

Client App

Cloud Server

Base Station /
Pi

Nodes

Sensors / Devices

Phase 1

View raw
JSON text in
web browserSimple Access
PointBreadboard
Basic logging
to textBreadboard
Arduino
Basic interface
code

Feb

Phase 2

View text
in table of
valuesRead sensor readings
in database
push data to
Clientpush data
to base
stationArduino
Create module
with common
interface

March

Phase 3

Add buttons
to control/config
nodes
View network
statusSpin up server
+ domain
Install software
Add login functionBe able to
control nodes
+ configure
Keep track of
connected nodesDesign PCB
Accept control
+ configs
from base stationDesign PCB
modules

April

Phase 4

Add graphs
make prettyAllow user to
register base station
bridge base station
and client subnetsAdd option to
move records to
usb drive
Accept connections
from server
Add "sync" to add
nodeGet PCB
working
"sync" to
base stationGet PCB
working

Summer / 1866

Phase 5

Bug fixes
Code cleanupBug fixes
Code cleanupBug fixes
Code cleanupBug fixes
Code cleanupBug fixes
Add more
sensors / functionality

Phase 6

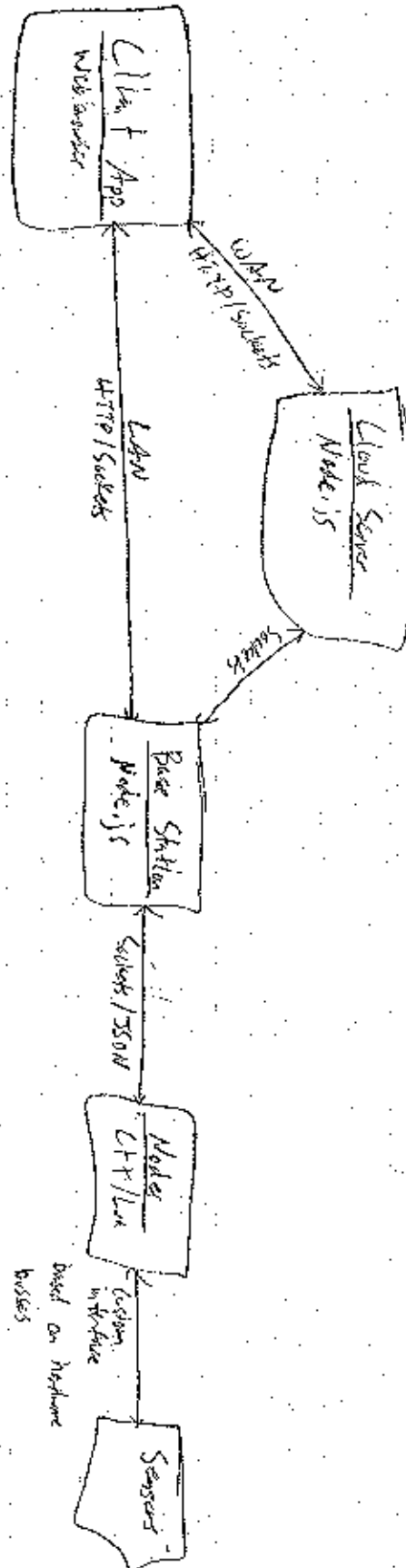
Planning
Design RequirementsPlanning
Design RequirementsPlanning
Design RequirementsPlanning
PCE'sPlanning
Design Requirements

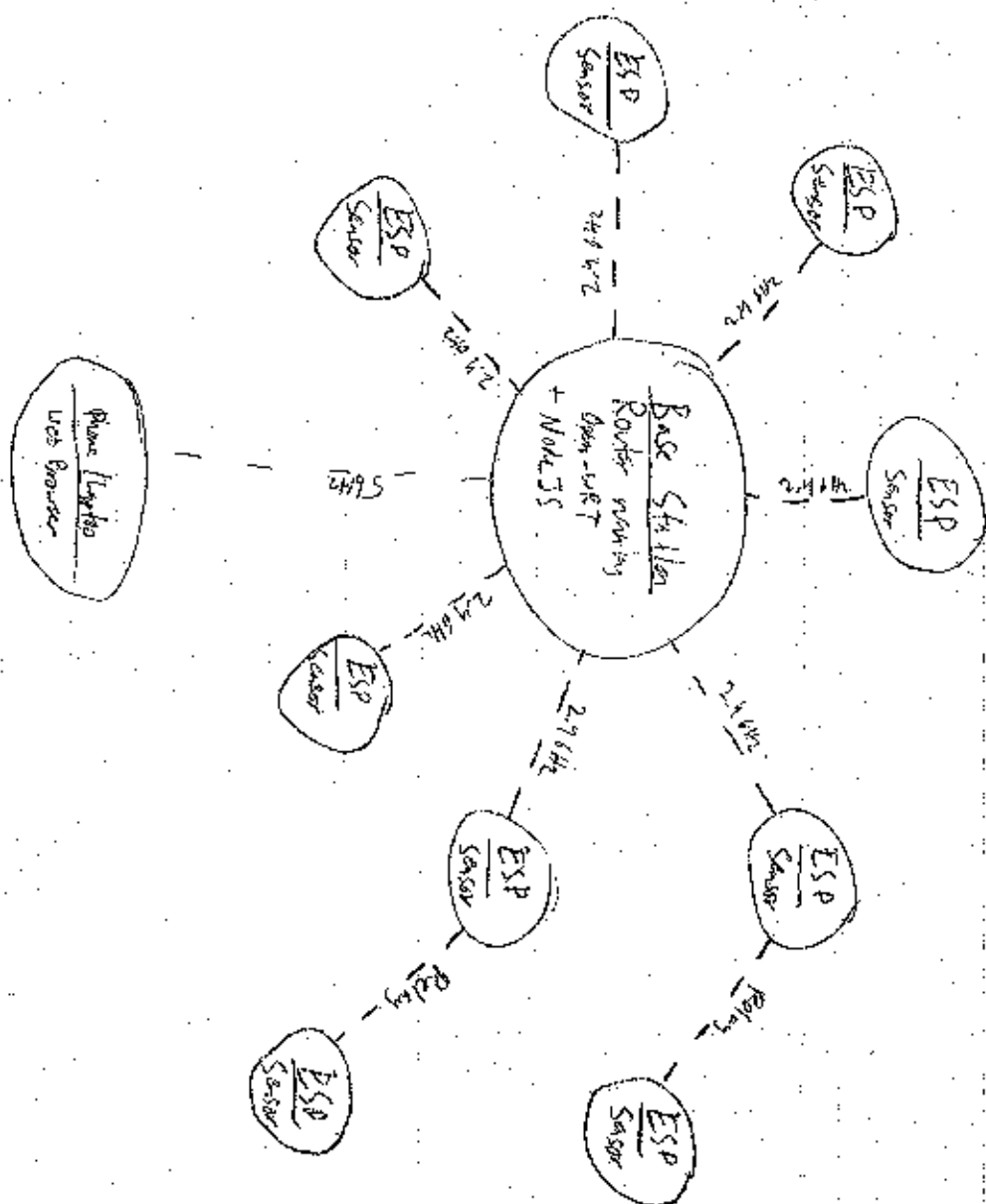
January

Phase 3.5

Clean up for
my presentation
or reports

May





PCB - Modular!

- Input: 5V, USB Power
 - ICSP Header for Flashing Firmware
 - 3 ports for sensors
 - 5V
 - GND
 - I²C SDA
 - I²C SCL
 - One wire
- All sensors have same pins, ESP figures out what the sensor is
- All boards are running the same Firmware
 - Communicate with other nodes via JSON strings through websockets
 - Try to talk to base station first, relay through other nodes
 - Keep it simple

Types of "Sensor" modules

- T Temperature
- T Humidity
- D Brightness
- T Sound
- T Motion
- Camera
- P Rain
- D Wind

Types of Controller modules

- T Thermostat
- N Relay / Outlet
- N LED
- T 8x8 LED graph (50% color LEDs, can color rows differently)
- N 8 RGB LED bar visualizer

Can't fly

Switch to connect to power / base station to download SSID / Pass list

Use Router or Raspberry Pi for data processing / storage

- keep database on USB (maybe temp on local)
- use either Postgres or SQLite
- use Node.js to connect to nodes and push readings to database
- use Node.js to push data to UI through Socket.io

Web app served through router

- later add. client server to connect to
- list / configure nodes
 - add friendly name
 - set sampling rate
 - link nodes (thermometer \rightarrow thermostat), add triggers
- view graphs of sensor data
 - recent plot
 - daily averages / accumulated values
- control things
 - turn on lights / heating / outlets
 - configure nodes (thermostat set points)