Python based implementation of Aqua Monitoring System using Raspberry Pi

Mohan Allam (mohanallam@gmail.com)
Dr. K. Padma Vasavi (subbusouri@gmail.com)

VISHNU
UNIVERSAL LEARNING

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN
Bhimavaram, A.P

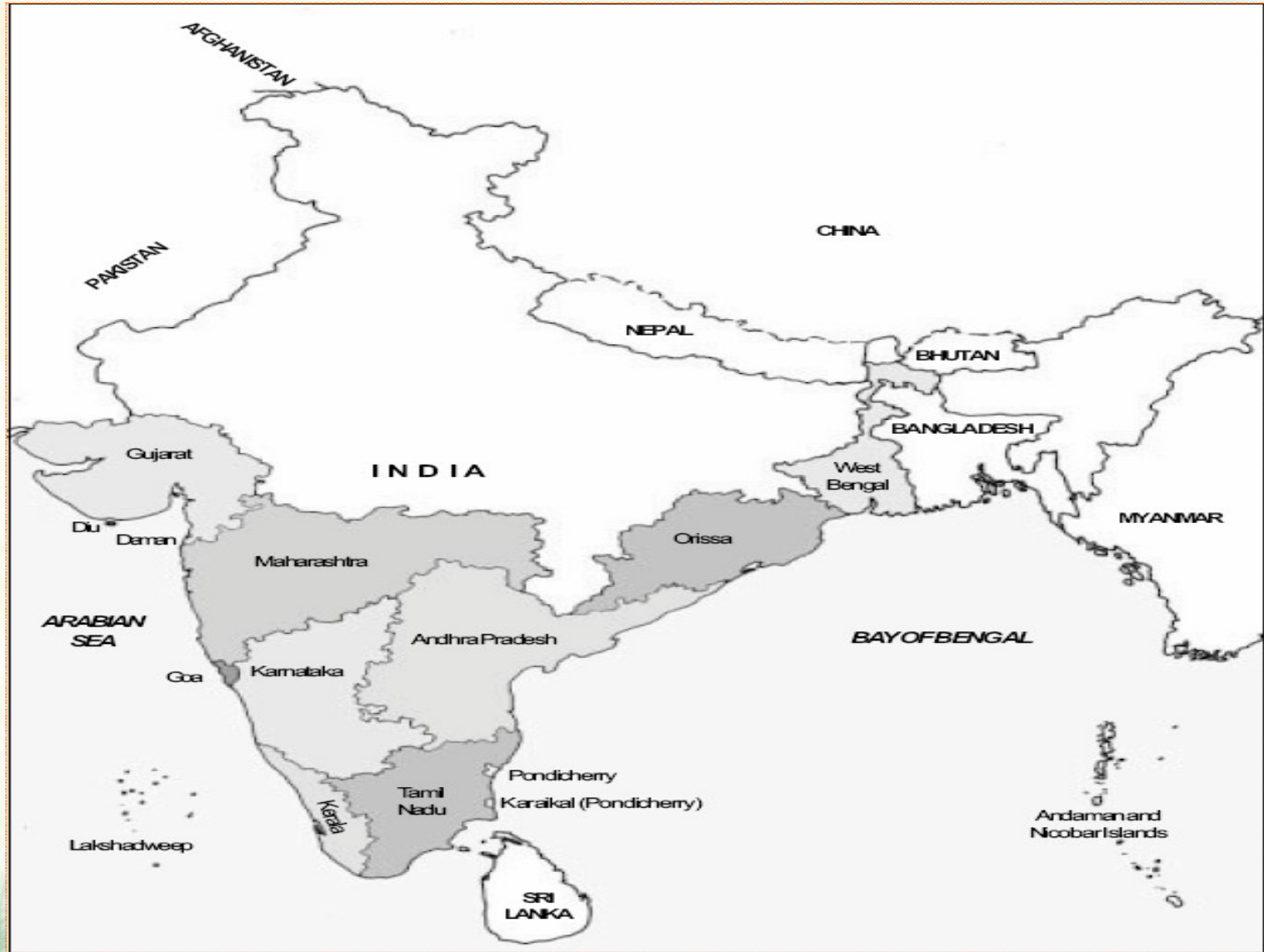
OUTLINE

- Introduction
- Need of Water Quality Monitoring in Aqua Culture
- Implementation
- Mobile Application
- Conclusion
- References

INTRODUCTION

Shrimps……

Shrimp Culture in India



NEED OF
WATER QUALITY MONITORING IN SHRIMP CULTURE

Need of Water Quality in Shrimp Ponds

- Water Quality (WQ) determines the ultimate ***Success*** or ***Failure*** of an aqua culture.
- The farmer must measure, record and manage WQ all through the growing season.
- Water Quality parameters affect respiration, feeding, metabolism and reproduction of shrimps.

Water Quality Parameters

- pH
- Temperature
- Dissolved Oxygen
- Hardness
- Ammonia, Nitrite & Nitrate

Water Quality Parameters

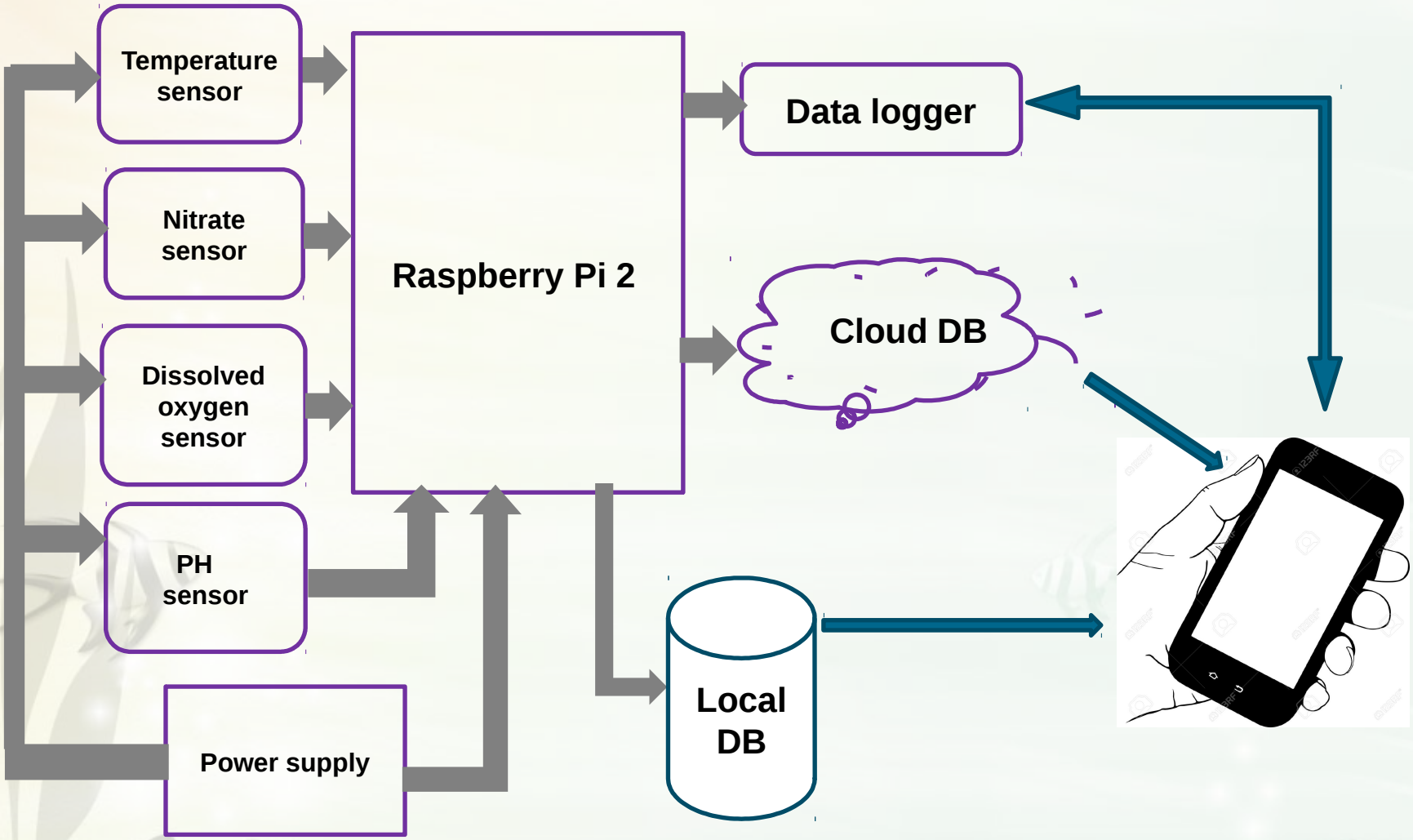
Water Parameter	Optimum evel
Temperature	26-33C
Salinity	10-25 ppm
Dissolved oxygen	>3ppm
PH	7.5-8.5
Total Ammonia Nitrogen	<1.0ppm
Total Nitrate Nitrogen	<5.0ppm
Nitrite Nitrogen	<0.01ppm
Sulphide	<0.03ppm
Biological Oxygen Demand	<10ppm
Chemical Oxygen Demand	<70ppm
	(parts per million)

In General…

- Formers take water samples from pond.
- Get results from Aqua Labs.
- Expensive.
- Difficult Process.
- Take More Time.

IMPLEMENTATION

ARCHITECTURE



SENSOR SUITE FOR WATER QUALITY SCREENING

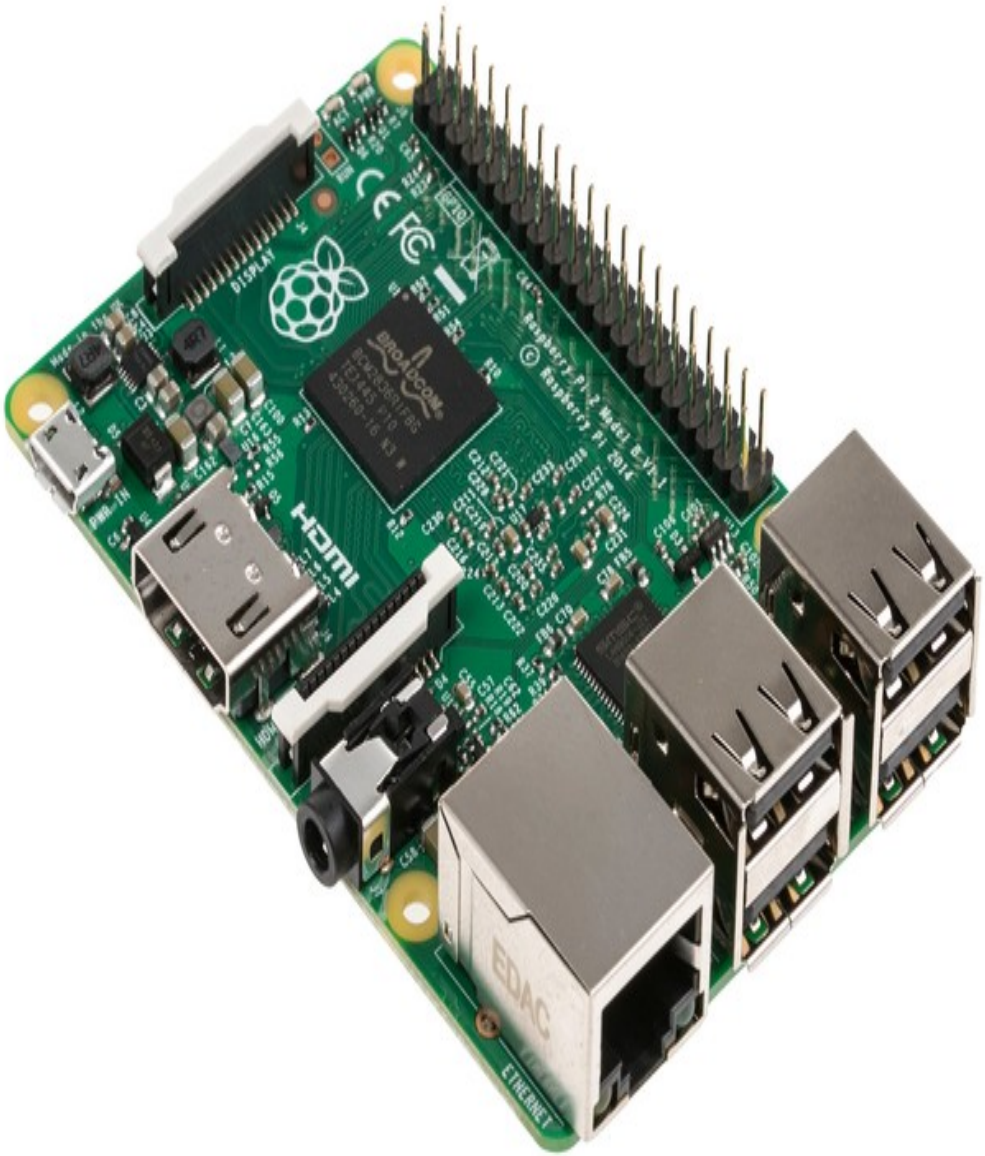
DISSOLVED OXYGEN SENSOR

NITRATE SENSOR

PH SENSOR

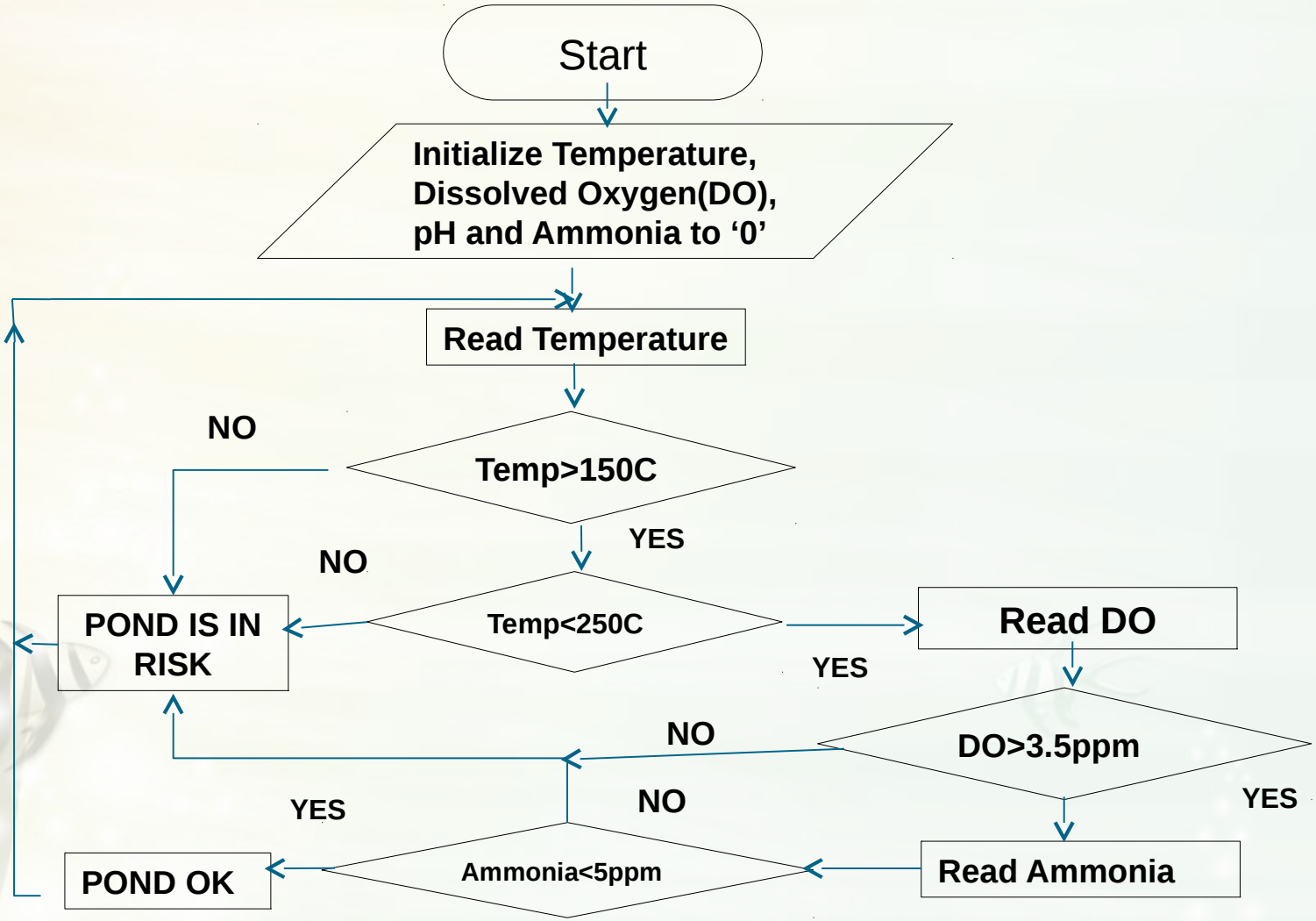
Temperature sensor:

Raspberry Pi 2



1. Processor
2. GPIO Pins
3. SD Card Slot
4. Ethernet Port
5. USB Port
6. HDMI Port
7. 3.5mm Jack

FLOW CHART


Start
Initialize Temperature, Dissolved Oxygen(DO), pH and Ammonia to '0'
Read Temperature
Temp>150C
NO
YES
Temp<250C
NO
POND IS IN RISK
Read DO
YES
DO>3.5ppm
NO
YES
Read Ammonia
Ammonia<5ppm
NO
YES
POND OK

Phases of Implementation

Phase – I :

- Acquire and store data from sensors .
- Development of mobile app.

Phase – II:

- Capturing images of shrimps.
- Make images available to User.

Phase – III:

- Identification of diseased shrimps from images.
- Alert the User by sending message.

PHASE – I

- Acquire and store data from sensors .

- Development of mobile app.

Raspberry Pi to Mobile App. Implementation

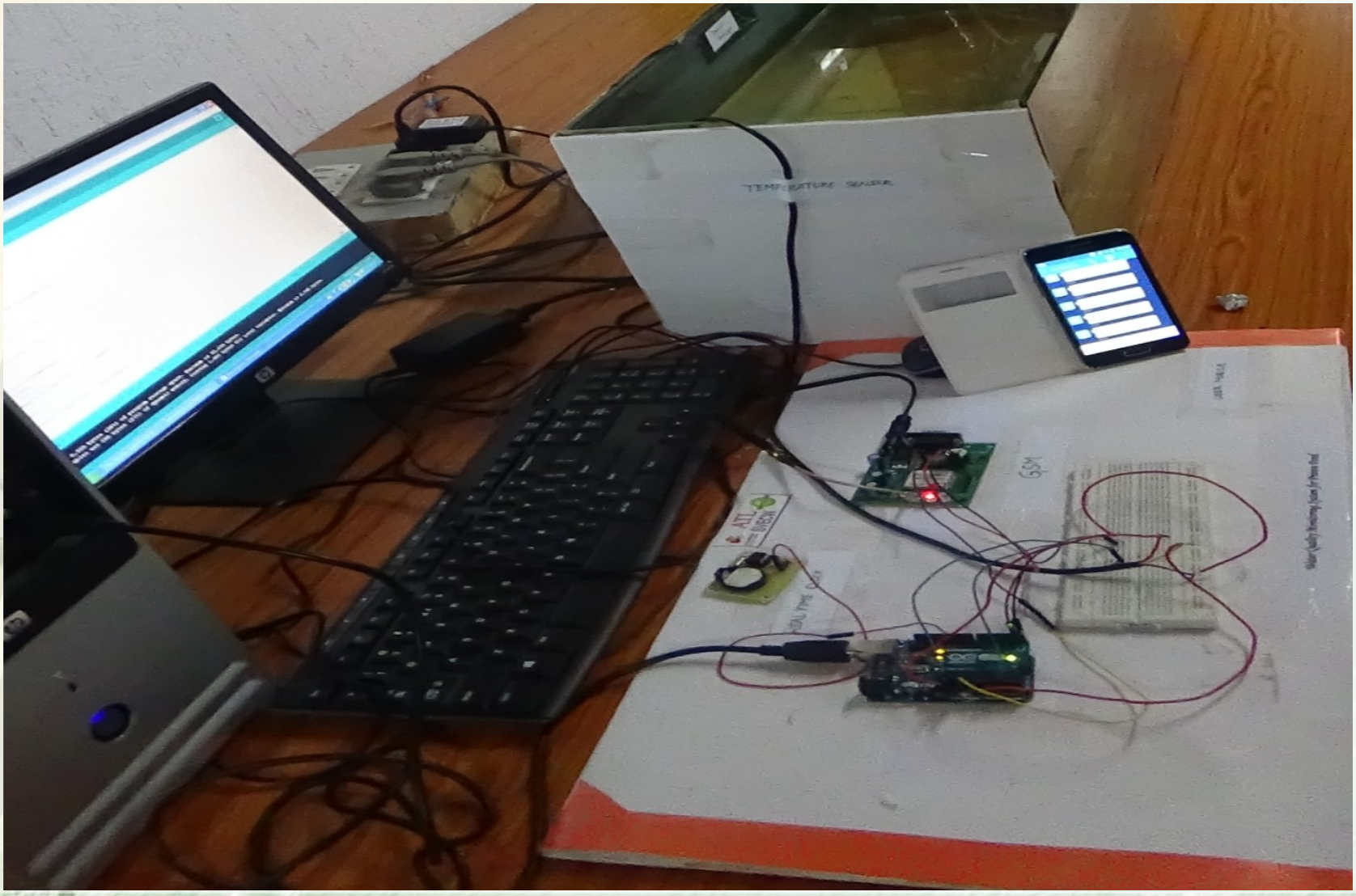
DISSOLVED OXYGEN SENSOR

NITRATE SENSOR

PH SENSOR

Temperature sensor:

Sensor Suite Set-up



Python Coding for …

1. Accessing data from sensors (GPIO)
2. Socket Programming

hello123.py

File Edit Search Options Help

```
import sys
import socket
import select
import os
import glob
import time as tm
import RPi.GPIO as gp
import RPi.GPIO as rp
import RPi.GPIO as GPIO # always needed with RPi.GPIO
from time import sleep  # pull in the sleep function from time module
pause_time = 0.02          # you can change this to slow down/speed up
HOST = ''   # Symbolic name meaning the local host
PORT = 7728        # Arbitrary non-privileged port

gp.setmode(gp.BOARD)
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)
GPIO.setup(18, GPIO.OUT)# set GPIO 25 as output for white led
GPIO.setup(24, GPIO.IN,GPIO.PUD_UP)# set GPIO 25 as Input for temperature sensor

#getting sensors into system

os.system('modprobe w1-gpio')
os.system('modprobe w1-therm')

base_dir='/sys/bus/w1/devices/'
device_folder=glob.glob(base_dir+'*28')[0]
device_file=device_folder+'/w1_slave'
```

hello123.py

File Edit Search Options Help

```
device_folder=glob.glob(base_dir+'28*')[0]
device_file=device_folder+'/w1_slave'
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM) #socket opening
print 'Socket created'
try:
    s.bind((HOST,PORT)) #socket binding
except socket.error , msg:
    print 'Bind failed. Error code: ' + str(msg[0]) + 'Error message: ' + msg[1]
    sys.exit()
print 'Socket bind complete'
s.listen(1) #socket listening
print 'Socket now listening'

def read_temp_raw():
  f=open(device_file,'r')
  lines=f.readlines()
  f.close()
  return lines

while 1:
    conn, addr = s.accept() # socket accepting
    print 'Connected with ' + addr[0] + ':' + str(addr[1])
    data = conn.recv(5)
    reply = 'OK...' + data
    read_temp()

    conn.close()

def read_temp():
  lines=read_temp_raw()
  while lines[0].strip()[-3:]!='YES':
    time.sleep(0.2)
    lines=read_temp_raw()
  equals_pos=lines[1].find('t=')
  if equals_pos!=-1:
    temp_string=lines[1#[equals.pos+2:]
    temp_c = float(temp_string)/1000.0
    temp_f=temp_c*9.0/5.0+32.0
    conn.send(temp_c,temp_f)
s.close()
GPIO.cleanup()
```

Raspberry Pi - Cloud Python Coding….

DISSOLVED OXYGEN SENSOR

NITRATE SENSOR

PH SENSOR

Temperature sensor:

Raspberry Pi, Cloud (Parse) Logging Library

- Get the ParsePy library
- Install on your pi using the command

```
sudo python setup.py install
```

Temperature Logging sample Python code

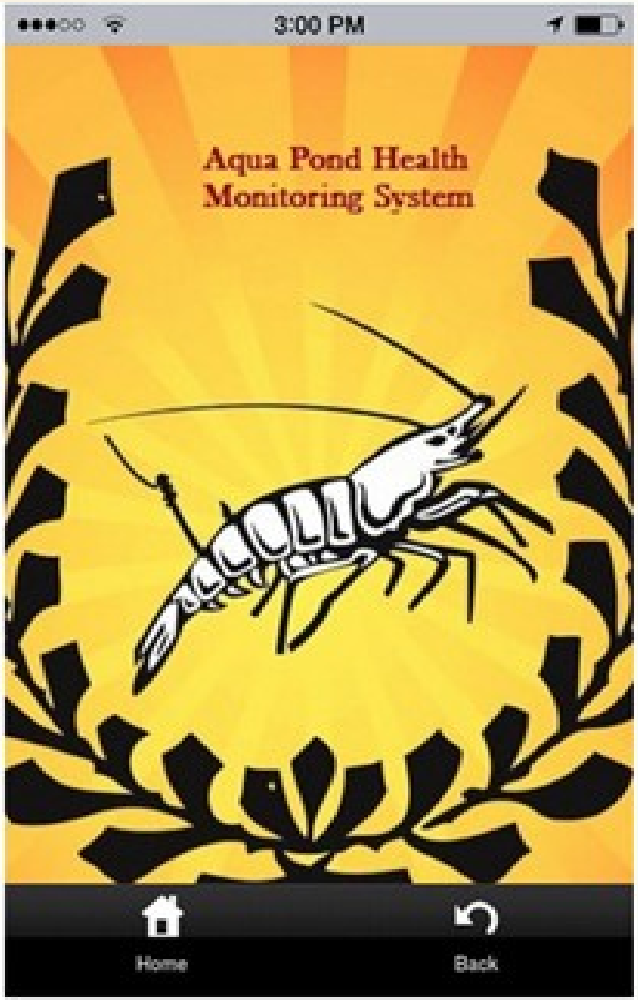
```
def getTempForFile(self,file):
 try:
  f = open(self.tempDir + file + "/w1_slave", 'r')
 except IOError as e:
  print "Error: File " + self.tempDir + file + "/w1_slave" + " doesn't exist"
  return;
 lines=f.readlines()
 crcLine=lines[0]
 tempLine=lines[1]
 result_list = tempLine.split("=")
 temp = float(result_list[-1])/1000 # temp in Celcius
 temp = temp + self.correctionFactor # correction factor
 #if you want to convert to Celcius, comment this line
 temp = (9.0/5.0)*temp + 32
 if crcLine.find("NO") > -1:
  temp = -999
 if(self.debug):
  print "Current: " + str(temp) + " " + str(file)
 return float(int(temp*100))/100
```

Sample Data Logging in Cloud

objectId String	Probe String	Temperature Number
LJeVPtD7ef	28-00000466742d	72.95
CbY1i6gA8b	28-00000466742d	73.06
oOJ9MhUz9t	28-00000466742d	72.95
fyuGjO5Uea	28-00000466742d	73.06
GcqoWoGqqE	28-00000466742d	72.95
kK8fq79x1G	28-0000053559c2	72.95
llwILv5tQw	28-00000466742d	73.06
U50FTwNFKe	28-0000053559c2	73.06
2cRaILJpF3	28-0000053559c2	72.95
R5LO0fkaKD	28-0000053559c2	73.06
tbqCk8Mbj2	28-00000466742d	72.95
K5JxYHZRRu	28-0000053559c2	73.73
oEdWAlkN20	28-0000053559c2	73.84
u3OJEjDytA	28-0000053559c2	73.73
NfuaZo3kiY	28-00000466742d	73.62
rK4gsHxJ32	28-0000053559c2	73.84
18gAlVXnKL	28-0000053559c2	73.73
Cmhe4noYrX	28-00000466742d	73.73
kTK0azGj7J	28-0000053559c2	73.84

Aqua Pond Health Monitoring System (Mobile Application)

Welcome Screen

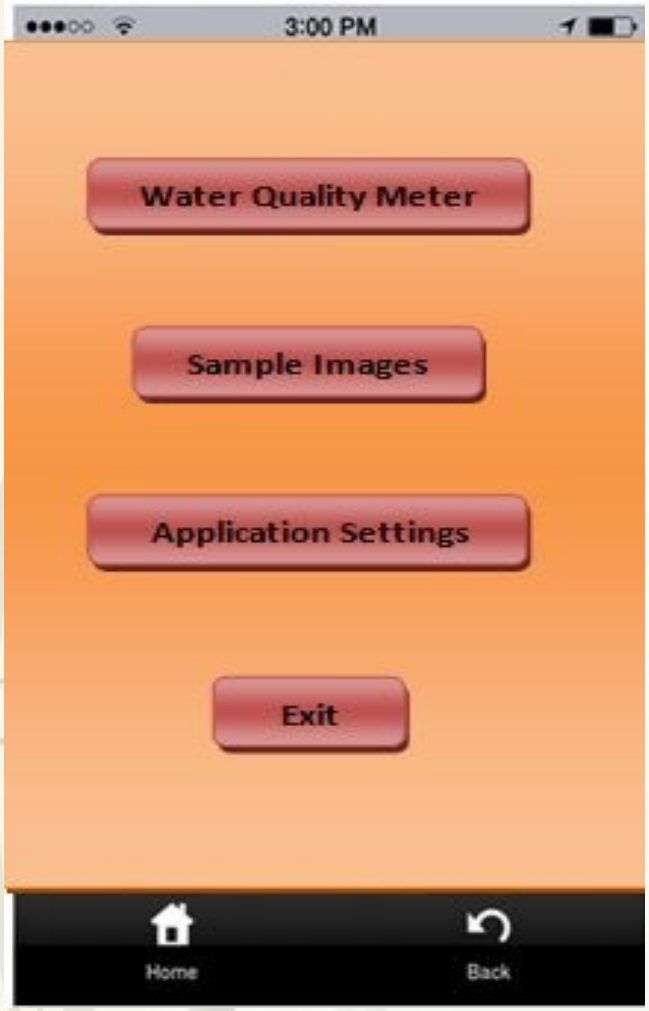


Start Screen

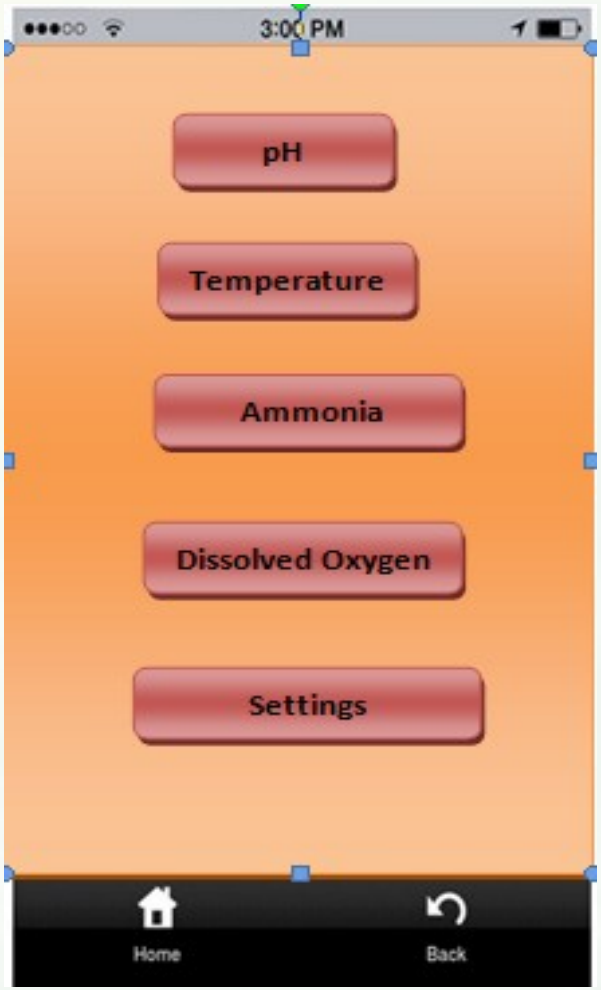
Registration Screen

Login Screen

Home Screen

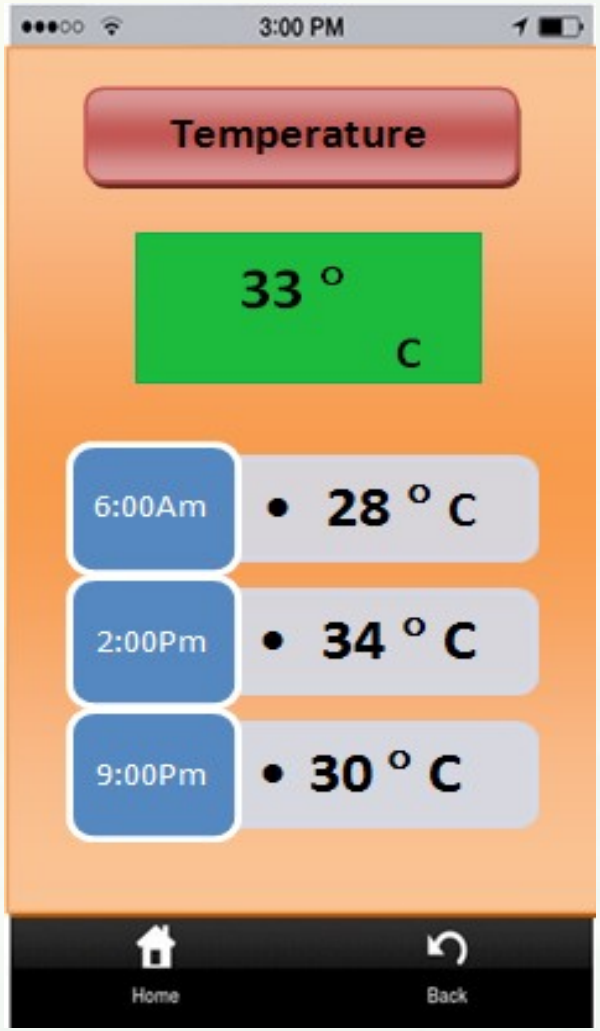


Water Quality



pH

Temperature



PHASE – II

- Capturing images of shrimps.

- Make images available to User.

Python Programming : Capture an Image

Start by installing the Python `picamera` and GPIO library packages:

```
sudo apt-get install python-picamera python3-picamera python-rpi.gpio
```

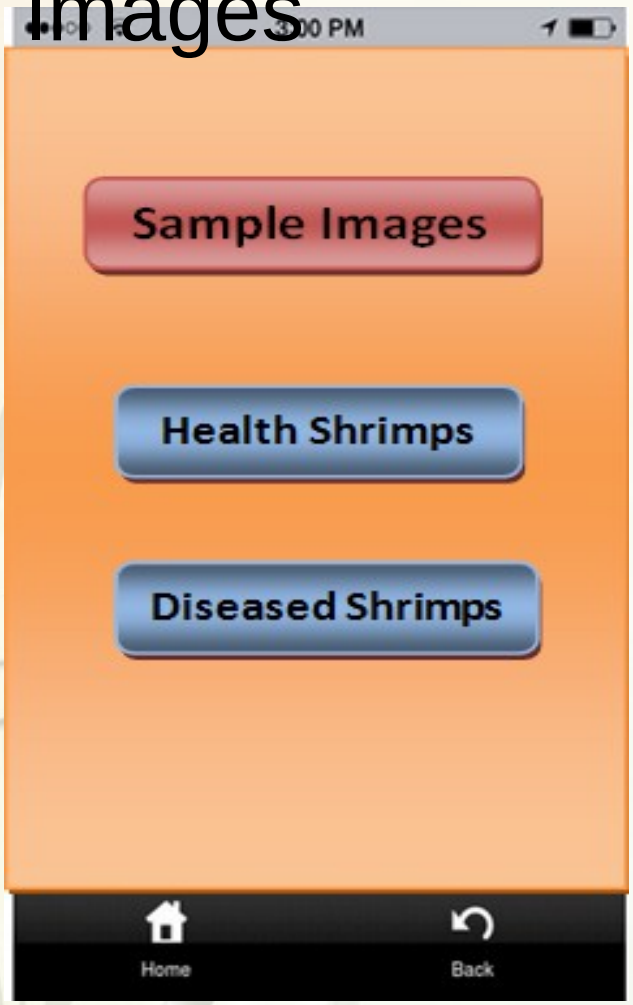
1. At the command prompt enter `startx` to start the graphical desktop environment
2. Double click on `LXTerminal` to start a command line, and enter `sudo idle &` to start the Python environment
3. Select `File > New Window` from the menu to start a text editor
4. Enter the following code (case is important!):

```
import time
import picamera

with picamera.PiCamera() as camera:
    camera.start_preview()
    time.sleep(5)
    camera.capture('/home/pi/Desktop/image.jpg')
    camera.stop_preview()
```

5. Select `File > Save` from the menu and give your script a name, e.g. `workshop.py`
6. Select `Run > Run Module` from the menu (or just press `F5`) to run the script

Sample Images

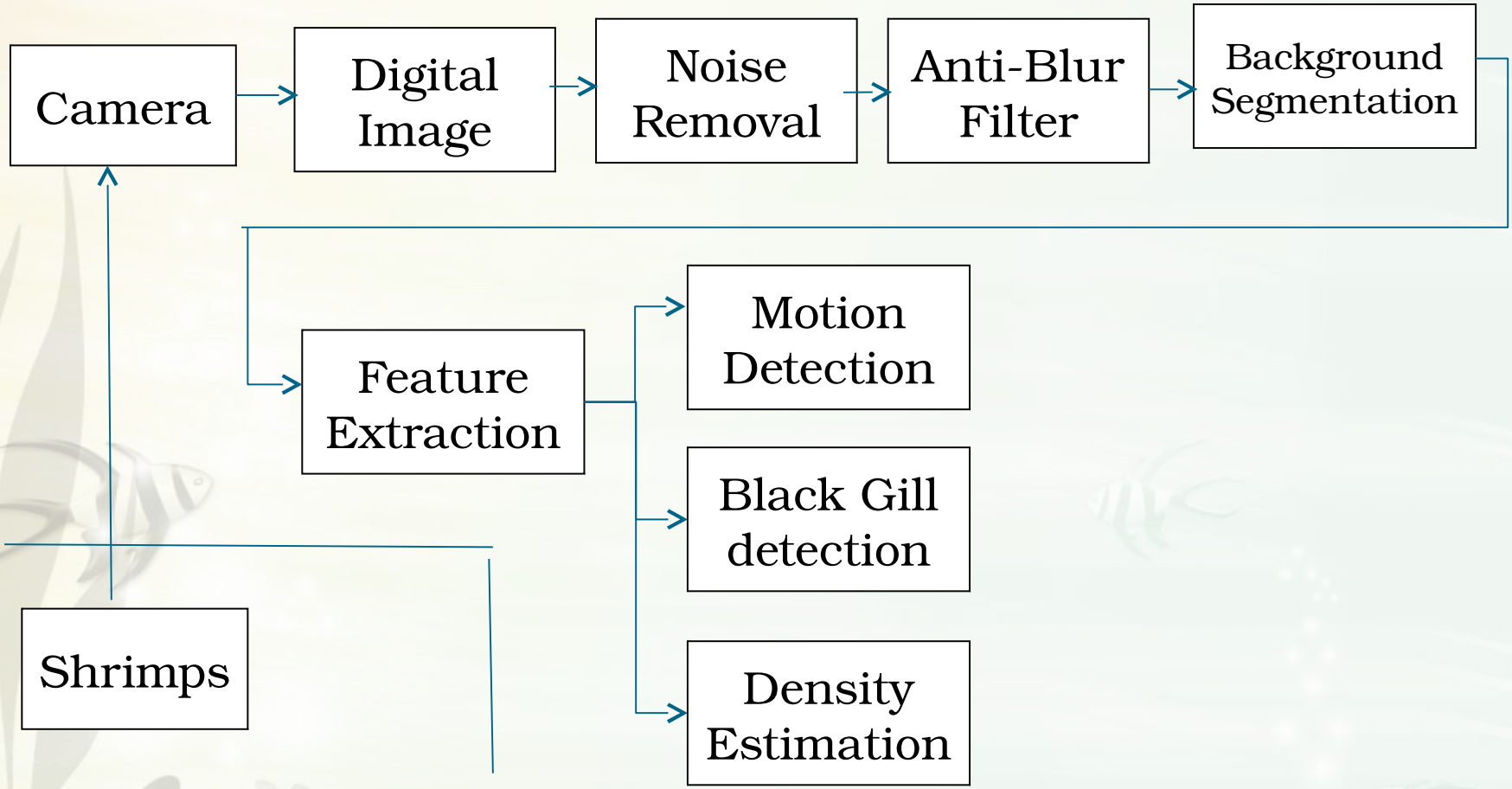


Healthy Shrimps

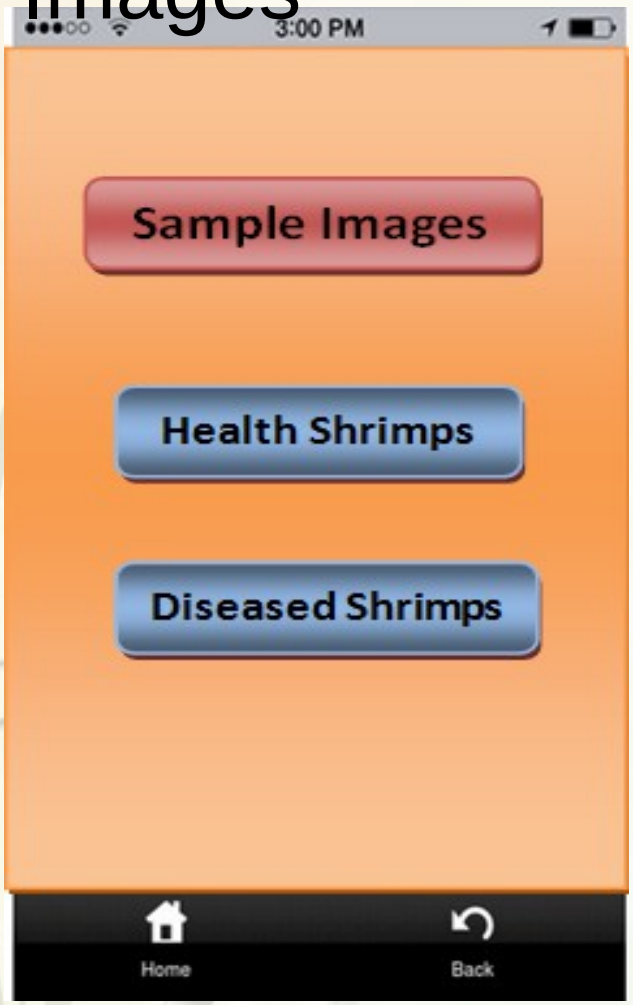
PHASE – III

- Identification of diseased shrimps from images.
- Alert the User by sending message.

Explanatory Schematic or Image



Sample Images



Diseased Shrimps

Conclusion

- A Sensor suite for monitoring the water quality of aqua pond is discussed
- It would aid the aquaculture farmers to retain their profits in growing shrimps owing to good water quality

References

[1] M. C. M. Beveridge, "Cage, Aquaculture", Fishing News Book LTD., Farnham, Surrey, England. pp. 352, 198

[2] B. H. Buck, G. Krause, T. Michler, A. Berg-Pollack, M. Brenner, C. M. Buchholz, J. A. Busch, R. Fisch,M. Geisen, A. Haasbach, A. Koch, S. Kodeih, T. Manefeld, S. Meyay, S. Saphic, D. Voss, and O. Zielinksi, "Meeting (he quest for spatial efficiency: Progress and/Prospects of Extensive Aquaculture within Offshore Wind Farms", GAIA, 2007

[3] O. Zielinski, B. Cembella, and R. Heuermann, "Bio-optical sensors onboard autonomous profiling floats", Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering - OMAE, Hamburg, pp. 1-6, 200


THANK
YOU