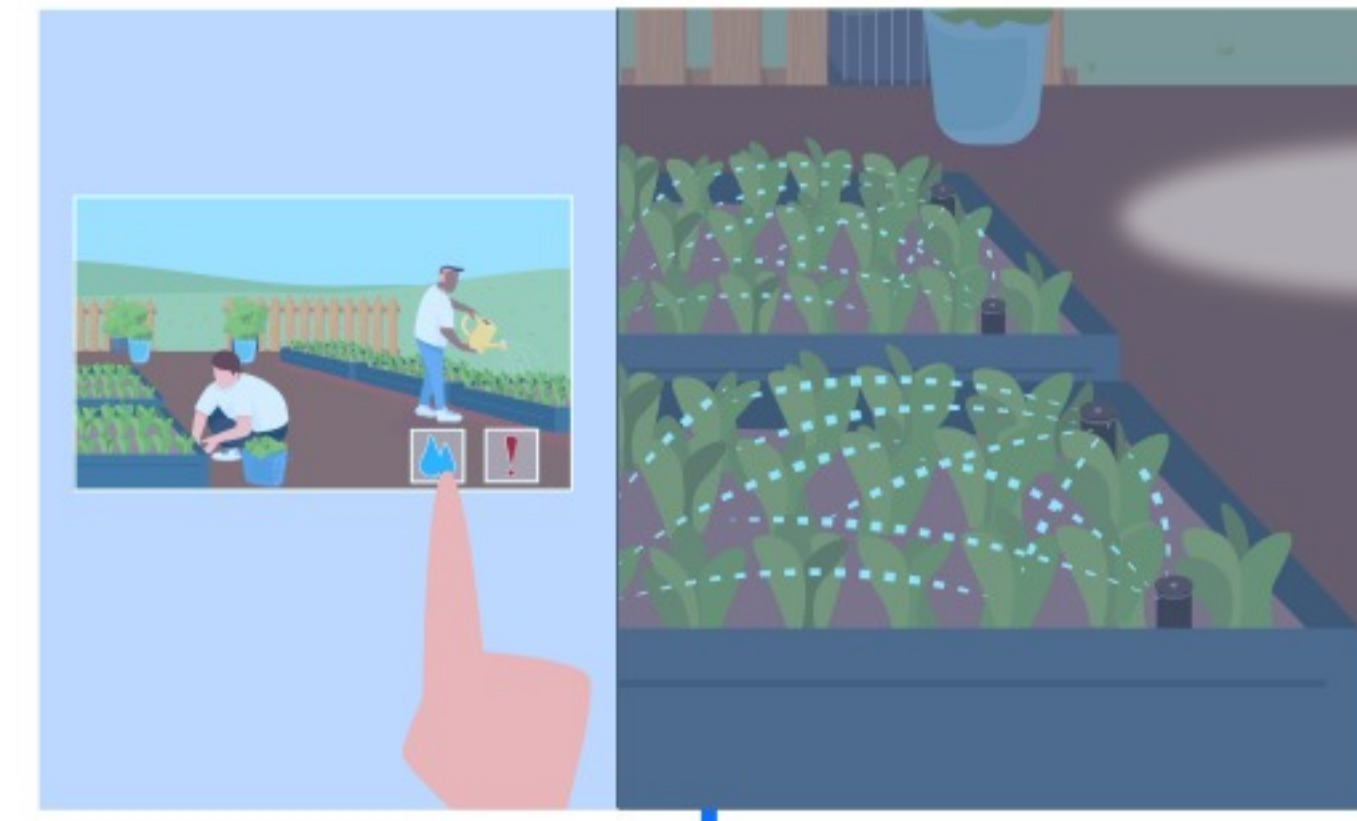
Friends Helping

Users can voice message others to communicate with them to share their experiences and plants status



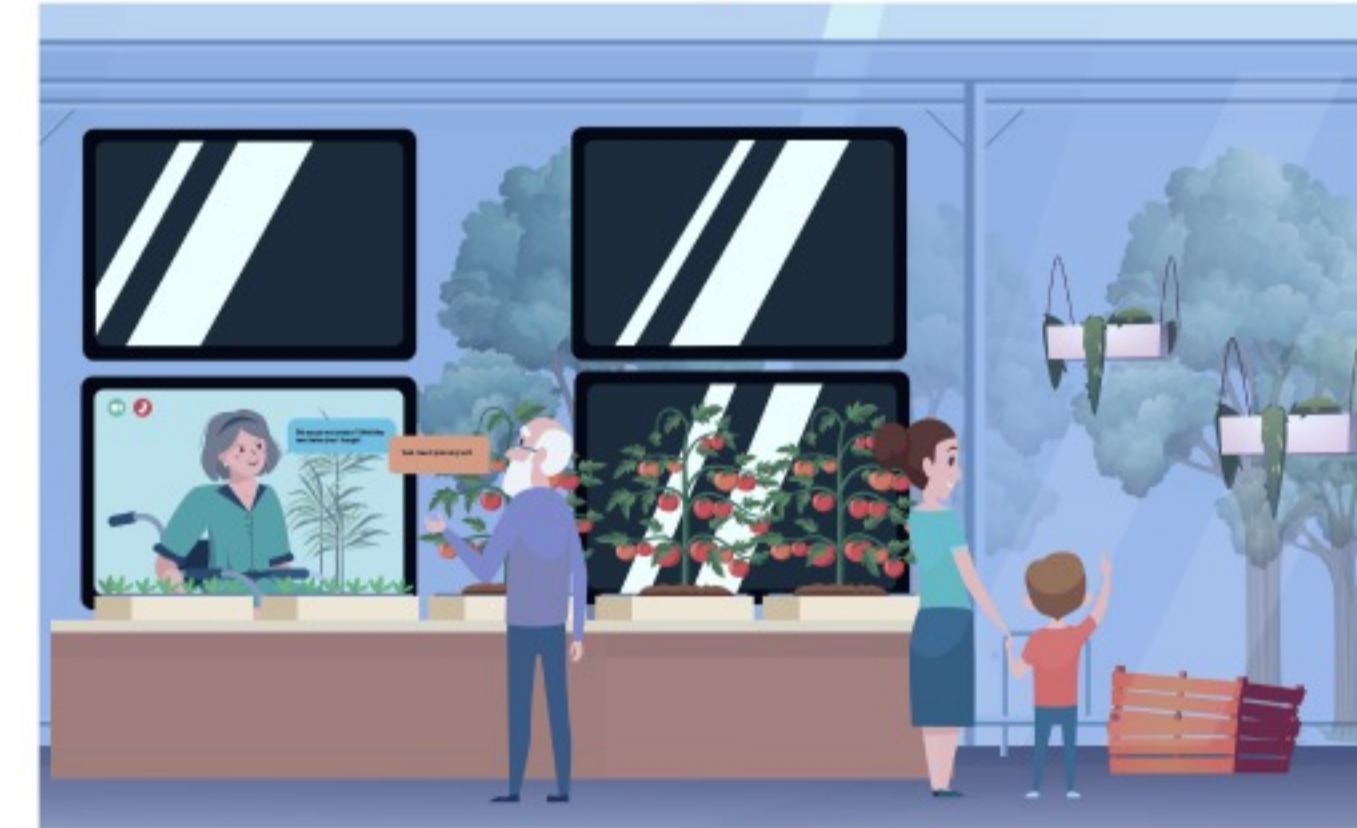
One-click watering

Users at home can also monitor plants in the community garden and watering the plants



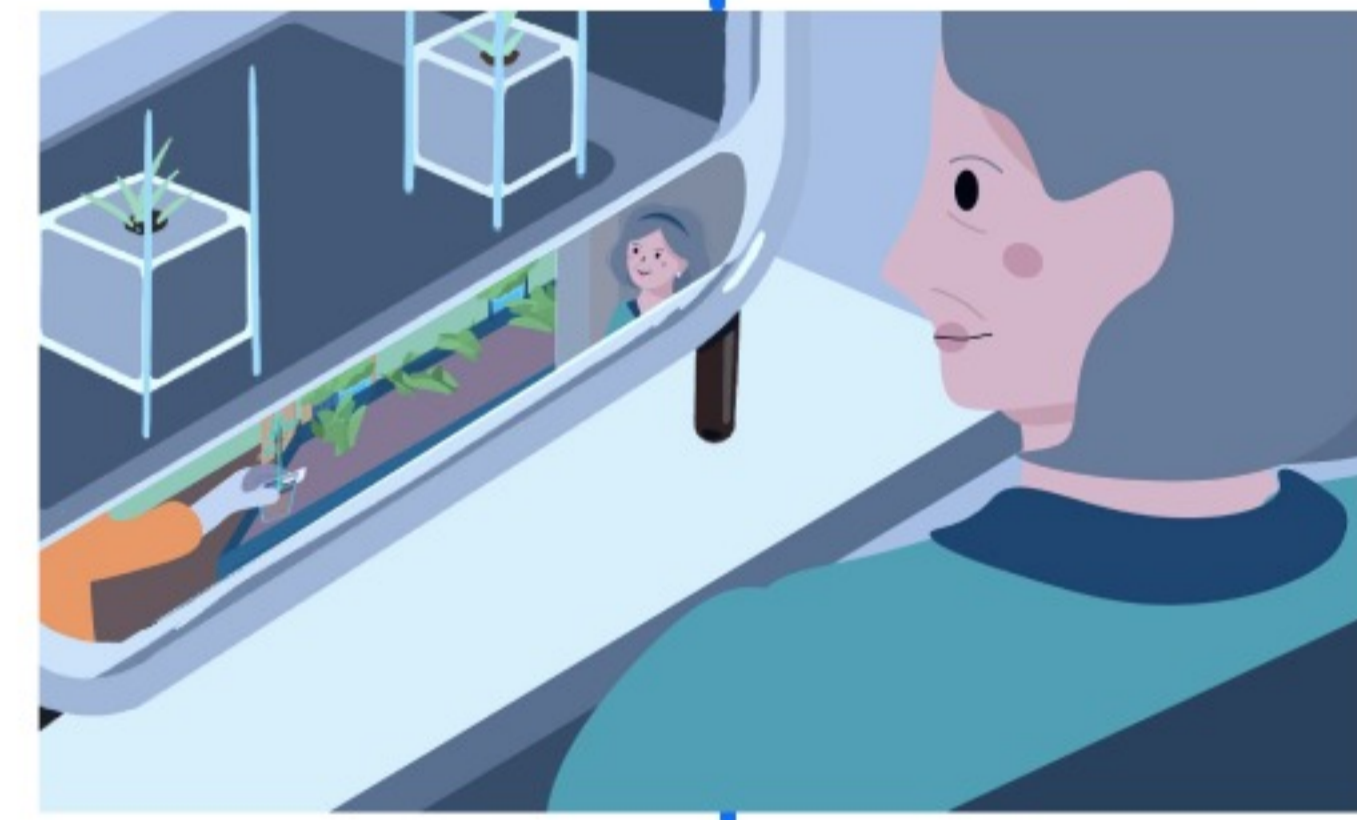
Video Chat

The device can video chat with people in the community garden and get most updated info about their plants



Real-time planting

See their plants growing in the garden



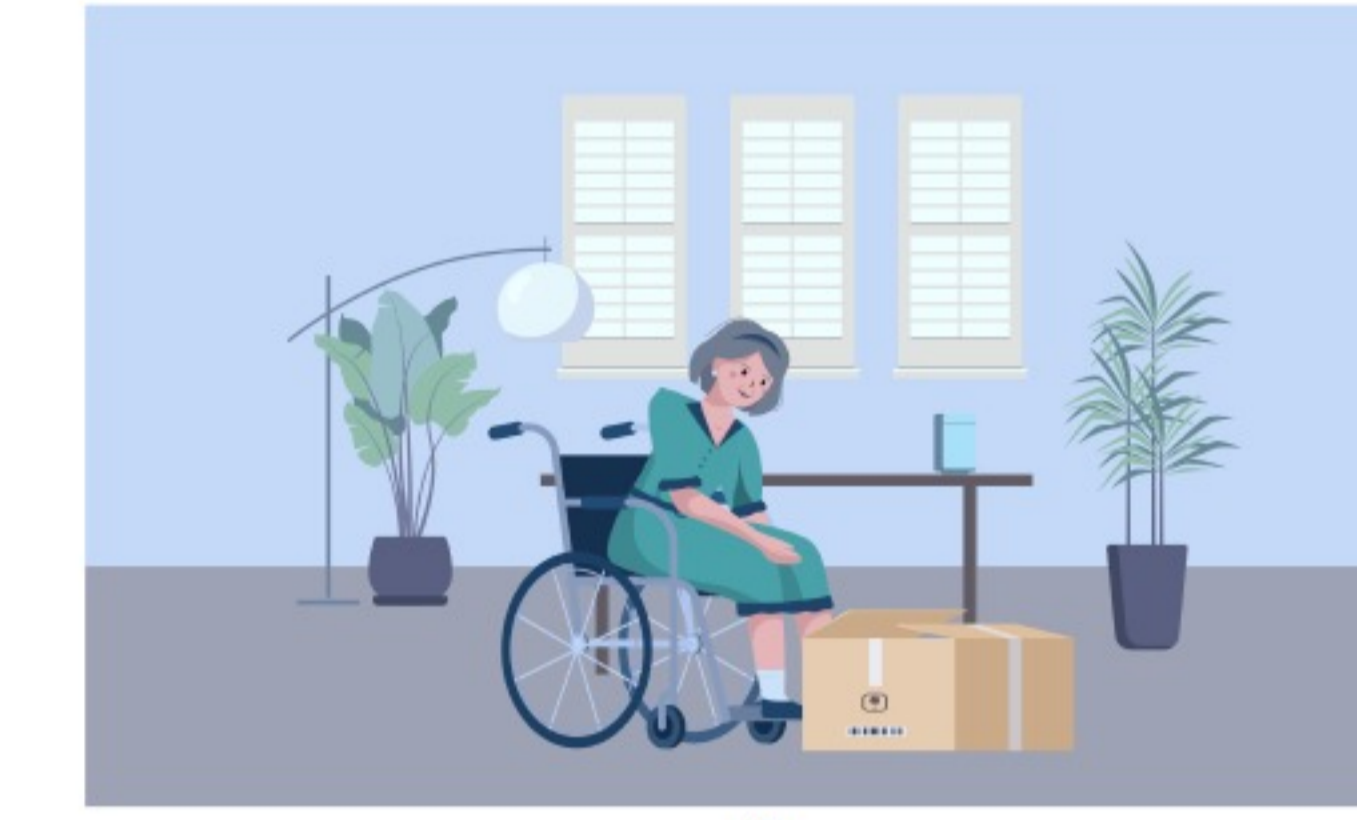
Device Delivery

The Device can be ordered online or in-store, users can choose either to pick it up or have it delivered



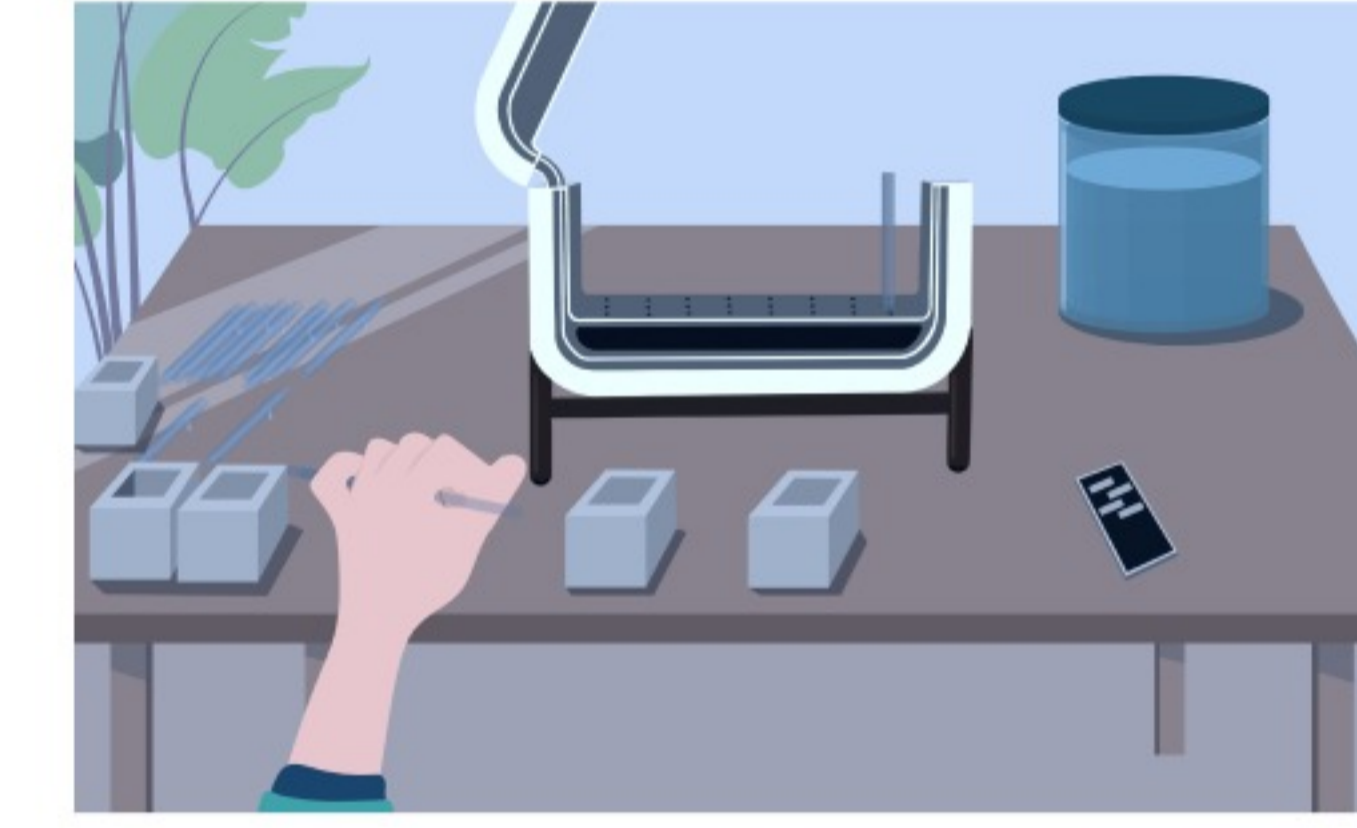
Unpacking

The package is easy and quick to open with only one hand



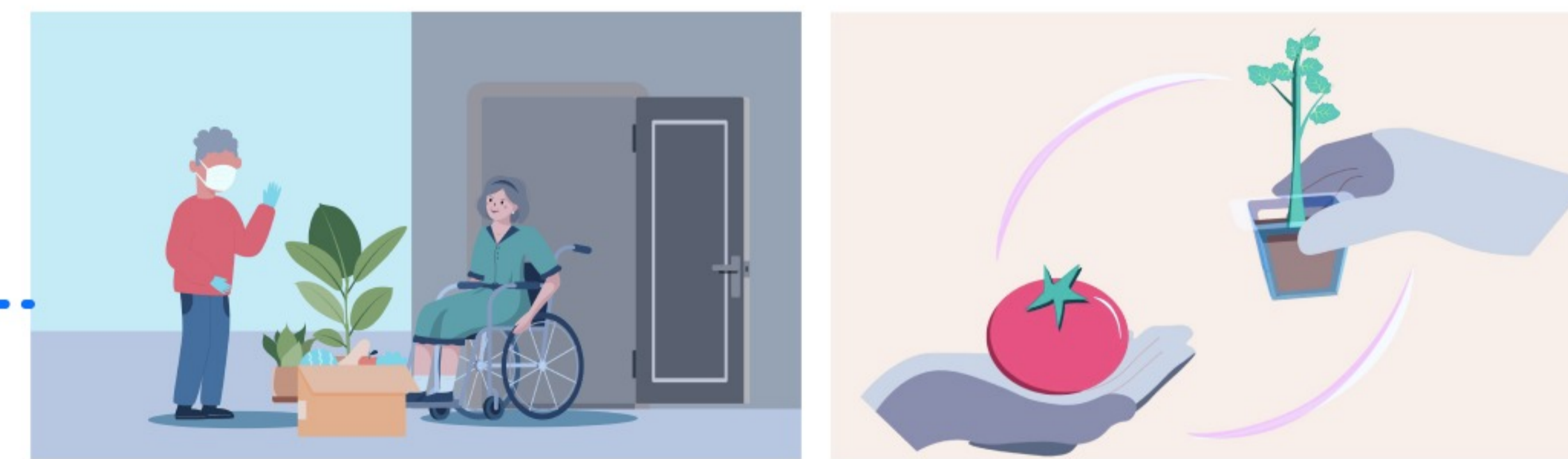
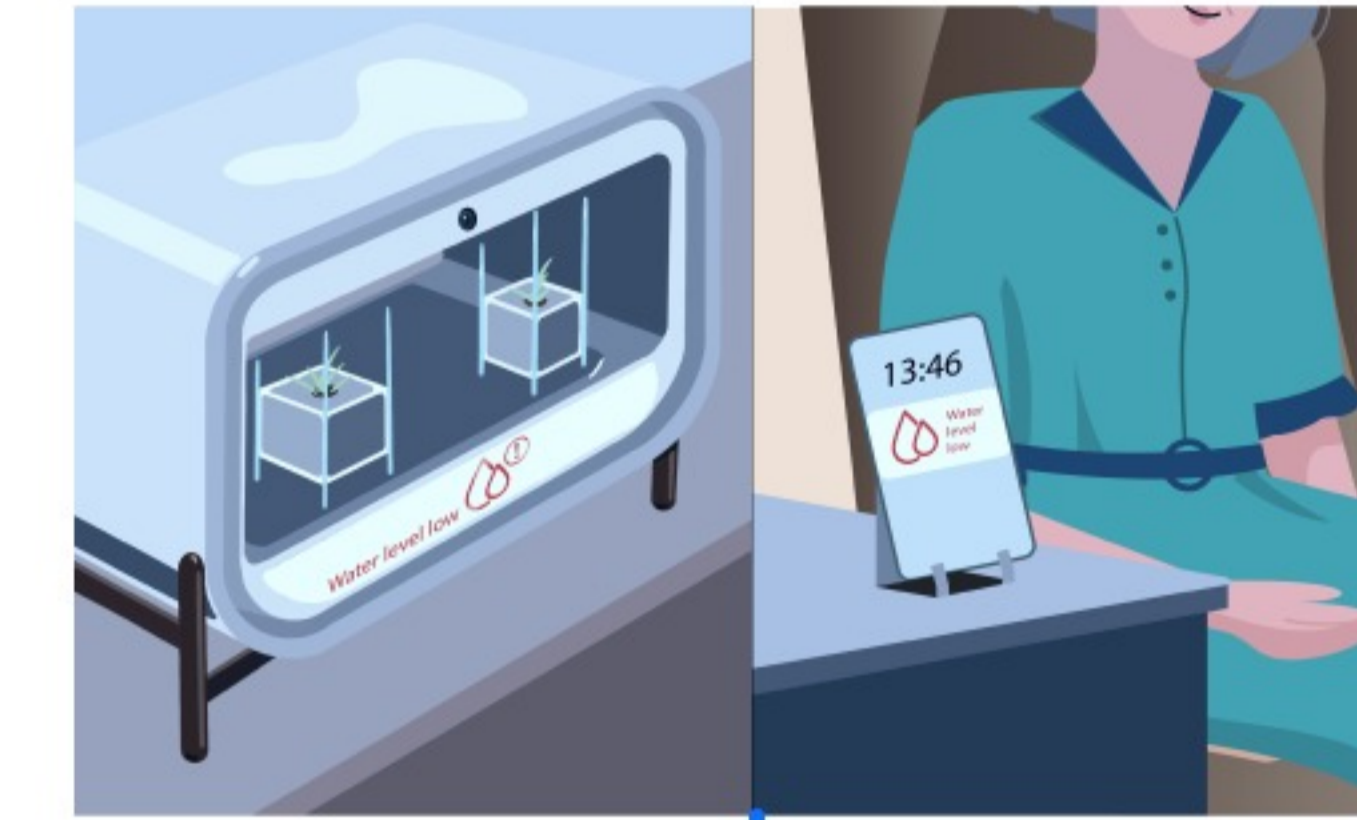
Assemble Device

The device can be easily assembled by one hand and users can choose and arrange the number of the seed kit box based on their will.



Multi-platform reminders

Reminders of caring for plants will be sent out on different platforms simultaneously



Exchange Harvest and Sprout

When fruits and vegetables become ripe, friends can give each other their harvests and exchange the startup seedings.

Design

Outcomes Blueprint

APP

- Control the device
- Send reminder to users
- Help communicate with friends
- Monitor the plants

Planting device

- Semi-automatic planting machine
- Indoor planting
- Modularized design

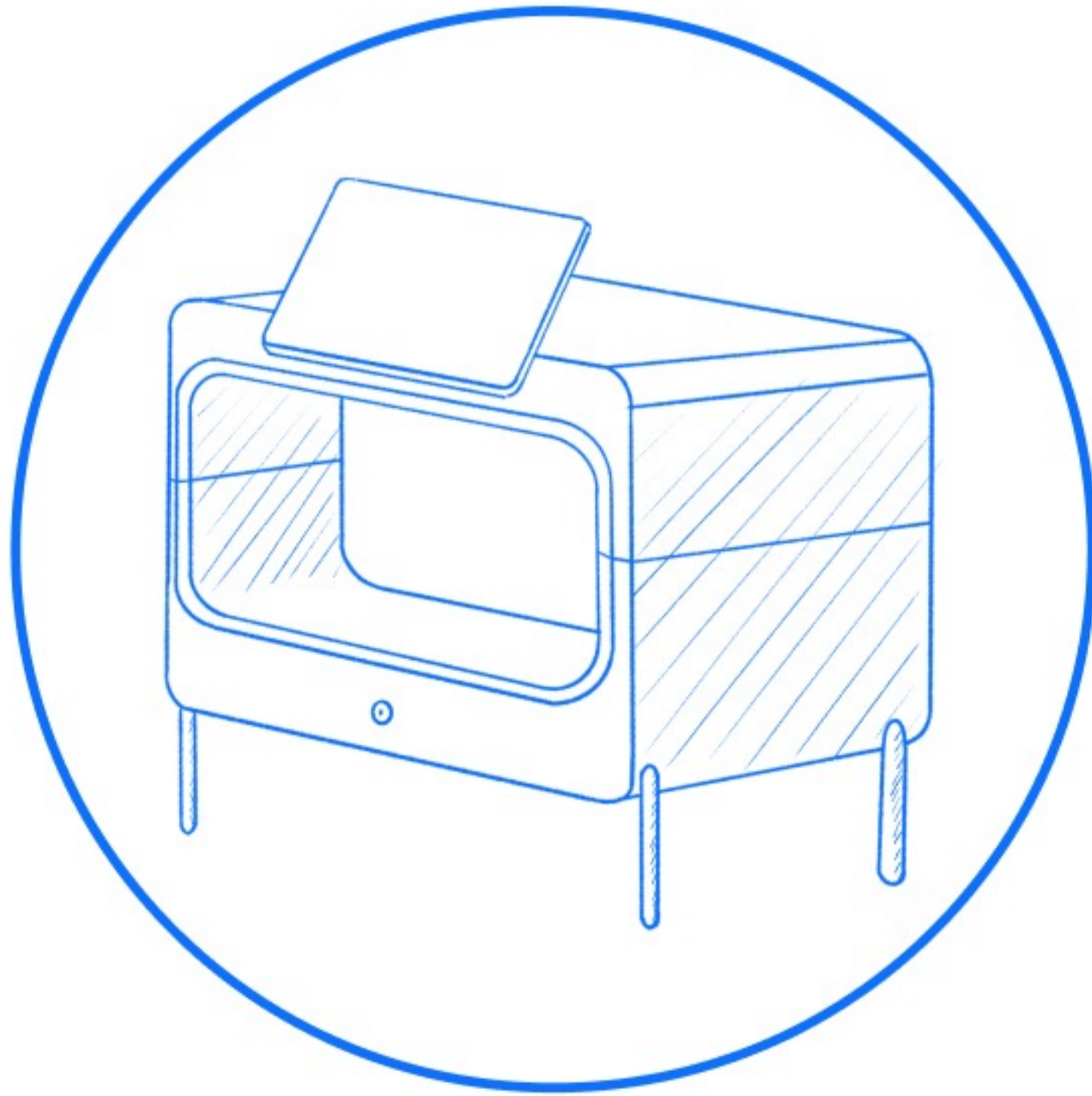
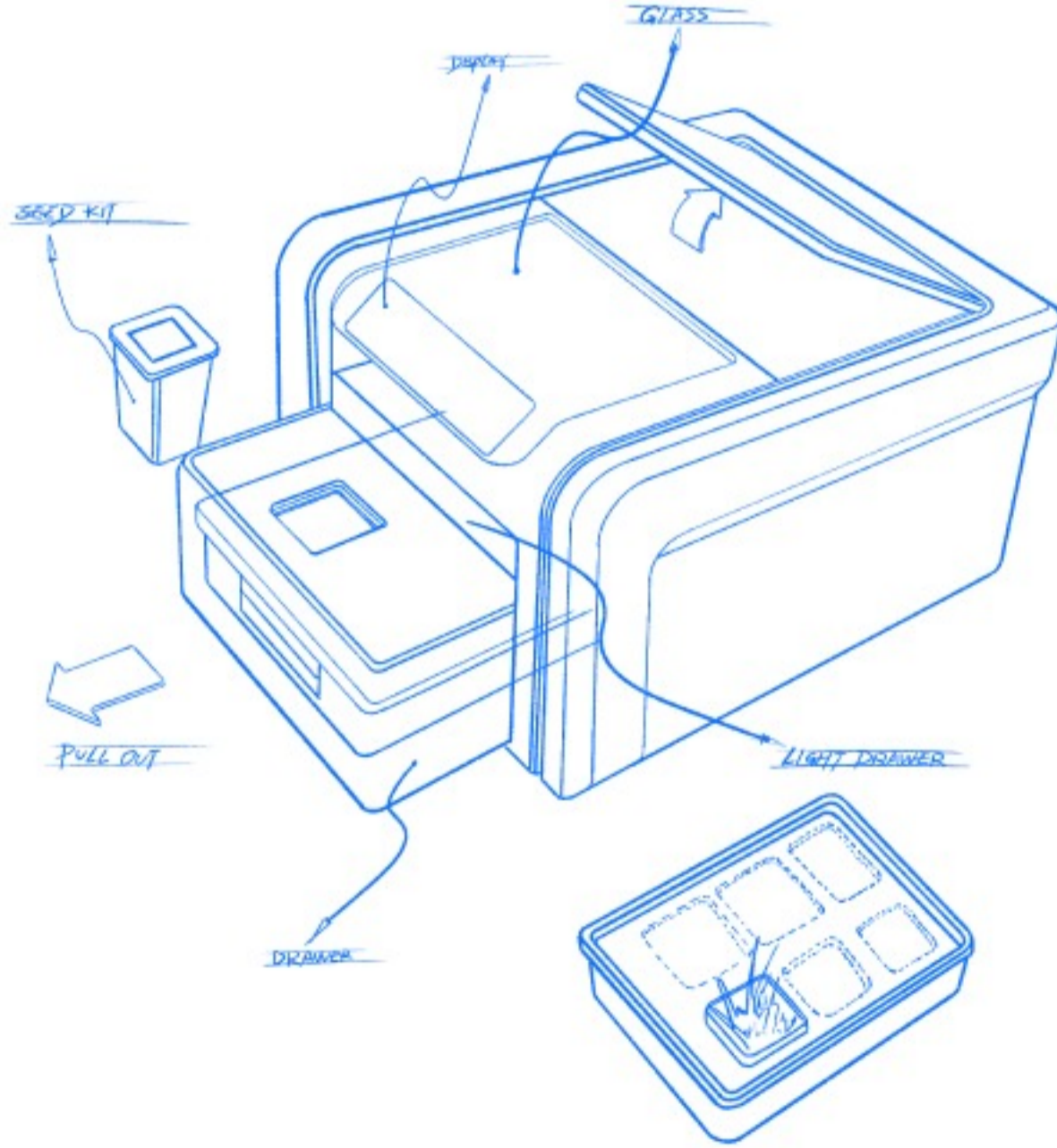
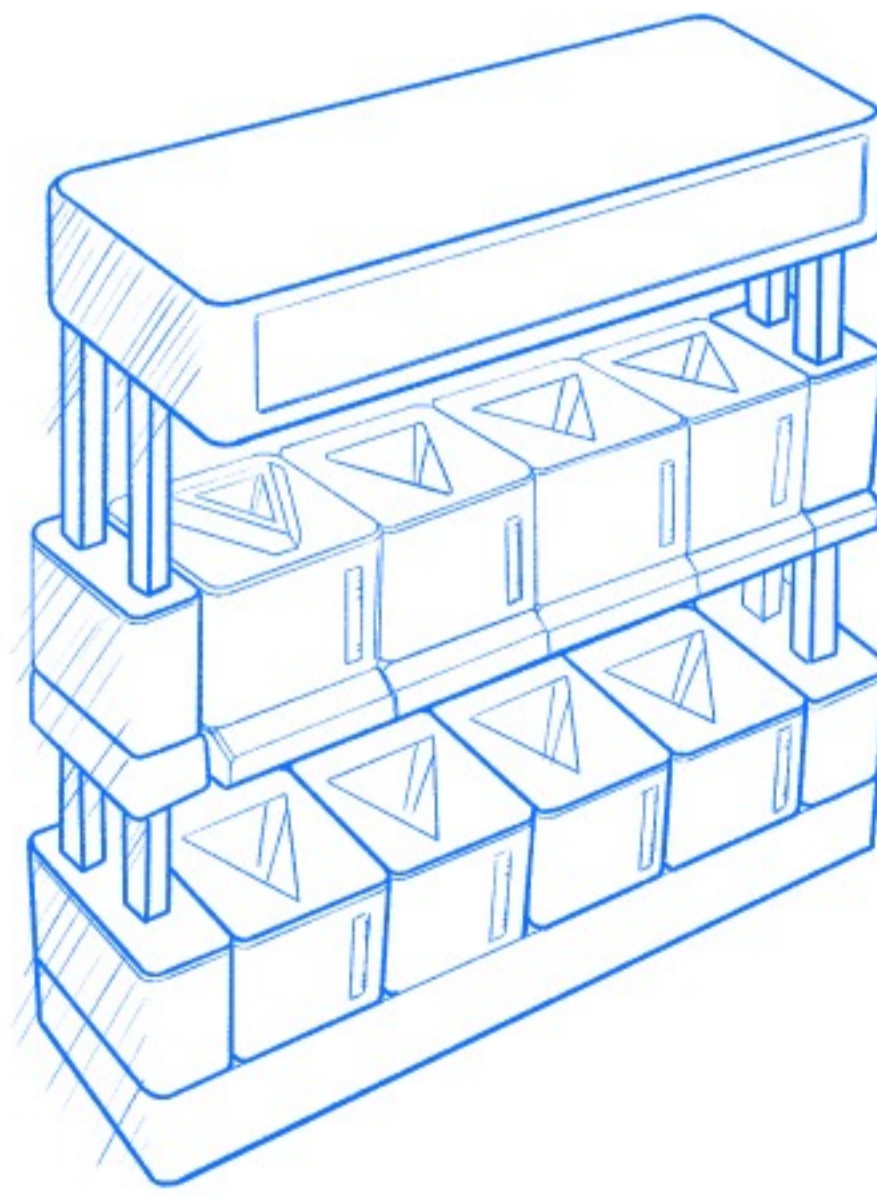
Service

- Planting assistance
- Plants delivery
- Manage the whole system

Depend on the community service abilities

Planting Device

Concepts

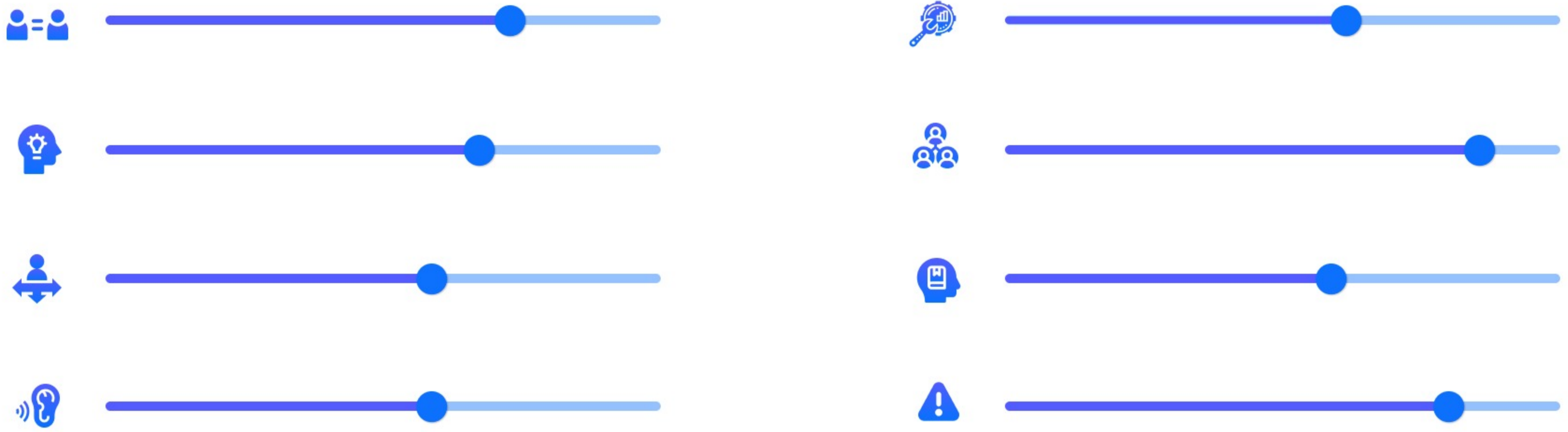


Evaluations

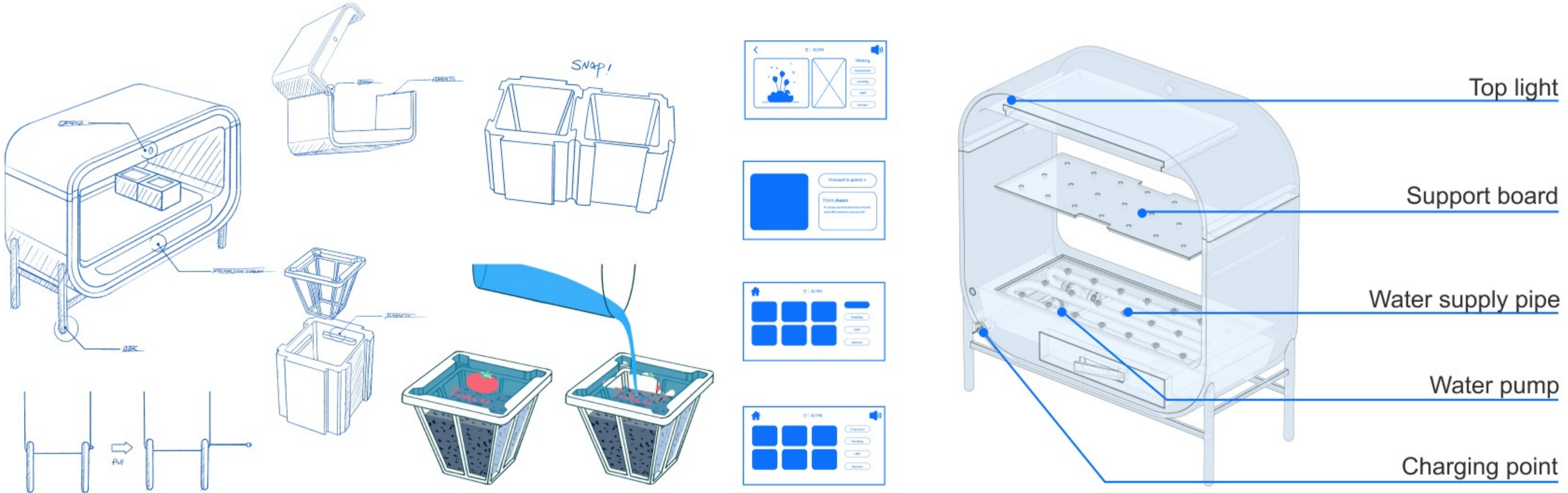
The third concepts meet the most requirements. So, the third concept is chosen and evaluated.

Equitable use Simple and intuitive Flexibility Perceptible Information

Tolerance & Errors Low physical efforts Memorability Other Requirements



Final Concept



Top light

Support board

Water supply pipe

Water pump

Charging point

CASE STUDY




Review the documents of 50 patients with MCI or right stroke

The documents include the information below:

1. Demography (gender, age, race, culture, country)
2. Personal history (medical, education, occupation, family)
3. Geography (residence, living environment)
4. Social (close relationship, social connection)

Based on the documents, we generate the descriptions of the target user.

We describe the target user through:

-  **Persona**
-  **Activity limitation charts (IADL and ADL)**
-  **Living environment**

USER PROFILING

Persona

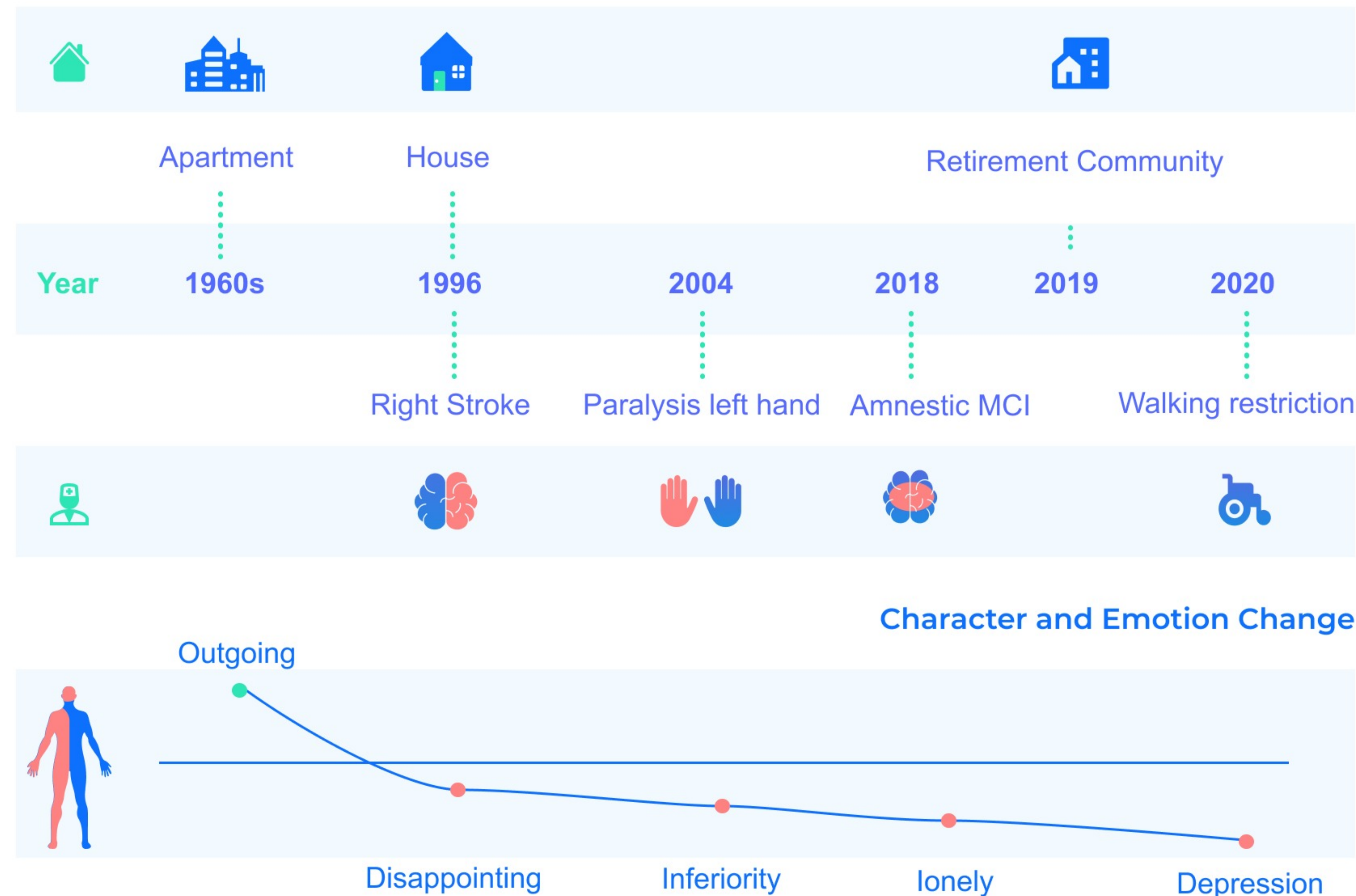


Annie

Age: 70
 Gender: Female
 Education level: college degree
 American Widowed

Hobbies:

Chatting
 Planting
 DIY
 Watching TV
 Cooking with Friends



Activity Limitation Charts (IADL & ADL)

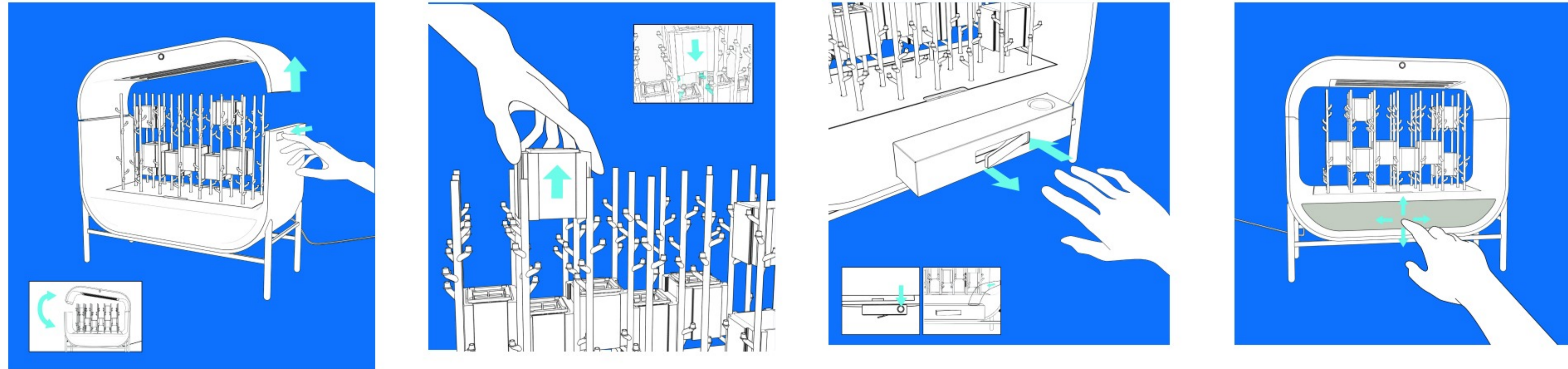
ADL

ADL functions	Independent	Needs Help	Dependent	Can not do
Bathing	●			
Dressing	●			
Grooming		●		
Mouth Care	●			
Toileting		●		
Transferring			●	
Walking	●			
Climbing stairs		●		
Eating	●			

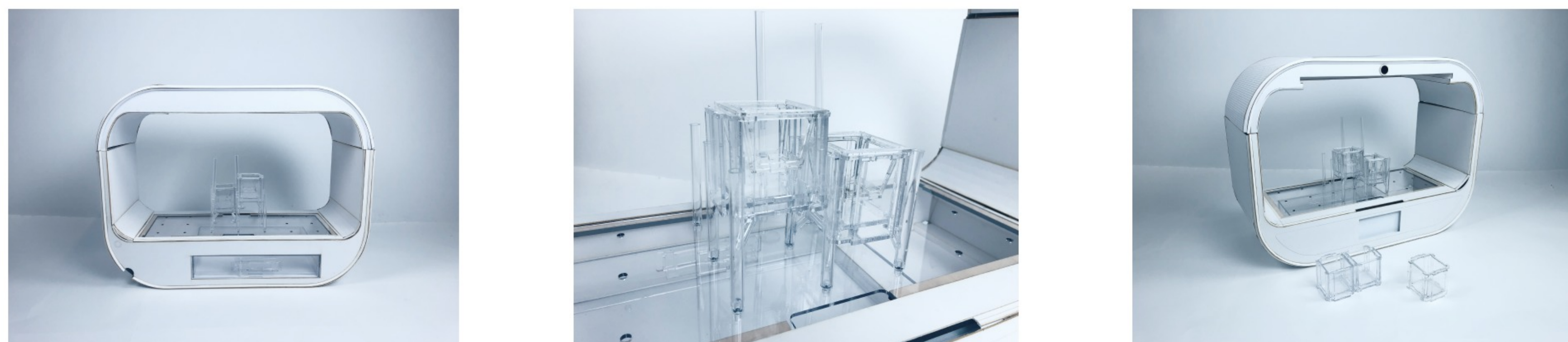
IADL

IADL functions	Independent	Needs Help	Dependent	Can not do
Cooking		●		
Managing medications		●		
Using the phone and looking up numbers	●			
Doing housework		●		
Doing laundry		●		
Driving or using public transportation			●	
Managing finances		●		
Shopping		●		

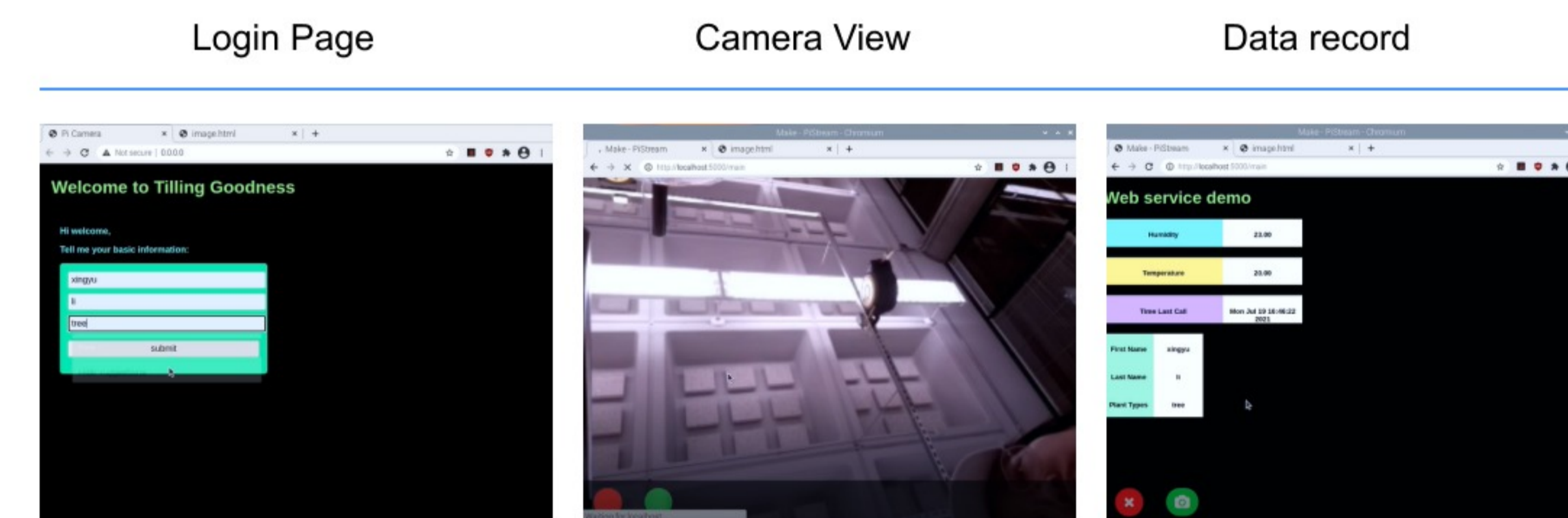
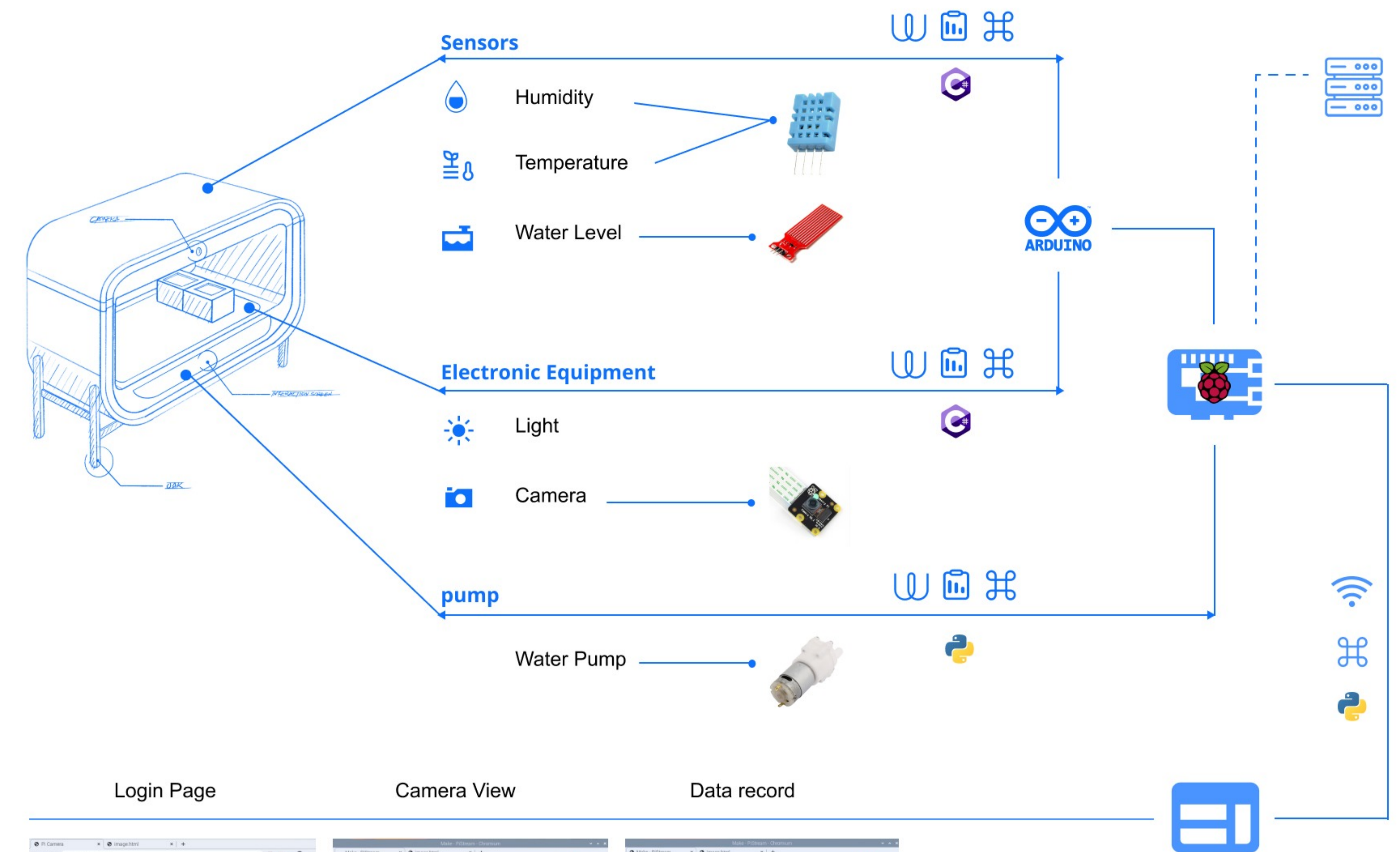
User Flow



Rendering & Prototype



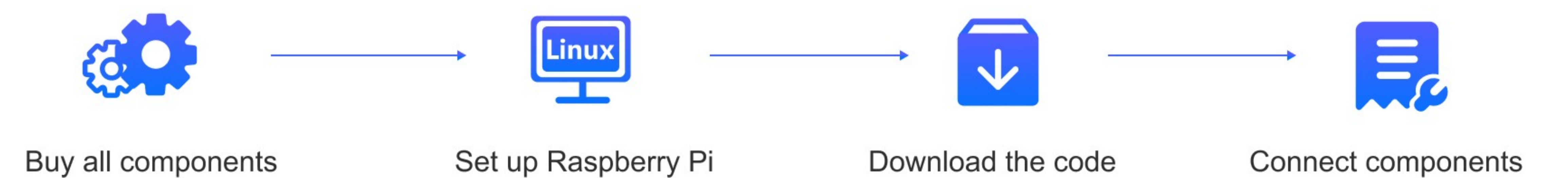
IOT & Technology achievement Done by myself



Coding & Manufacturing

Download the code from link : https://github.com/xingyuli1998/pi_test.git

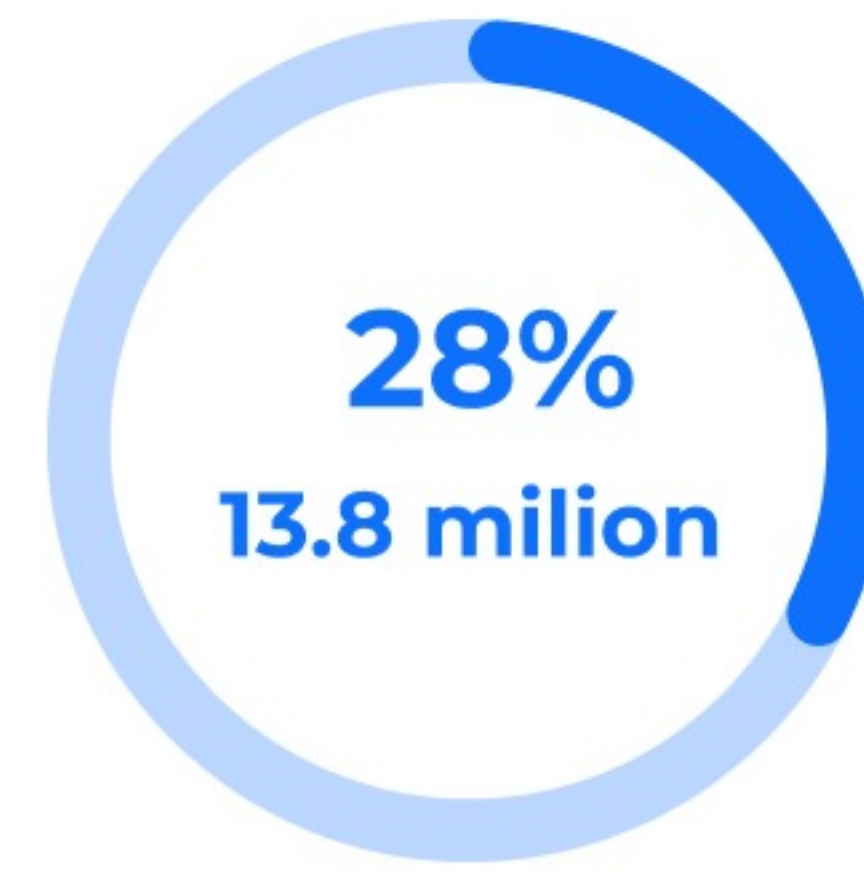
Steps to set up the device:



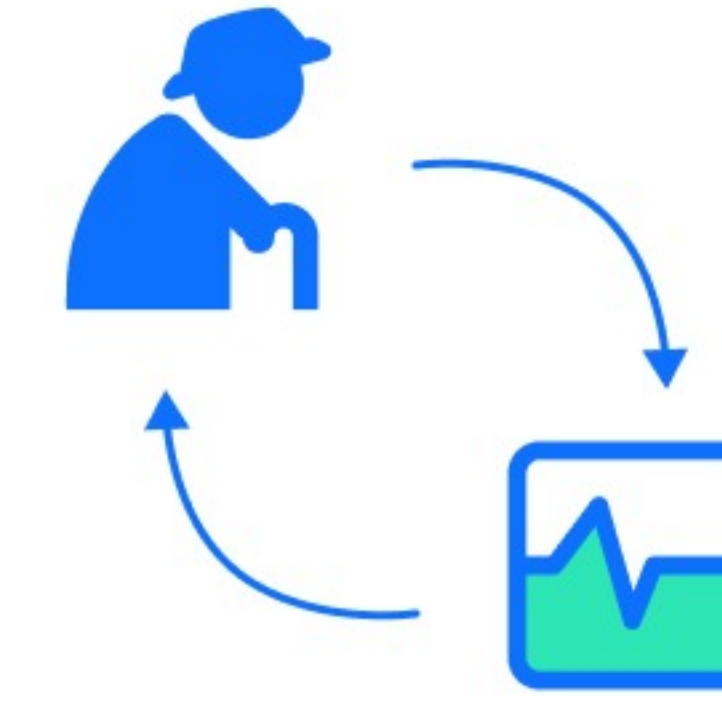
Living Environment



LONELINESS








About 28% of adults aged over 65 years are considered to be lonely/ experience some degree of loneliness, which is approximately 13.8 million people.






Moreover, loneliness has been found to cause various health problems.(including mental health)

Retirement Community

There are three main types of buildings in the community that can provide the service below. The functions of the community is very important for building the service system.

 Housing	 Integrated Communities	 Care Homes
<p>Known as sheltered housing, Retirement flats or communities</p>	<p>Retirement villages, Housing-with-cares, Assisted livings or Independent livings</p>	<p>Nursing Homes, Residential Homes, Old People's Homes</p>
<p> • Part-time wardens • Emergency call systems</p>	<p> • 24-hour onsite staff • Care or domiciliary service • Restaurant / Cafe</p>	<p> • 24-hour care • Meals</p>
<p>Facilities</p> <ul style="list-style-type: none"> • Communal lounge • Gardens • Laundry facilities • Guest room 	<p>Facilities</p> <ul style="list-style-type: none"> • Hobby rooms • Gardens • Leisure Club • Library • Guest room • Hair Salons • Social event programme • Communal lounge • exercise room 	<p>Facilities</p> <ul style="list-style-type: none"> • Dining room • Activities • Communal lounges • Gardens

Competitive Analysis

Product	Target User	Functions	Strengths	Weakness
	<ul style="list-style-type: none"> • all ages • feel lonely 	<ul style="list-style-type: none"> • cultivating virtue plants • humidifier • listening to music 	<ul style="list-style-type: none"> • multi-functions • elegant outlook • meet needs of introverts 	<ul style="list-style-type: none"> • hard to build up connections with new friends
	<ul style="list-style-type: none"> • elders • feel lonley 	<ul style="list-style-type: none"> • talk with families • event suggestions • Send notifications • friend suggestions 	<ul style="list-style-type: none"> • intensive functions • elegant outlook • Age-friendly UI-UX design 	<ul style="list-style-type: none"> • small screens • hard to meet people face to face • not friendly to elders with limited activities
	<ul style="list-style-type: none"> • elders • feel lonley 	<ul style="list-style-type: none"> • talk with families • robots assistance 	<ul style="list-style-type: none"> • intensive functions • elder friendly UI/UX design 	<ul style="list-style-type: none"> • limited mobility • hard to meet people face to face

Interview points:

The hardest parts for elders to meet new friends are that:

- Little common topics or interests
- Mobility impariment
- Limited ways to meet friends
- Unfamiliar with social media

Language used here:



The detailed descriptions of codes are included in the README file. The following images show the parts of the code.

```

#Modified by smartbuilds.io
#Date: 27.09.20
#Desc: This web application serves a motion JPEG stream
# main.py
# Import the necessary packages
from flask import Flask, render_template, Response, request, url_for, redirect
from camera import VideoCamera
import time
import threading
import os
from index import Arduino_sensor

pi_camera = VideoCamera(flip=False) # flip pi camera if upside down.

class VideoCamera(object):
    def __init__(self, flip = False):
        self.ps = PIVideoStream()
        self.vs = self.ps.start()
        self.flip = flip
        time.sleep(2.0)

    def __del__(self):
        self.vs.stop()

    def flip_if_needed(self, frame):
        if self.flip:
            return np.flip(frame, 0)
        return frame

    def get_frame(self):
        frame = self.flip_if_needed(self.vs.read())
        ret, jpeg = cv2.imencode('.jpg', frame)
        return jpeg.tobytes()

    def get_picture(self):
        self.ps.capture()

# App globals (do not edit)
app = Flask(__name__)
app.route('/login', methods = ['POST', 'GET'])
def login_index():
    return render_template('login.html')
def login():
    if request.method == 'POST':
        name = request.form['name']
        return redirect(url_for('main'))
    else:
        name = request.args.get('name')
        return redirect(url_for('main'))

def index_login():
    while True:
        name = request.form.get('name')
        name = request.form.get('name')
        plantype = request.form.get('plantype')
        return frame, name, plantype

@app.route('/main', methods = ['POST', 'GET'])
def index():
    timelov = time.asctime( time.localtime(time.time()) )
    hum_value, ten_value = get_data()
    frame, name, plantype = index_login()
    return render_template('index.html', hum_value = hum_value, ten_value = ten_value, time_value = timelov, frame = frame, name = name)

def gen_camera():
    get camera frame
    while True:
        frame = camera.get_frame()
        yield b'--frame\r\n'
        b'Content-type: image/jpeg\r\n\r\n' + frame + b'\r\n\r\n'

# include <DHT.h>
#define DHTPIN A2
#define DHTTYPE DHT11
int sensorValue = 0;
const int sensorPin = A0;
int RelayPin=13;
DHT dht(DHTPIN, DHTTYPE);
void setup() {
    Serial.begin(9600);
    pinMode(13,OUTPUT);
    //digitalWrite(13,LOW);
    pinMode(sensorPin, INPUT);
    dht.begin();
}

void loop() {
    // Wait a few seconds between measurements.
    delay(500);
    sensorValue = analogRead(sensorPin);
    Serial.print(sensorValue);
    if(sensorValue < 550){
        digitalWrite(13,HIGH);
        float h = dht.readHumidity();
        float t = dht.readTemperature();
        if (isnan(h) || !isnan(t) ) {
            Serial.println("Failed to read from DHT sensor!");
            return;
        }
        Serial.println(h);
        Serial.println(t);
        delay(1000);
    }else{
        digitalWrite(13,LOW);
        float h = dht.readHumidity();
        float t = dht.readTemperature();
        if (isnan(h) || !isnan(t) ) {
            Serial.println("Failed to read from DHT sensor!");
            return;
        }
        Serial.println(h);
        Serial.println(t);
        delay(1000);
    }
}

@app.route('/main', methods = ['POST', 'GET'])
def index():
    timelov = time.asctime( time.localtime(time.time()) )
    hum_value, ten_value = get_data()
    frame, name, plantype = index_login()
    return render_template('index.html', hum_value = hum_value, ten_value = hum_value, time_value = timelov, frame = frame, name = name)

def gen_camera():
    get camera frame
    while True:
        frame = camera.get_frame()
        yield b'--frame\r\n'
        b'Content-type: image/jpeg\r\n\r\n' + frame + b'\r\n\r\n'

@app.route('/video_feed')
def video_feed():
    return Response(gen_pi_camera(),
                    mimetype='multipart/x-mixed-replace; boundary=frame')

@app.route('/picture', methods = ['POST', 'GET'])
def picture():
    pi_camera.get_picture()
    return render_template('image.html')

if __name__ == '__main__':
    app.run(host='0.0.0.0', debug=False)

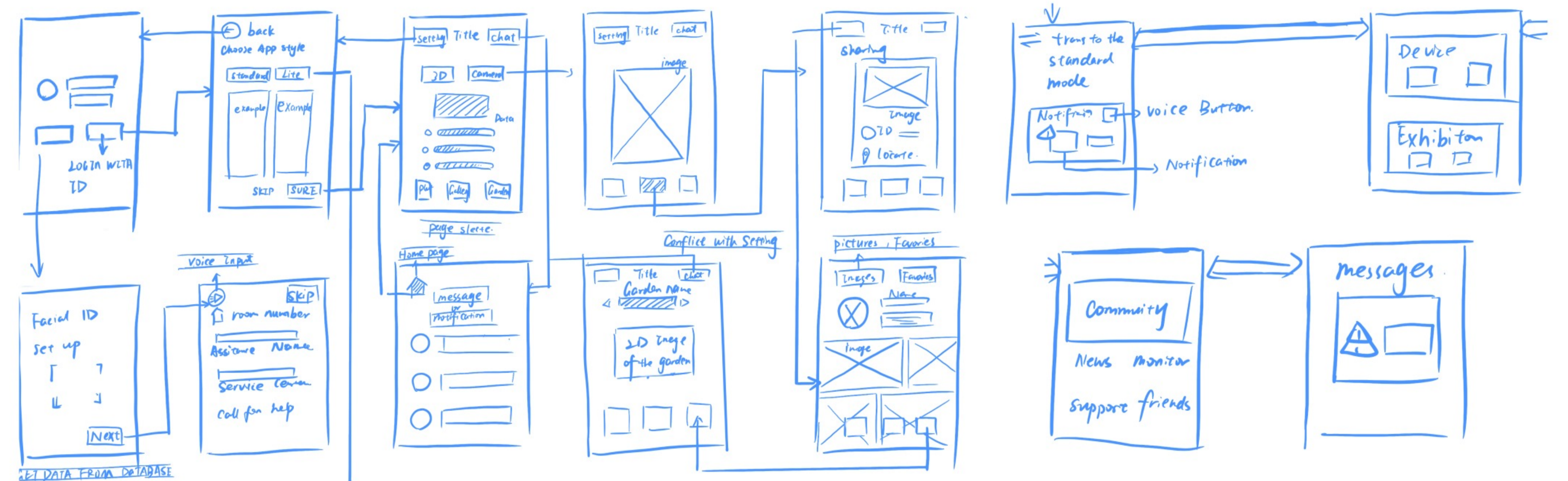
```

Application

First Round Design

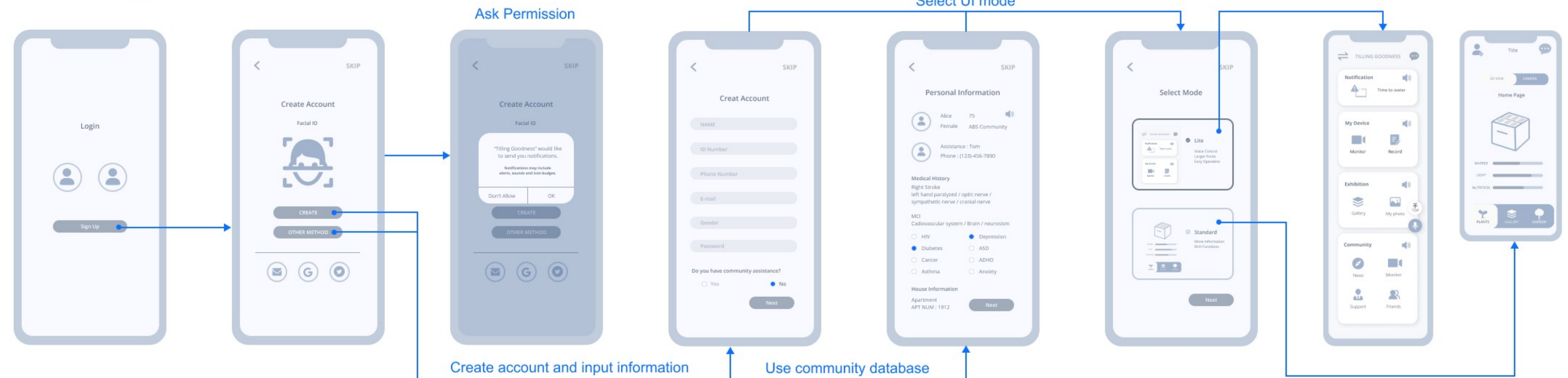
User Flow

Starting by brainstorming and establishing the features the app should have. I went on to explore the user flow through sketches.



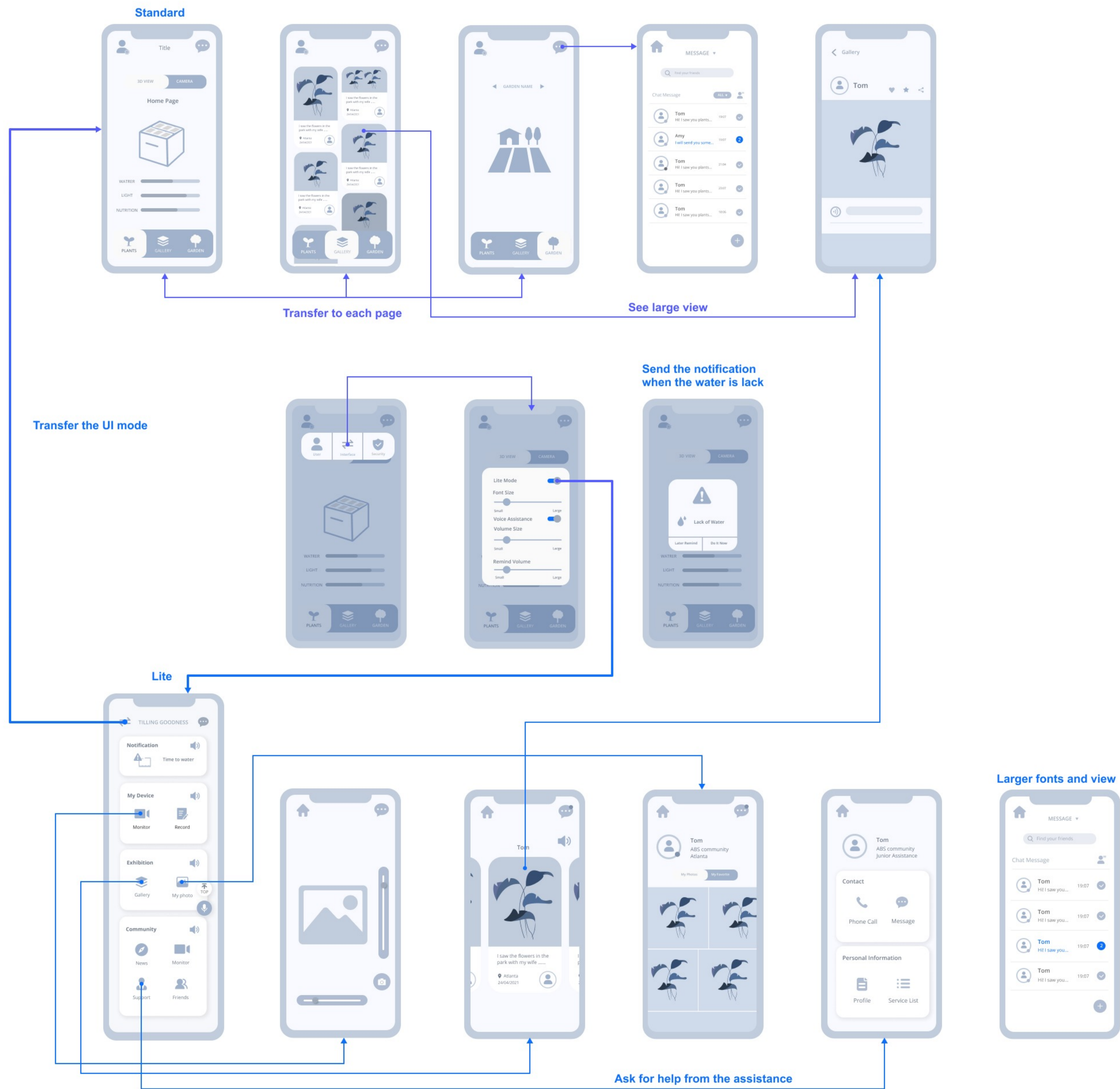
Wireframe

Onboarding



The Demo video link: <https://youtu.be/ZqffacPiGao>

Main Page



UI Kits

Input

Active

Active

Default

Default

Disable

Disable

Error

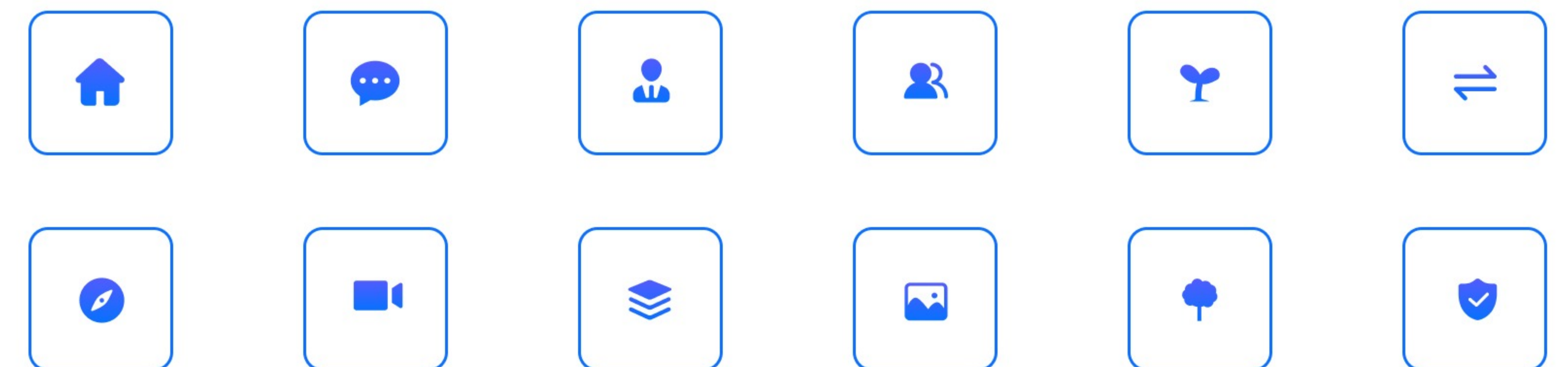
Active

The information input is wrong

Colors

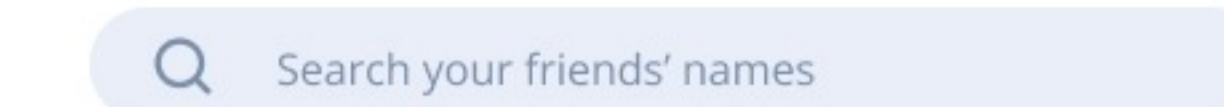


Iconography

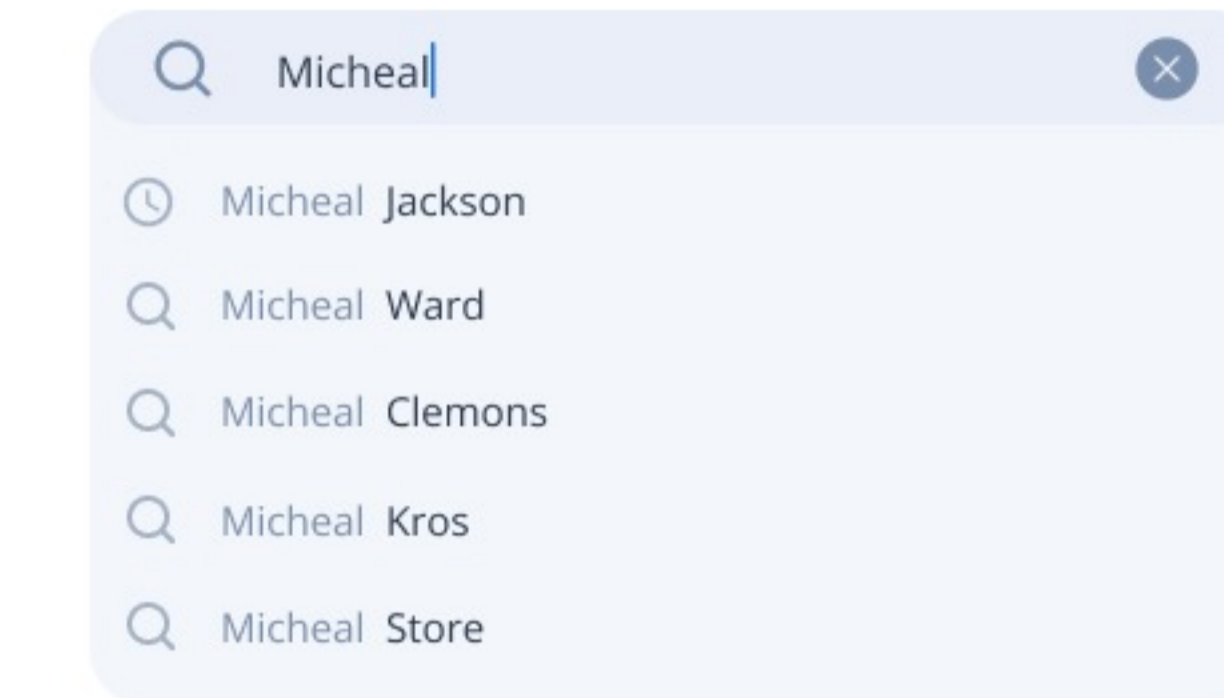


Search

Default



Active



Switches



Fonts

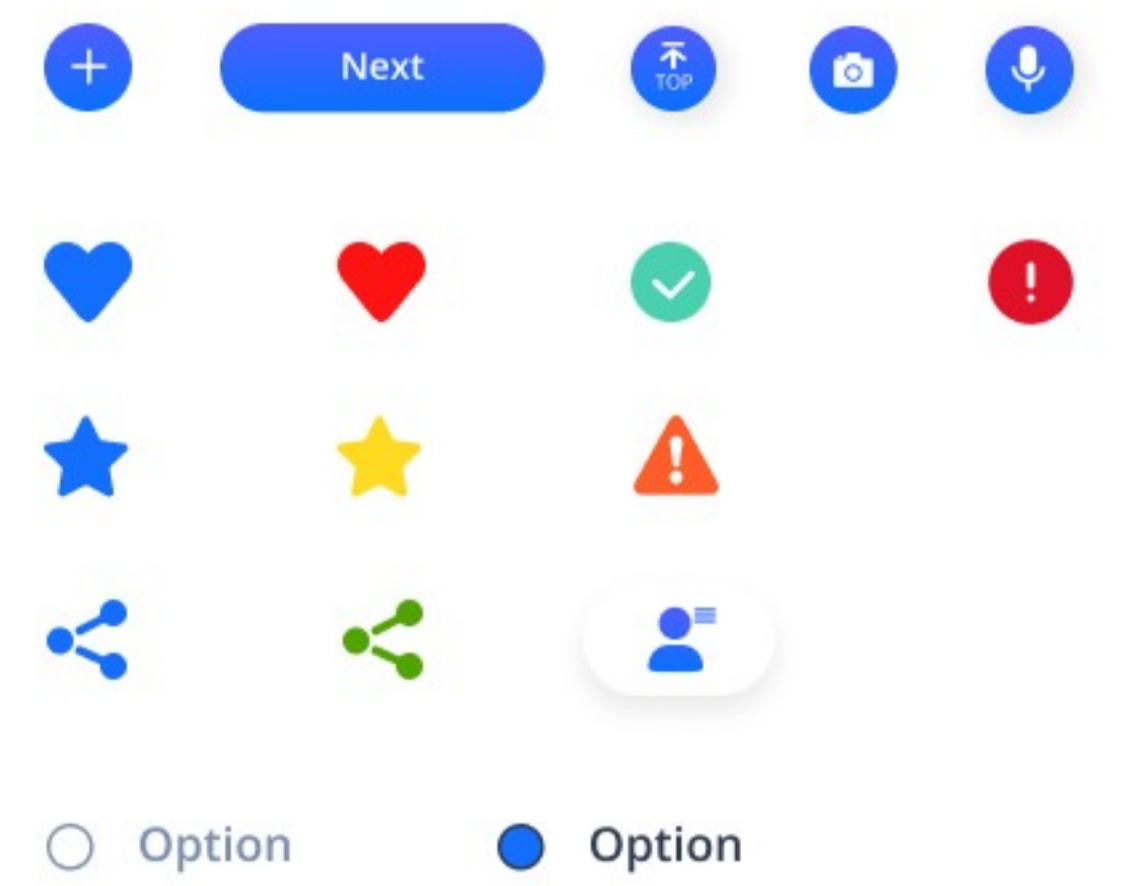
Regular SemiBold Bold

abcdefghijklmnopqrstuvwxy
 ABCDEFGHIJKLMNOPQRSTUVWXYZ

Tabs

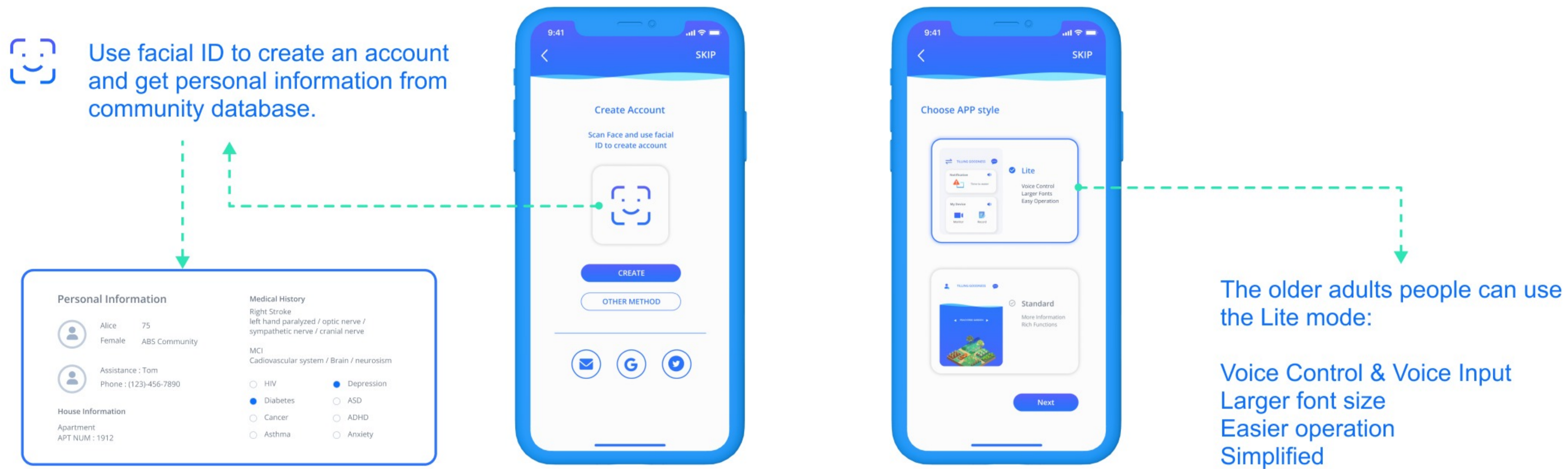


Buttons & Tags

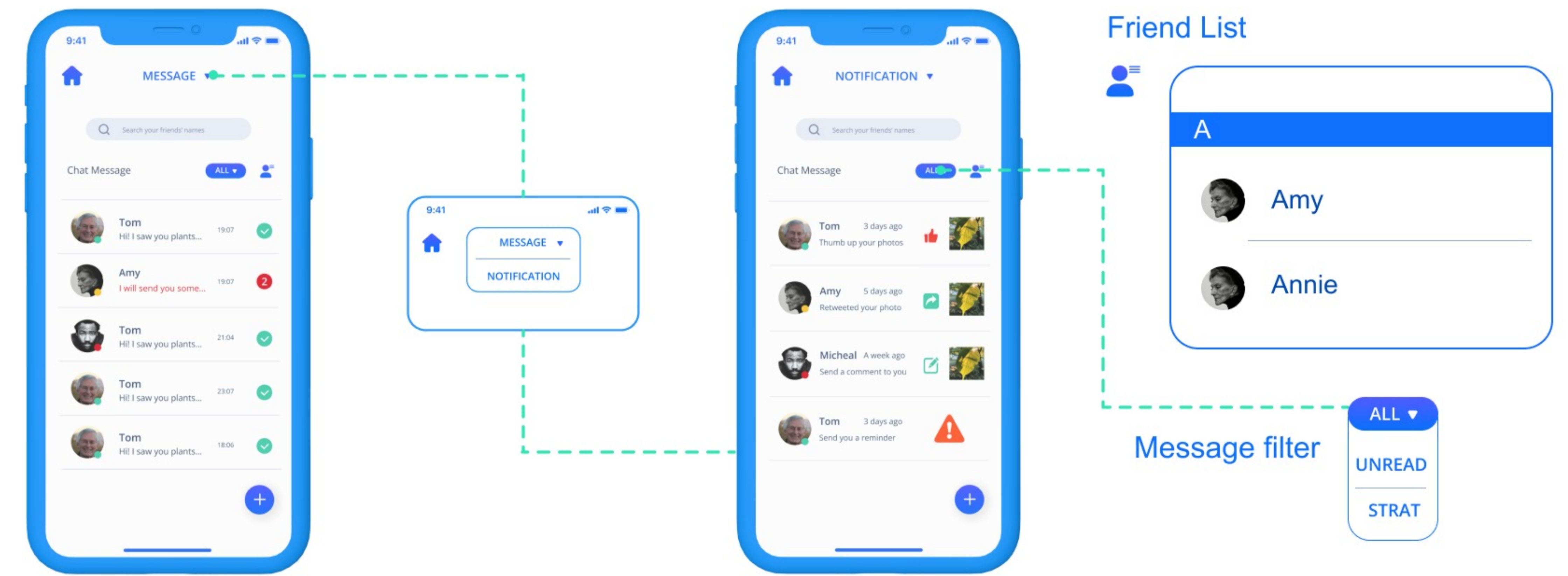


Final Design

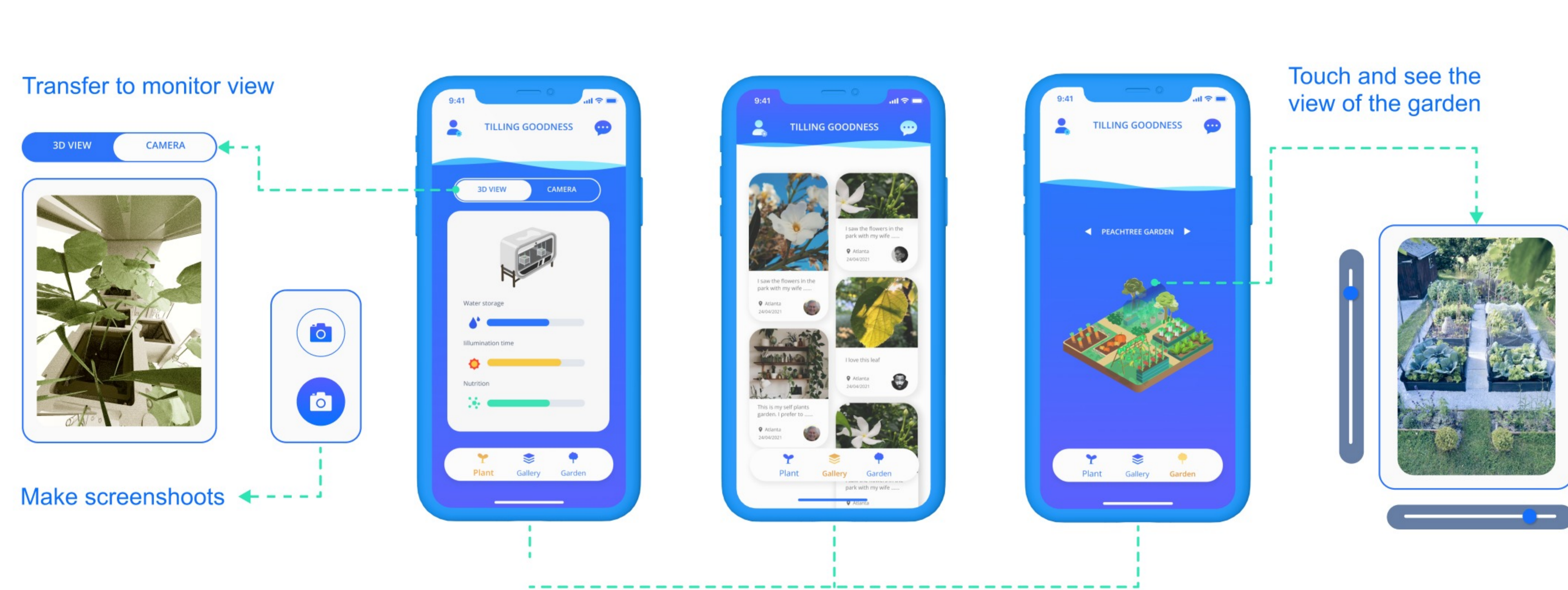
Onboarding



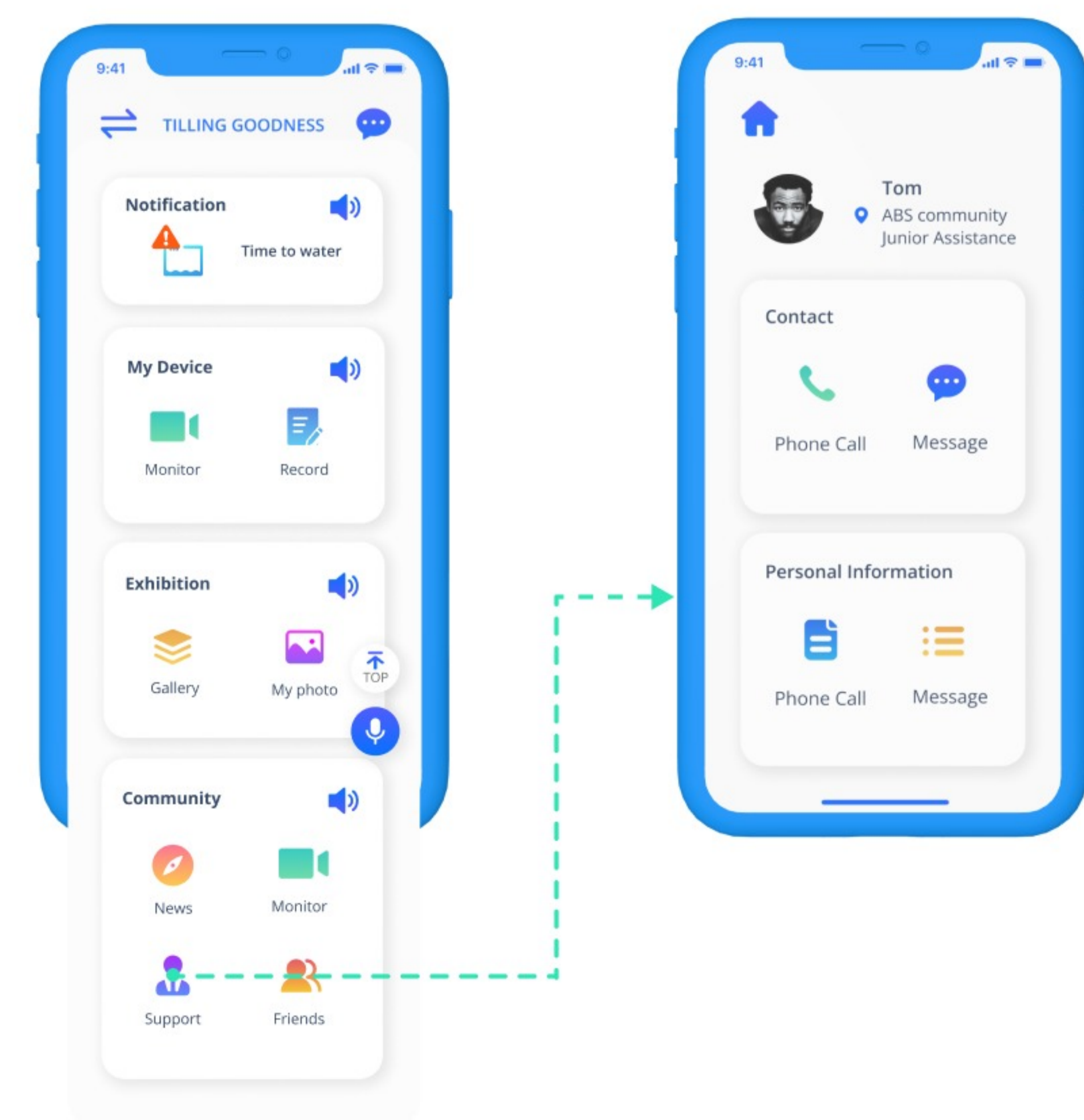
Message



Main Page



Lite Mode



Notification

