

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ
УНІВЕРСИТЕТ УКРАЇНИ “КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”**

ІІСА

Кафедра Системного проектування

Лабораторна робота 4

з дисципліни: «ПРОЕКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ»

**Розробити тестовий мобільний додаток за темою індивідуального
завдання.**

Виконав:

студент групи ДА-71

Михайловин Р. Г.

Київ – 2020

Функціонал CRUD

```
# lab4.py
import json

class Pet:
    def __init__(self, name, w=None, y=None, p=None, f=None, c=None,
history=None):

        self.name = name
        self.w = w if w else 0
        self.y = y if y else 0
        self.pfc = [p, f, c]
        self.history = history if history else []

    def analyze(self):
        pass

    def feed(self, pfc):

        self.history.append(pfc)

    def show(self):
        print('name: ', self.name, ' w: ', self.w, ' y: ', self.y, ' pfc: ',
self.pfc)

    def cl(self):

        self.history.clear()

    def tojson(self):

        return json.dumps(self, default=lambda o: o. dict )

class Food:
    def __init__(self, name, p, f, c, w=None, left=None):

        self.name = name
        self.w = w if w else 0
        self.left = left if left else self.w
        self.pfc = [p, f, c]

    def show(self):
        print(self.name, ': w:', self.w, ' left: ', self.left, ' pfc: ',
self.pfc)

    def tojson(self):

        return json.dumps(self, default=lambda o: o. dict )

def to_json(p_list, f_list):

    j_list = [p.tojson() for p in p_list]
    print(j_list[0])
    with open('pets.json', 'w') as p_file:
```

```

        json.dump(j_list, p_file)
    jf_list = [f.tojson() for f in f_list]
    print(jf_list[0])
    with open('food.json', 'w') as f_file:
        json.dump(jf_list, f_file)

def main():
    p_list = []
    with open('pets.json', 'r') as data_file:
        d = json.load(data_file)
    print(d)
    for i in d:
        s = json.loads(i)
        p_list.append(Pet(s['name'], s['w'], s['y'], s['pfc'][0], s['pfc'][1],
s['pfc'][2], s['history']))
    with open('food.json', 'r') as data_file:
        df = json.load(data_file)
    print(df)
    f_list = []
    for i in df:
        s = json.loads(i)
        f_list.append(Food(s['name'], s['pfc'][0], s['pfc'][1], s['pfc'][2],
s['w'], s['left']))
    pet = 0
    food = 0
    # p_list = [Pet('cat'), Pet('dog')]
    # meat = Food('meat', 50, 30, 20)
    # f_list = [meat]
    print('0 - choose pet')
    print('1 - pet data')
    print('2 - pet history')
    print('3 - feed pet')
    print('4 - clear pet history')
    print('5 - food data')
    print('6 - add pet')
    print('7 - delete last pet')
    print('9 - exit')
    while 1:
        a = int(input())
        if a == 0:
            pet = int(input())
            print('pet: ', pet)
        elif a == 1: # read
            p_list[pet].show()
        elif a == 2:
            print(p_list[pet].history)
        elif a == 3: # update
            p_list[pet].feed(f_list[food].pfc)
        elif a == 4: # delete
            p_list[pet].cl()
        elif a == 5: # read
            for food in f_list:
                food.show()
        elif a == 6: # create
            name = input('input pet name')
            p_list.append(Pet(name))
        elif a == 7: # delete
            a = p_list.pop()

```

```

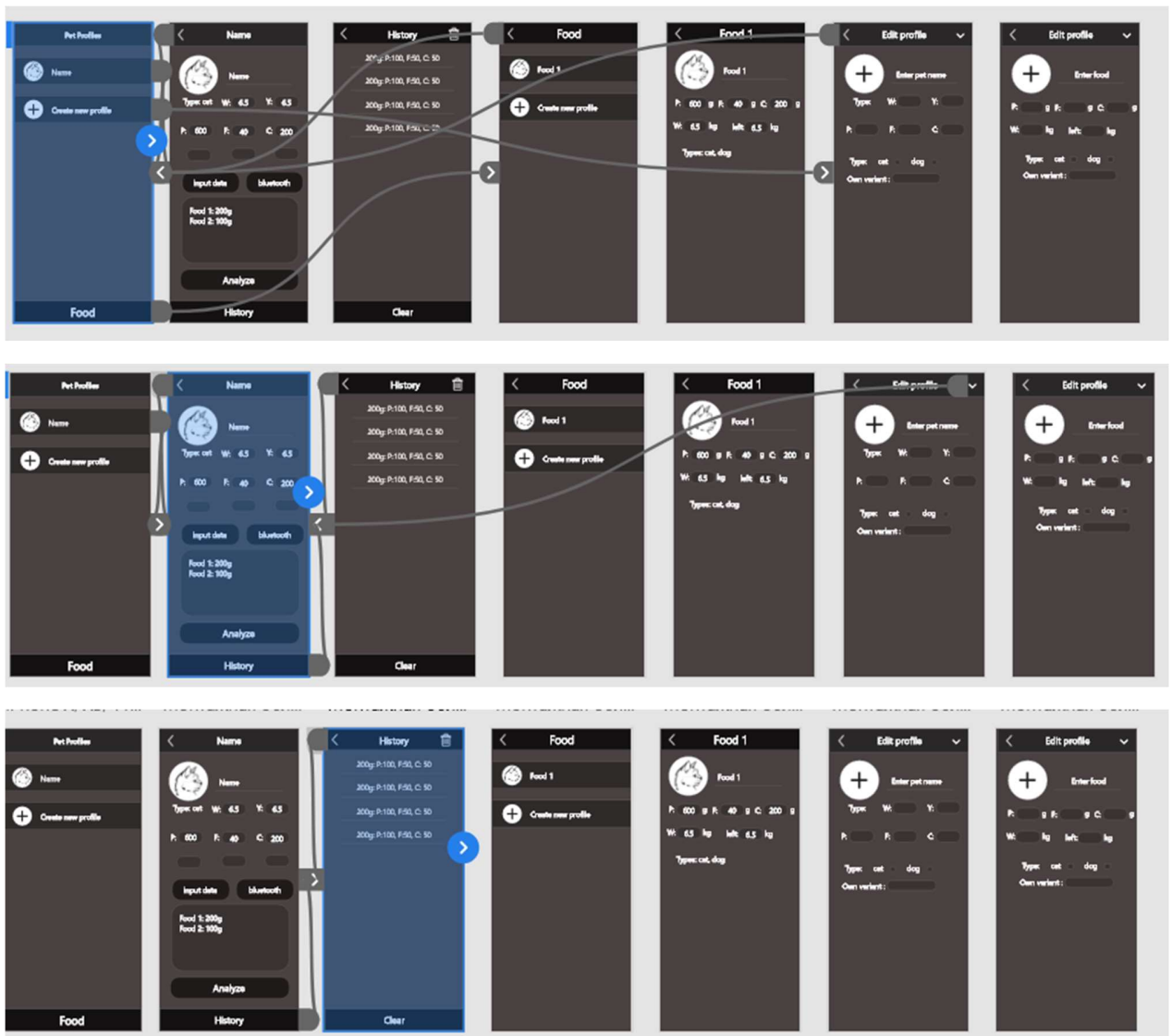
if len(p_list) == pet:
    pet -= 1
elif a == 9:
    to json(p_list, f_list)
    exit()

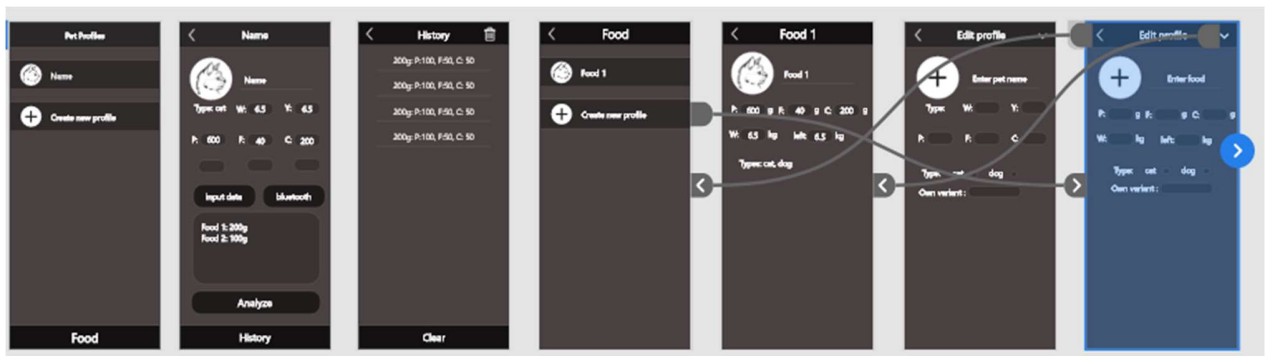
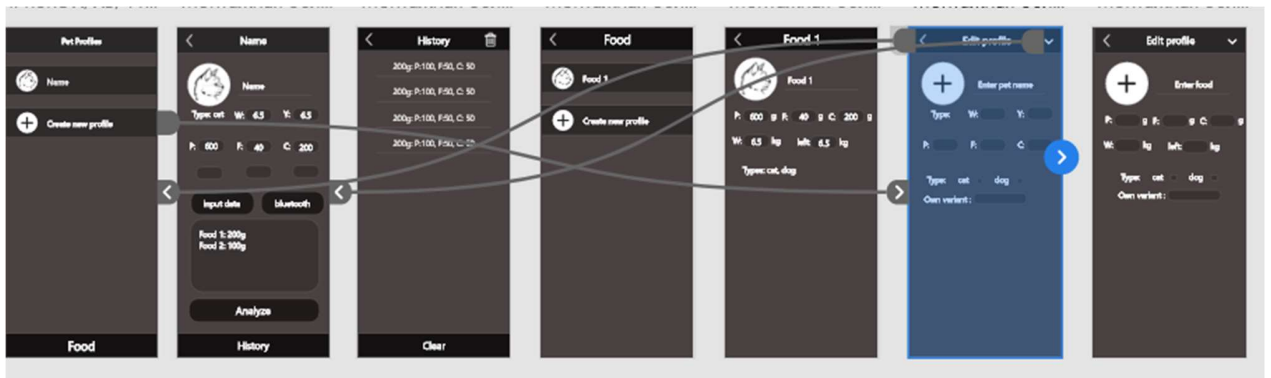
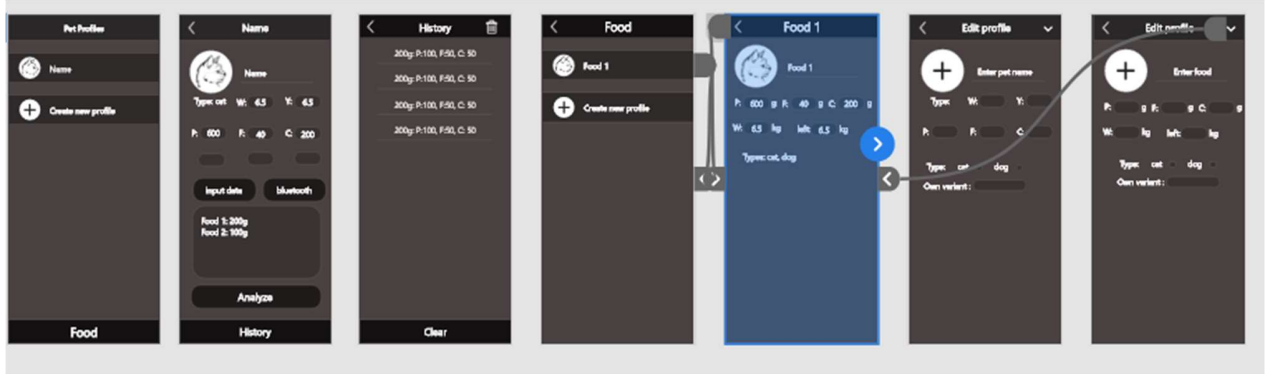
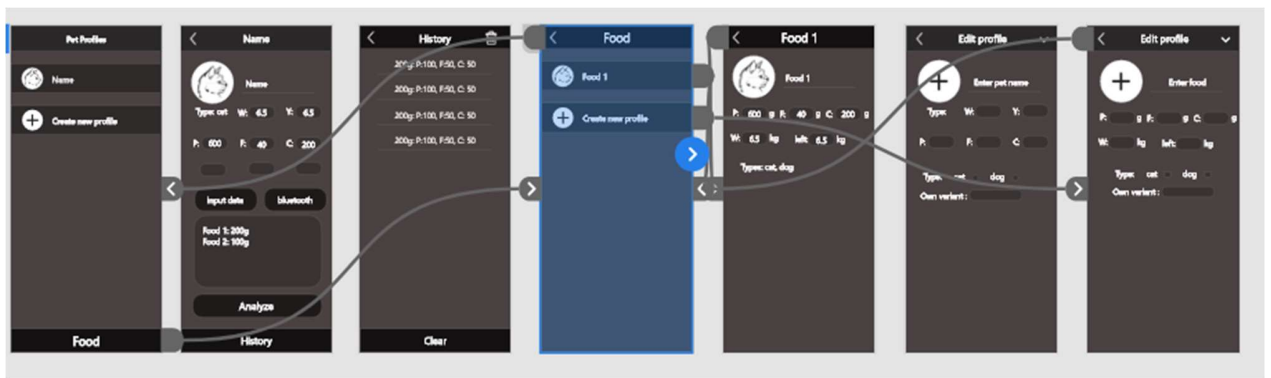
if __name__ == "__main__":
    main()

```

Реалізовано на мові програмування Python, використовувалася бібліотека json.

Wireframes





Висновок:

На лабораторній роботі було реалізовано функціонал CRUD