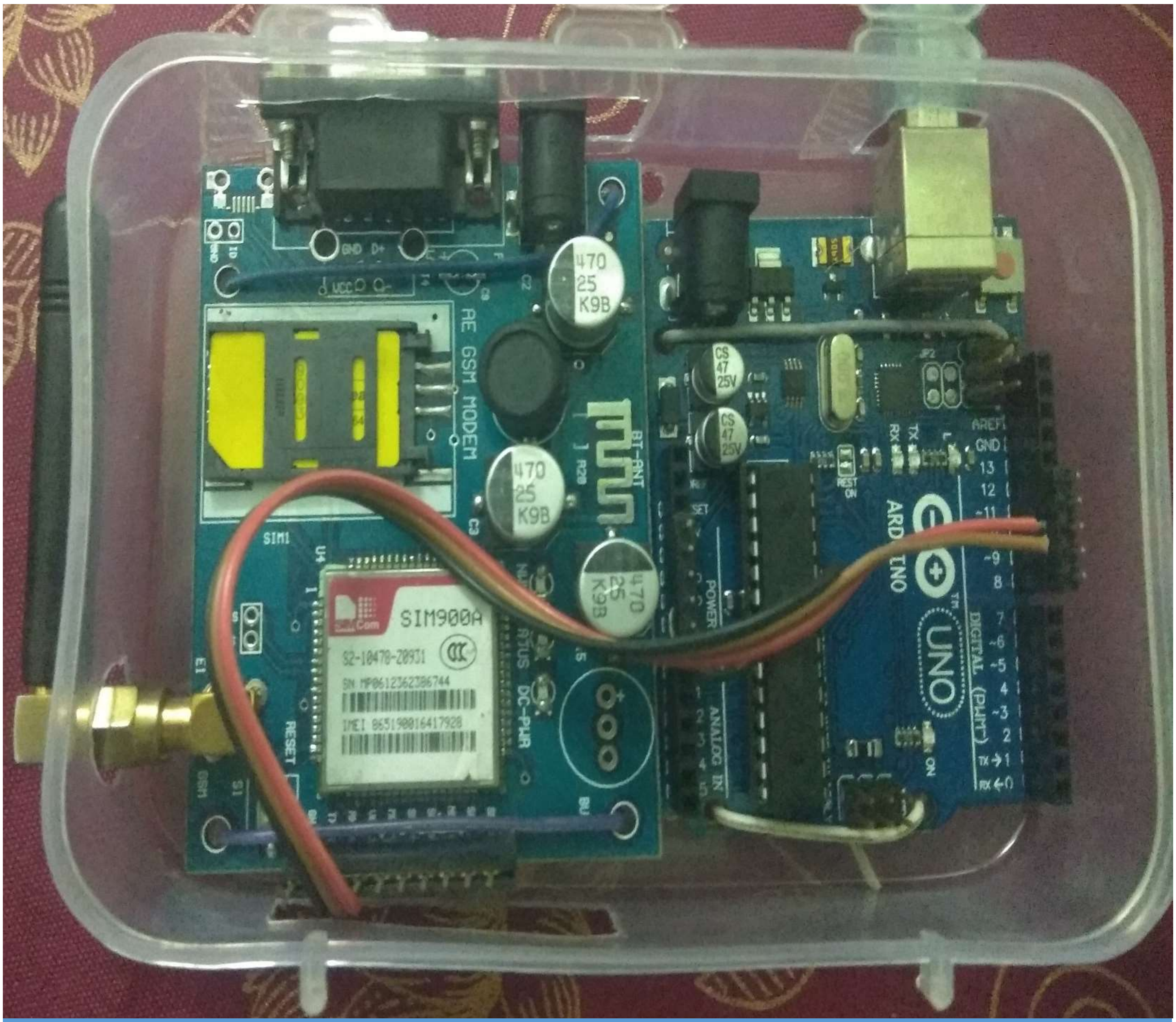


SMS Gateway

API Reference (User)

Project Work 4/27/18 Communication Engineering and Coding Theory

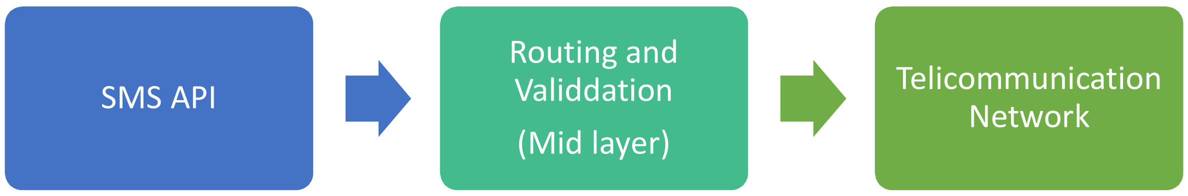
API Reference

Index

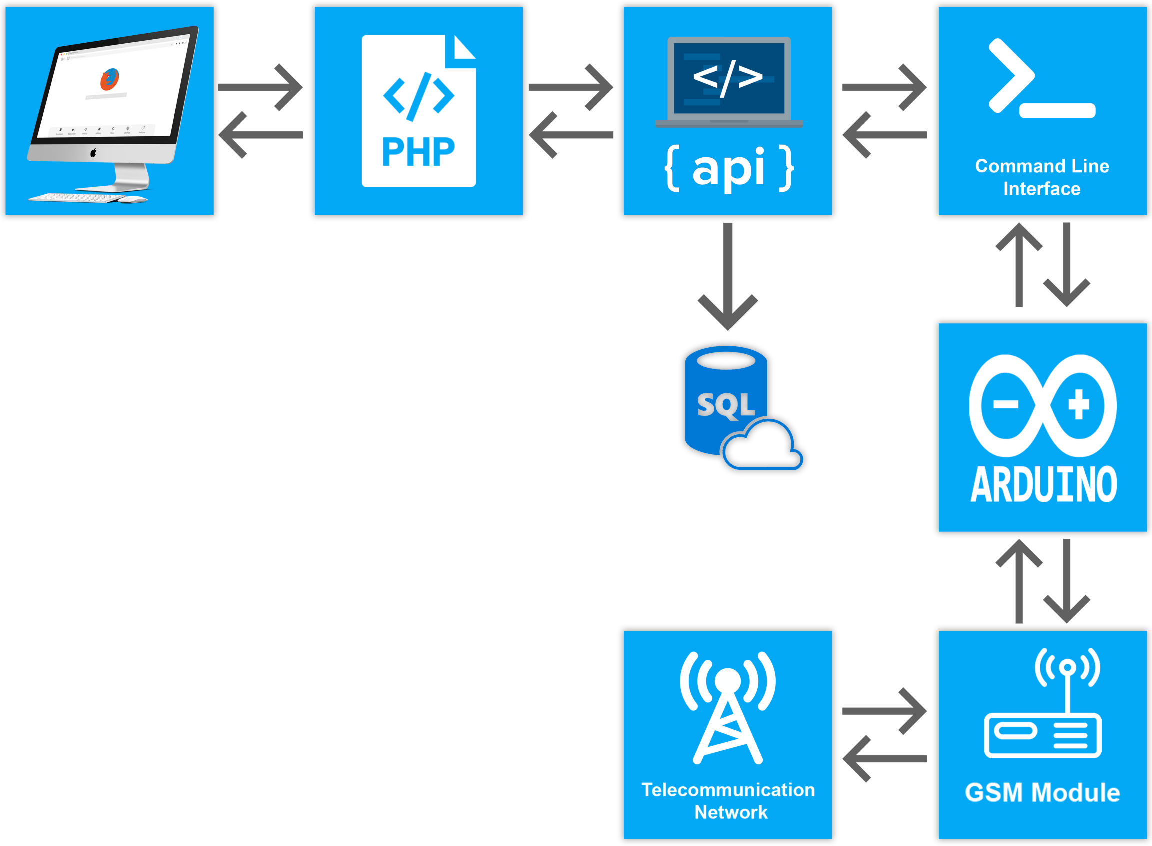
Introduction

Sending SMS through web server is a big challenge. An SMS gateway allows a computer (server) to send or receive Short Message Service (SMS) from a telecommunications network. There is lots of SMS gateway provides SMS service but their charges are bit high approx. ₹1 per SMS. Using this Gateway, you can send SMS at rate of less than ₹0.25. SIM900 based SMS Gateway is a three-layer secure Gateway which can send/receive SMS and call facility over internet. It provides SMS integration for website and application in low cost and best alternative for small business.

Block Diagram:



Working of SMS Gateway



1. **User Interface**: It is first layer of application which is written in HTML, JavaScript and CSS. This layer is responsible for making interface to user. The UI layer can be customized as per need and can be vary between websites you want to use.
2. **Backend:** Backend of this app is made with PHP. This is also customizable layer and can be customize by user. This layer is mainly responsible for user validation and first layer of security, that is provided by website.
3. **API:** API (Application Program Interface) provides programming interface to developer. It is a hidden layer and mainly responsible for basic input and output. The API does second layer of verification by key which is provided to API user (i.e. developer). Also this layer log data which is passed through API for security reasons. This layer is written in PHP.
4. **Command Line Interface:** Only API can access command line interface. This layer is for sending command to hardware (i.e. Arduino). This layer is written is C# and it send command in form of serial data to Communication (COM) ports. The CLI is the fundamental bridge between Telecommunication network and server.
5. **Arduino:** Arduino receive command form either computer(server) or form GSM module command line and process them and return output to server computer.
6. **GSM Module:** GSM module receive "AT" command form Arduino and response as per command received it send data to telecommunication network via SIM and response back to Arduino.

Sending Message and Call

The API accept http(https) POST request with valid access key form web server and results JSON variable which contains result code and result description of performed operation. In case of success it returns result code "1" with message "Success". The valid operations are.

1. Sending SMS (jQuery Ex)

```
$.ajax({
        type: 'POST',
        url: "../API/SMSGateway.php",
        data: "phone=" + <phone> + "&message=" + <message> + "&key=" + <key> +
"&operation=sms",
        dataType: "JSON",
        success: function(resultData) {
                console.log(resultData);
        }
});
```

2. Sending OTP (jQuery Ex)

```
$.ajax({
	type: 'POST',
	url: "../API/SMSGateway.php",
	data: "phone=" + <phone> + "&message="+< message> + "&key=" + <key>+
"&operation=otp",
	dataType: "JSON",
	success: function(resultData) {
		console.log(resultData);
	}
});
```

3. Call Verification (jQuery Ex)

```
$.ajax({
	type: 'POST',
	url: "../API/SMSGateway.php",
	data: "phone=" + <phone> + "&key=" + <key> + "&operation=call",
	dataType: "JSON",
	success: function(resultData) {
		console.log(resultData);
	}
});
```

4. Get last operation status (jQuery Ex)

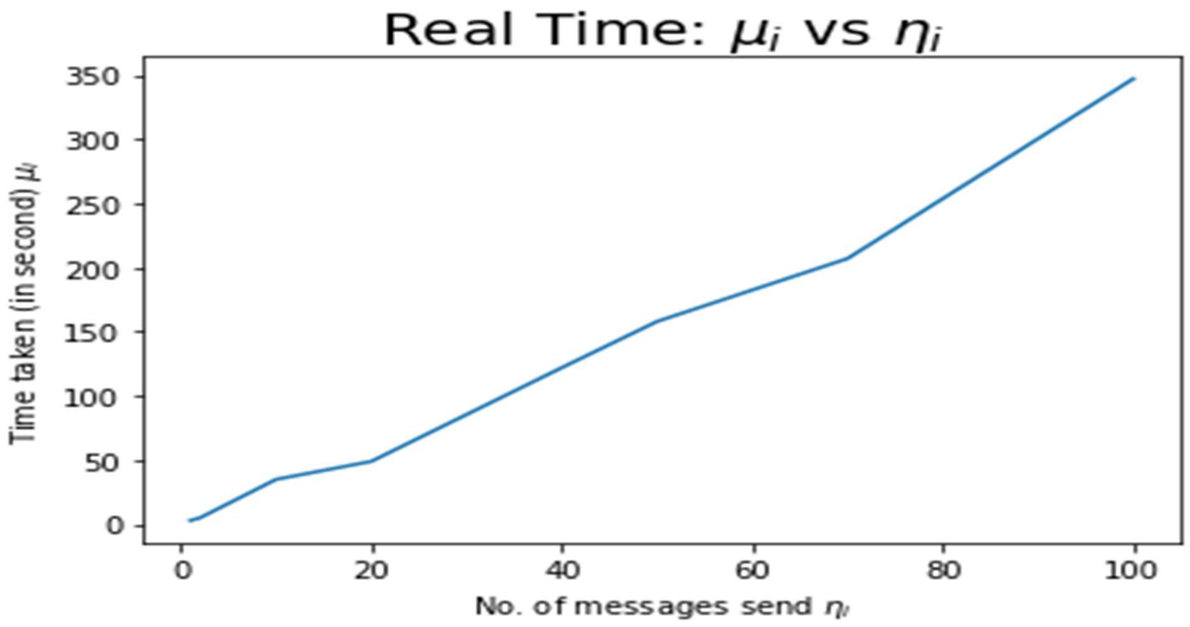
```
$.ajax({
	type: 'POST',
	url: "../API/SMSGateway.php",
	data: key=" + <key> + "&operation=get",
	dataType: "JSON",
	success: function(resultData) {
		console.log(resultData);
	}
});
```

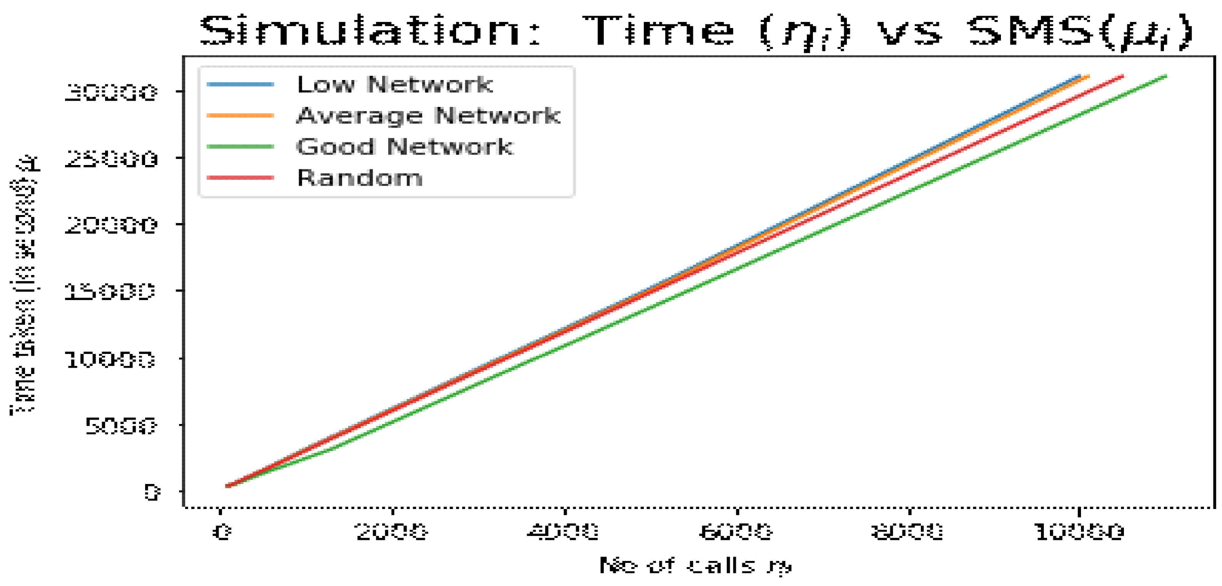
API Response code and Description

Response Code	***Description***
1	Success
21	API key validation error
38	SMS Gateway denied your access because you are using an invalid API key while SMS
51	Gateway denied your access because you are using a invalid API key
52	SMS Gateway not connected.
58	Gateway denied your access because you are using an invalid API key
80	Message can't be empty in SMS mode.
83	Message can't be null in SMS mode.
91	Invalid operation for Gateway or wrong parameter binding.
94	Invalid parameters list for Call.
104	Invalid parameters list for fetching data for Gateway.
107	Operation is not defined with Invalid parameter.
196	Message can't be null in OTP mode.
200	SQL Error, Gateway can't process your request at this time.
228	SQL Error, Gateway can't process your request at this time.
248	SQL Error, Gateway can't process your request at this time.
278	SQL Error, Gateway can't process your request at this time.
71	Invalid operation. (Denied by 2nd layer)
42	Gateway Exception. Error in *get* command.(3rd layer)
1	Success, Message in OTP mode
30	Telecommunication Network Exception. Error in *sim* command.
42	Telecommunication Network Exception. Error in *sms* command.
71	Telecommunication Network Exception. (In case of XSS script)
57	Telecommunication Network Exception. (In case of XSS script)
61	Telecommunication Network Exception. Crashed or Unexpected error.

Bulk SMS sending

SIM900 SMS Gateway support high speed SMS sending using one port of this gateway, the gateway is able to send SMS at the rate of 3 sec.(approx.) per SMS in good network to 5 sec.(approx.) in low network. The graph below is shown how fast the Getaway send a SMS for bulk query (It may possible receiving may take more than 15 min).

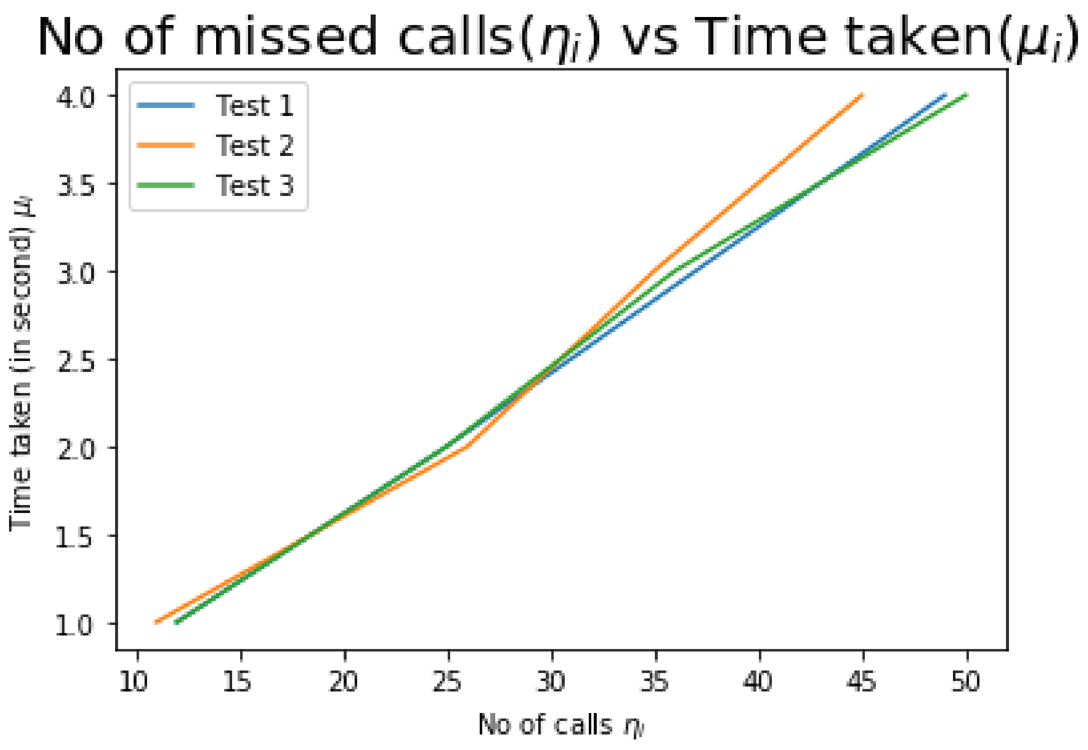




Missed call (*Checking Phone existence*)

The gate gives full ring to the given mobile number and return "Success" if done the status flag will "OK" if mobile ring else return the "FAIL". (Underdeveloped Future: This future currently checks the valid mobile numbers but in next version it will support OTP via call.)

The average time of one missed call is 12.5 sec in some cases and the simulation graph is:



Application

1. Sending bulk promotional message.

   Bulk SMS in India is a highly growing trend, almost all the businesses need to send text messages for various reasons. Whether it is about launching a new product or to make a nationwide announcement, bulk messaging is ideal for all. With SIM900 Gateway bulk SMS software we will manage all aspects of your requirements from start to finish!

   The Gateway is design as well as run campaign and create customized bulk SMS campaign based on your business needs and time frame, we will facilitate that for customer.

2. OTP verification

   An OTP is more secure than a static password, especially a user-created password, which is typically weak. OTP's may replace authentication login information or may be used in addition to it, to add another layer of security. OTP tokens are usually pocket-size fobs with a small screen that displays a number. The OTP PIN is generated by our platform and each generated PIN is tied to a distinct user and valid only for one login session.

   OTP generated by this gateway is more random and return by as JSON variable.

3. Checking phone number is in use or not.

The missed call future is introduced by getaway which check the weather a mobile number is in use or not.

Advantages over available Gateway

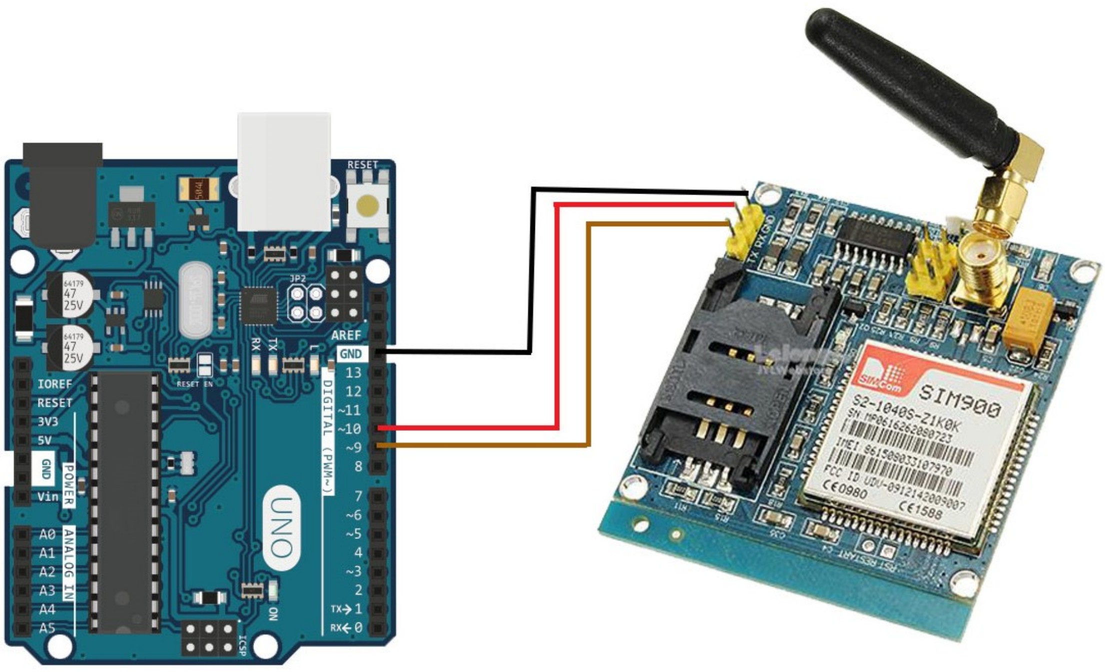
1. HTTP based API interface.
2. Low cost of installation.
3. Per SMS charge is very less to current available Gateways.
4. Calling Future.
5. Multi mobile no support.
6. Can send SMS in two different port at a time.
7. Hardware may install on different IoT device.

Limitation of Gateway

1. SMS sending speed is slow (3 to 5 second) due to dependency of network.
2. Call-back may take longer time than usual.
3. No instant hardware damage sensor or reporting mechanism (Develop soon).

Information About Project

Circuit:



UI Code (HTML):

```html
<!DOCTYPE html>
<html lang="en">
<head>
        <title>Send Message</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-
scale=1">
        <link rel="icon" type="image/png" href="images/icons/favicon.ico"/>
        <!--
                bootstrap(MIT), and material icon(google)
        -->
        <link rel="stylesheet" type="text/css"
href="api/bootstrap/css/bootstrap.min.css">
        <link rel="stylesheet" type="text/css" href="fonts/font-awesome-
4.7.0/css/font-awesome.min.css">
        <link rel="stylesheet" type="text/css"
href="api/animate/animate.css">
        <link rel="stylesheet" type="text/css" href="api/css-
hamburgers/hamburgers.min.css">
        <link rel="stylesheet" type="text/css"
href="api/select2/select2.min.css">
        <!--
                main designs
        -->
        <link rel="stylesheet" type="text/css" href="css/util.css">
        <link rel="stylesheet" type="text/css" href="css/main.css">
</head>
```

```
<body>

        <div class="limiter">
                <div class="container-login100">
                        <div class="wrap-login100">
                                <div class="login100-pic js-tilt" data-tilt>
                                        <a href="../logout.php"><img
src="images/img-01.png" alt="Logout"></a>
                                </div>

                                <form class="login100-form validate-form"
method="post">
                                        <span class="login100-form-title">
                                                Send a Message
                                        </span>

                                        <div class="wrap-input100 validate-
input" data-validate = "Valid phone is required: Ex : 1234567890">
                                                <input class="input100"
type="text" name="phone" placeholder="Phone No">
                                                <span class="focus-
input100"></span>
                                                <span class="symbol-
input100">
                                                        <i class="fa fa-
envelope" aria-hidden="true"></i>
                                                </span>
                                        </div>

                                        <div class="wrap-input100 validate-
input" data-validate = "Message required">
                                                <input class="input100"
type="text" name="messsage" placeholder="Message" value="Download our
latest app GATE CSE form play store http://bit.ly/GATECSE">
                                                <span class="focus-
input100"></span>
                                                <span class="symbol-
input100">
                                                        <i class="fa fa-
envelope" aria-hidden="true"></i>
                                                </span>
                                        </div>

                                        <div class="wrap-input100 validate-
input" data-validate = "Invalid format.">
                                                <input class="input100"
type="text" name="key" placeholder="API Key">
                                                <span class="focus-
input100"></span>
                                                <span class="symbol-
input100">
                                                        <i class="fa fa-lock"
aria-hidden="true"></i>
                                                </span>
                                        </div>

                                        <div class="container-login100-form-
btn">
                                                <button class="login100-form-
btn">
                                                        Send
```

```
                                                  </button>
                                          </div>

                                          <div class="text-center p-t-12">
                                                  <span class="txt1">
                                                          Generate
                                                  </span>
                                                  <a class="txt2"
href="api_key.php">
                                                          API Key?
                                                  </a>
                                          </div>

                                          <div class="text-center p-t-136">
                                                  <a class="txt2" href="#">
                                                          API Documantation
                                                          <i class="fa fa-long-
arrow-right m-l-5" aria-hidden="true"></i>
                                                  </a>
                                          </div>
                                  <input type="hidden" name="callback"
value="../sms/send/index.php" />
                                  <input type="hidden" name="operation" value="sms" />
                                             </form>
                                  </div>
                          </div>
          </div>



          <script src="api/jquery/jquery-3.2.1.min.js"></script>
          <script src="api/bootstrap/js/popper.js"></script>
          <script src="api/bootstrap/js/bootstrap.min.js"></script>
          <script src="api/select2/select2.min.js"></script>
          <script src="api/tilt/tilt.jquery.min.js"></script>
          <script >
                  $('.js-tilt').tilt({
                          scale: 1.1
                  })
          </script>
          <script src="js/main.js"></script>

</body>
</html>
```

UI CODE (JAVASCRIPT JQUERY):

```
(function ($) {
    "use strict";

    /*==================================================================
    [ Validate ]*/
    var input = $('.validate-input .input100');

    $('.validate-form').on('submit',function(){
        var check = true;
               event.preventDefault();
        for(var i=0; i<input.length; i++) {
            if(validate(input[i]) == false){
             showValidate(input[i]);
             check=false;
            }else if(i == 0){
                              if(input[i].value.length < 10 ||
input[i].value.trim().match(/^[0-9]*$/) == null){
                                      showValidate(input[i]);
                                      check=false;
                              }
                       }else if(i == 2){
                              if(input[i].value.length != 64){
                                      showValidate(input[i]);
                                      check=false;
                              }
                       }
        }
               if(check){
                       $(".login100-form-btn").prop('disabled', true);
                       $(".login100-form-btn").html('Sending...');
                       $.ajax({
                              type: 'POST',
                              url: "../API/SMSGateway.php",
                              data:
"phone="+input[0].value+"&message="+input[1].value+"&key="+input[2].value+"
&operation=sms",
                              dataType: "JSON",
                              success: function(resultData) {
                    $(".login100-form-btn").html('Updating...');
                                      console.log(resultData);
                    setTimeout(function(){
                        $.ajax({
                            type: 'POST',
                            url: "../API/SMSGateway.php",
                            data: "key="+input[2].value+"&operation=get",
                            dataType: "JSON",
                            success: function(getresultData) {
                                console.log(getresultData);
                                alert(getresultData);
                                $(".login100-form-btn").html('Send');
                                $(".login100-form-btn").prop('disabled',
false);
                            }
                        });
                    }, 3000);
                              }
                       });
               }
        return check;
    });


    $('.validate-form .input100').each(function(){
        $(this).focus(function(){
           hideValidate(this);
        });
    });

    function validate (input) {
        if($(input).attr('type') == 'email' || $(input).attr('name') ==
'email') {
            if($(input).val().trim().match(/^([a-zA-Z0-9_\-\.]+)@((\[[0-
9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.)|(([a-zA-Z0-9\-]+\.)+))([a-zA-Z]{1,5}|[0-
9]{1,3})(\]?)$/) == null) {
                return false;
            }
        }
        else {
            if($(input).val().trim() == ''){
                return false;
            }
        }
    }

    function showValidate(input) {
        var thisAlert = $(input).parent();

        $(thisAlert).addClass('alert-validate');
    }

    function hideValidate(input) {
        var thisAlert = $(input).parent();

        $(thisAlert).removeClass('alert-validate');
    }



})(jQuery);
```

ARDUINO CODE:

```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(9, 10);
String last_call_state = "NONE";
String form_computer;
String form_sms;
bool message_mode = false;
bool sms_ph = false;
int find_text(String needle, String haystack) {
  int foundpos = -1;
  for (int i = 0; i <= haystack.length() - needle.length(); i++) {
    if (haystack.substring(i,needle.length()+i) == needle) {
      foundpos = i;
    }
  }
  return foundpos;
}
void setup()
{
  Serial.begin(9600);
  mySerial.begin(9600);
}
void loop()
{
  if(Serial.available()){
    form_computer = Serial.readString();
    if(form_computer == "msg"){
      sms_ph = true;
      message_mode = true;
    }else if(message_mode){
      if(sms_ph){
        sms_ph = false;
        mySerial.println("AT+CMGS=\"" + form_computer+"\"");
      }else{
        message_mode = false;
        mySerial.println(form_computer+ "^Z");
      }
    }else if(form_computer == "last"){
      Serial.println(last_call_state);
    }else{
      mySerial.println(form_computer);
      last_call_state = "PROCESSING";
    }
  }else if(mySerial.available()){
    form_sms = mySerial.readString();
    if(find_text("OK", form_sms) != -1){
      last_call_state = "OK";
    }else{
      last_call_state = "FAIL";
    }
  }
}
```

Reference

1. http://www.propox.com/download/docs/SIM900.pdf
2. https://store.arduino.cc/usa/arduino-uno-rev3
3. http://www.ti.com/lit/ds/symlink/tusb4020bi.pdf
4. http://www.ti.com/lit/ds/sllse32g/sllse32g.pdf
5. https://en.wikipedia.org/wiki/GSM_frequency_bands