

# GüçlüKadro



***CENG 491***

***SENIOR DESIGN PROJECT***

***INITIAL DESIGN REPORT***

**GROUP MEMBERS**

***RECEP ALABAŞ***

***YAŞAT KADİR BİLGİN***

***BERK ÇAKIR***

***ÖZER DURMAZ***

# **INDEX**

<b>1 Introduction</b>	<b>3</b>
1.1 Purpose of this document	3
1.2 Problem Redefinition	3
1.3 Design Goals and Objectives	3
1.4 Design Constraints	4
<b>2 HARDWARE DESIGNS</b>	<b>5</b>
<b>2.1 REQUIRED HARDWARE</b>	<b>6</b>
<b>3 SOFTWARE DESIGNS</b>	<b>11</b>
<b>4 MODELING</b>	<b>14</b>
4.1 Use Case Diagrams	16
4.1.1 Use Case Diagram I	16
4.1.2 Use Case Diagram II	16
4.1.3 Use Case Diagram III	18
4.2 SEQUENCE DIAGRAMS	21
4.2.1 Change Settings Module	21
4.2.2 Display Module	26
4.3 DATA FLOW DIAGRAMS	30
4.3.1 LEVEL 0	30
4.3.2 LEVEL 1	32
4.3.3 LEVEL 2	35
<b>5 USER INTERFACES</b>	<b>40</b>
<b>6 GANNT CHART</b>	<b>47</b>

## **1) INTRODUCTION**

### **1.1) Purpose of this document**

This document is prepared to supply an initial design to our project.

The purpose in this document is to show our design process and the solutions for the design problems. Our designs are not the complete and final ones, but they will be suitable for this report because this is an initial design report. However, we will be work much on these till our final design project and we will introduce the perfect designs in the next report.

### **1.2) Problem Redefinition**

Wireless communications is a rapidly growing part of the communication systems. While the wireless communication quality improves, the demand of using wireless technology increases. For that reason wireless access points have an important part at the market. However, it is important to develop a different type of wireless access point to have a big portion from the market. Our prototype will be different by its LCD monitor. The user can access some kind data from this monitor namely, the number of users, their names and their connection speeds connected to the access point. Also, it will be give information about their connection period of times and about the errors in the connection. Moreover, the user will be able to block some unwanted mac addresses.

### **1.3) Design Goals and Objectives**

#### **a) Extensibility:**

However we showed our designs in this report, they are not very strict ones. Our designs have extensibility in order to add new specialties to the project. We tried to design our prototype considering that any other improvement will be added later.

#### **b) Robustness:**

The product should be able to deal with invalid user inputs. To illustrate, our product will be able to manage with pushing of its buttons at the same time, in other words wanting different data at the same time. It will be able to do error checking and returns this error reports to the user by the LCD display.

**c) Reliability:**

We are planning to program our product, as reliable as possible. In order to accomplish this aim, we will test our program many times. We think that we will succeed to produce a bug free product.

**d) Functionality:**

Our product's functions were mentioned in the previous report, namely Requirement Analysis Report. Our product will function in the manner that was visited in our previous report.

**e) Usability:**

Usability is one of the most important issues in our project. Our general user interface should affect the user by its easiness. The user will not encounter with any complexity when using the product. The user will sense the familiarity and the user-friendliness of the menus.

## **1.5) Design Constraints**

**a) Performance:**

Due to the fact that performance is a very important issue for users, we considered the efficiency of our product in our designs. Our product will work very efficient and show high performance.

**b) Time:**

According to the Ceng 491 syllabus, our project must be finished by June. Because of that reason we must work hard and provide a prototype at the end of this semester. We must follow our plans stated in our Gannt chart in this report.

**c) Funding:**

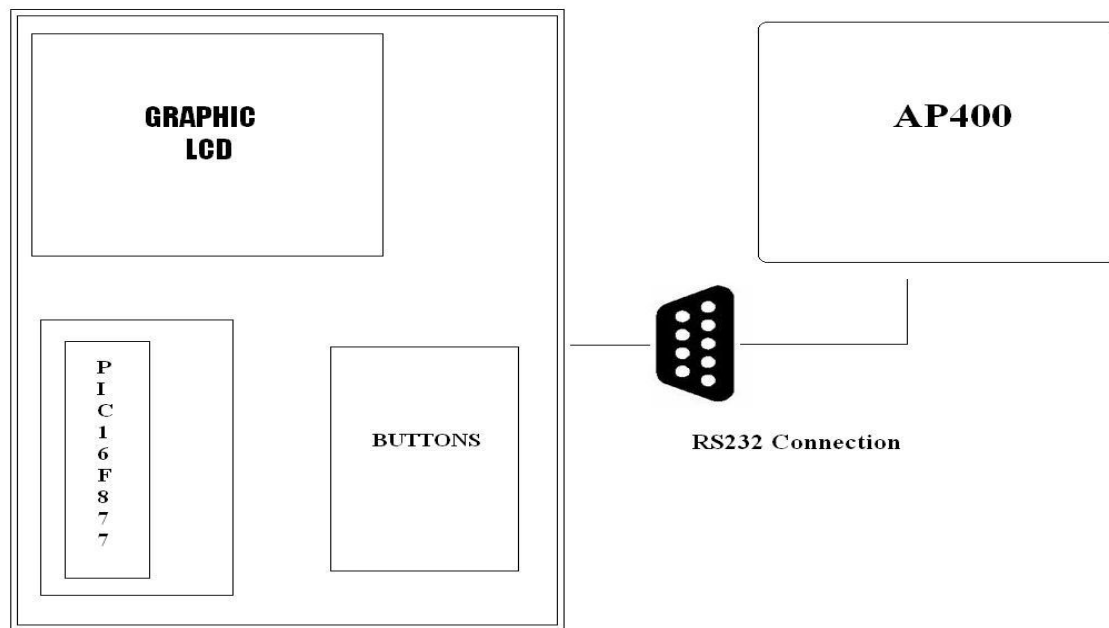
Computer Engineering Department is going to pay us about 50\$ for the expenditures of us for our project. Until now, we bought a 128x64 graphic LCD and RS-232 to USB converter. These didn't cost as much as the fund, so we are able buy our other necessities.

#### **d) Resource:**

We need some software and hardware resources in order to complete our project. Software resources are supplied by our teaching assistant and the company. Also, we can provide the hardware resources by the help of our teaching assistant. In addition to them, we need some books and articles to learn the details of our working scope. Every member of the project including the company, and the teaching assistant are providing us these kinds of useful documents. Also, we are trying to find documents via internet and Metu Library.

## **2) HARDWARE DESIGN**

The hardware design of our project consists of two main parts. (Figure 1). First one is Airties AP400 access point. This is the access point that we are going to use it with special hardware in order to design a model of an improved access point. The hardware of AP400 is discussed in the next section. The second part is the special hardware we are going to use to construct the model. It will consist of a PCB (printed circuit board) and the required hardware (a microcontroller, an LCD, an oscillator, etc.) on it. The details of the PCB are discussed in the next chapter. AP400 will be connected to the PCB with an RS232 cable. With this connection the needed information about the network traffic will be transferred to the board and will be displayed on the graphic LCD. Also there will be some buttons on the board which will be used to change the screen, to pass from one information display to another (e.g. while the default information, date and time is displaying, pressing the button will show the users connected at that time.) The type of the microcontroller we are going to use is pic16f877, just because it can fulfill our demands and we are familiar it from ceng336.



*Figure 1*

## **2.1)REQUIRED HARDWARE**

### **AIRTIES AP400**

Airties AP400 is an access point (a special hardware software combination which can connect wireless communication devices) that is an embedded system design running Linux. It is IEEE 802.11b/g compliant. It has WPA (Wi-Fi Protected Access), WPA2 (IEEE 802.11i), WEP (Wired Equivalent Privacy) encryption and 802.1x standards. Also AP400 has ADM6996f, four-port 10/100 Mbps TX/FX plus two 10/100 MAC port Ethernet switch controller with all ports supporting 10/100 Mbps Full/Half duplex, and this chip has its own embedded memory.



*Picture 1*

## MICROCONTROLLER

### PIC16F877 Pin Diagram

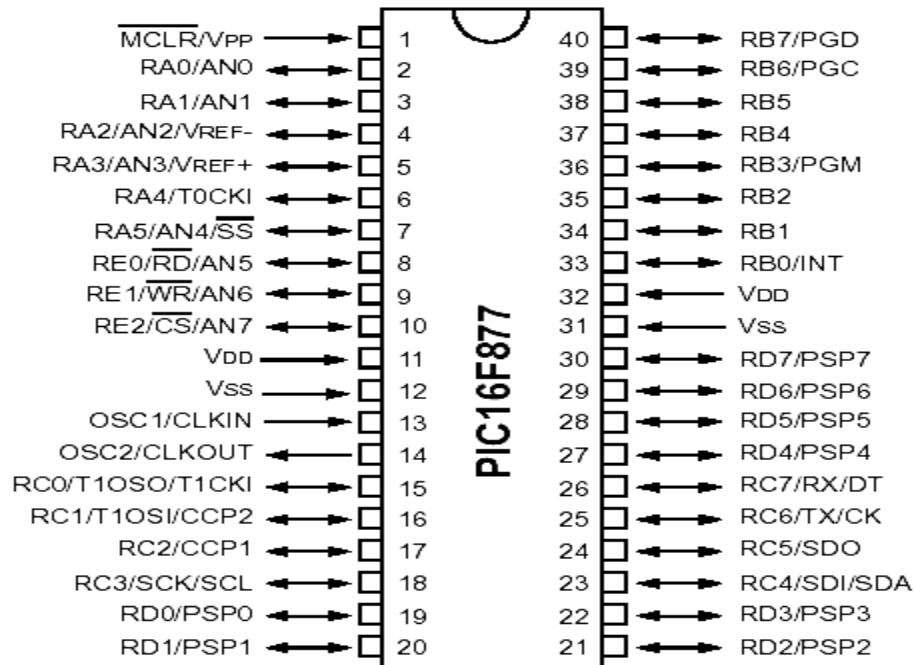


Figure 2

### PIC16F877 Main Features

- Operating speed: DC - 20 MHz clock input  
DC - 200 ns instruction cycle
- Up to 8K x 14 words of FLASH Program Memory,  
Up to 368 x 8 bytes of Data Memory (RAM)  
Up to 256 x 8 bytes of EEPROM Data Memory
- Interrupt capability (up to 14 sources)
- Wide operating voltage range: 2.0V to 5.5V

## Pinout Description

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	13	14	30	I	ST/CMOS <sup>(4)</sup>	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	14	15	31	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
MCLR/VPP	1	2	18	I/P	ST	Master Clear (Reset) input or programming voltage input. This pin is an active low RESET to the device.
RA0/AN0	2	3	19	I/O	TTL	PORTA is a bi-directional I/O port. RA0 can also be analog input0. RA1 can also be analog input1. RA2 can also be analog input2 or negative analog reference voltage. RA3 can also be analog input3 or positive analog reference voltage. RA4 can also be the clock input to the Timer0 timer/counter. Output is open drain type. RA5 can also be analog input4 or the slave select for the synchronous serial port.
RA1/AN1	3	4	20	I/O	TTL	
RA2/AN2/VREF-	4	5	21	I/O	TTL	
RA3/AN3/VREF+	5	6	22	I/O	TTL	
RA4/T0CKI	6	7	23	I/O	ST	
RA5/SS/AN4	7	8	24	I/O	TTL	
RB0/INT	33	36	8	I/O	TTL/ST <sup>(1)</sup>	PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. RB0 can also be the external interrupt pin.  RB3 can also be the low voltage programming input. Interrupt-on-change pin. Interrupt-on-change pin. Interrupt-on-change pin or In-Circuit Debugger pin. Serial programming clock. Interrupt-on-change pin or In-Circuit Debugger pin. Serial programming data.
RB1	34	37	9	I/O	TTL	
RB2	35	38	10	I/O	TTL	
RB3/PGM	36	39	11	I/O	TTL	
RB4	37	41	14	I/O	TTL	
RB5	38	42	15	I/O	TTL	
RB6/PGC	39	43	16	I/O	TTL/ST <sup>(2)</sup>	
RB7/PGD	40	44	17	I/O	TTL/ST <sup>(2)</sup>	
RC0/T1OSO/T1CKI	15	16	32	I/O	ST	PORTC is a bi-directional I/O port. RC0 can also be the Timer1 oscillator output or a Timer1 clock input. RC1 can also be the Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output. RC2 can also be the Capture1 input/Compare1 output/PWM1 output. RC3 can also be the synchronous serial clock input/output for both SPI and I <sup>2</sup> C modes. RC4 can also be the SPI Data In (SPI mode) or data I/O (I <sup>2</sup> C mode). RC5 can also be the SPI Data Out (SPI mode). RC6 can also be the USART Asynchronous Transmit or Synchronous Clock. RC7 can also be the USART Asynchronous Receive or Synchronous Data.
RC1/T1OSI/CCP2	16	18	35	I/O	ST	
RC2/CCP1	17	19	36	I/O	ST	
RC3/SCK/SCL	18	20	37	I/O	ST	
RC4/SDI/SDA	23	25	42	I/O	ST	
RC5/SDO	24	26	43	I/O	ST	
RC6/TX/CK	25	27	44	I/O	ST	
RC7/RX/DT	26	29	1	I/O	ST	
RD0/PSP0	19	21	38	I/O	ST/TTL <sup>(3)</sup>	PORTD is a bi-directional I/O port or parallel slave port when interfacing to a microprocessor bus.
RD1/PSP1	20	22	39	I/O	ST/TTL <sup>(3)</sup>	
RD2/PSP2	21	23	40	I/O	ST/TTL <sup>(3)</sup>	
RD3/PSP3	22	24	41	I/O	ST/TTL <sup>(3)</sup>	
RD4/PSP4	27	30	2	I/O	ST/TTL <sup>(3)</sup>	
RD5/PSP5	28	31	3	I/O	ST/TTL <sup>(3)</sup>	
RD6/PSP6	29	32	4	I/O	ST/TTL <sup>(3)</sup>	
RD7/PSP7	30	33	5	I/O	ST/TTL <sup>(3)</sup>	
RE0/ $\overline{RD}$ /AN5	8	9	25	I/O	ST/TTL <sup>(3)</sup>	PORTE is a bi-directional I/O port. RE0 can also be read control for the parallel slave port, or analog input5. RE1 can also be write control for the parallel slave port, or analog input6. RE2 can also be select control for the parallel slave port, or analog input7.
RE1/ $\overline{WR}$ /AN6	9	10	26	I/O	ST/TTL <sup>(3)</sup>	
RE2/ $\overline{CS}$ /AN7	10	11	27	I/O	ST/TTL <sup>(3)</sup>	
Vss	12,31	13,34	6,29	P	—	Ground reference for logic and I/O pins.
VDD	11,32	12,35	7,28	P	—	Positive supply for logic and I/O pins.
NC	—	1,17,28,40	12,13,33,34		—	These pins are not internally connected. These pins should be left unconnected.

Legend: I = input

O = output

— = Not used

I/O = input/output

TTL = TTL input

P = power

ST = Schmitt Trigger input

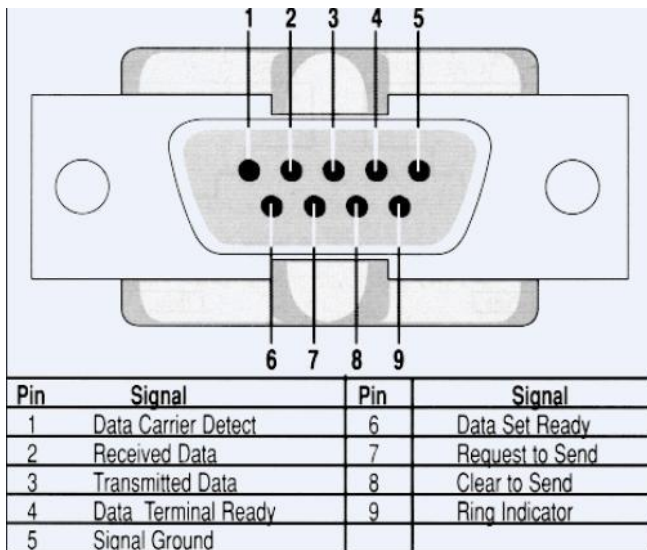


## PIC16F877 Functionality in Our Design

The microcontroller on our PCB will be responsible for three major operations. The communication with AP400 (data transfer between AP400 and the board), ensuring the LCD to display the information get from AP400 (the suitable graphic display about the network traffic), and to change the screen when a button is pressed (this will be done by interrupts).

## RS232

The communication between our designed board and AP400 will be established in



RS232 standards. In RS-232, data is sent as a time-series of bits. Both synchronous and asynchronous transmissions are supported by the standard. The connector type we are going to use is a DE-9 (D-subminiature 9 pin) connector and its pinout is described in Figure 3. Since the voltages of the board and AP400 are not the same (5V-3.3 V) the voltage translation will be done by MAX3378 integrated circuit.

*Figure 3*

## GRAPHIC LCD

The network information will be displayed on the LCD. The LCD we are going to use is HY-12864K-202-R model 128x64 graphic LCD (Figure 4).



design from concept to completion within a single application, Altium designer will be our design software.

Main features of Altium Designer:

- ✓ Synchronous design at PCB and FPGA (Field-Programmable Gate Array, a type of logic chip that can be programmed) levels, from concept to design in one application.
- ✓ With its functional user interface, hierarchical, multi-channel schematic capture.
- ✓ Fast PCB design with Pre- and post-layout signal integrity analysis after adding components.
- ✓ Situs Topological auto routing
- ✓ A design library including over 68,000 IP components and versatile archive for schematic level FPGA design.
- ✓ Versatile project management, design tools with bidirectional synchronization from schematic to PCB and from PCB to FPGA.
- ✓ Automatic and interactive FPGA pin optimization for routing.
- ✓ PCB-FPGA I/O synchronization.

Since PCB design is a relatively later work in our project, when we have much time to decide the software we are going to use. Because of the case we can change the software, we did a simple search about these tools, and the first alternative for Altium designer is the other software of Altium Limited, P-CAD. This software is similar to Altium designer since it is their another tool about PCB design, but it is focused on PCB design

After the design of the board with one of these software we are going to print it. How we will do it is not determined at this moment. Since it is a relatively hard work and requires experience, we may get it printed. Another way is we may print it.

### **3) SOFTWARE DESIGN**

Since AP400 has an embedded Linux system in it, all our programming part will be done in Linux environment. AP400 runs Linux and it has its own root file system (the necessary files needed to for the Linux system to run). There are some directories for some purposes (/bin includes the essential binaries considered part of the system, whereas /etc includes system configuration files.) Since our scripts will be put in one of these directories, their properties are very important for the system run properly. (E.g. if the script is put in /tmp

directory, it will be deleted after AP400 is restarted.) As all other projects, the details are important and the details of our software design are discussed in the later sections.

## **REDBOOT**

Redboot is an open source application for embedded systems. AP400 uses Redboot. It is based on the eCos Hardware Abstraction Layer. In our design, it is important because Redboot allows loading and execution of a kernel or executable from

either tftp or via the serial console. In order to reload the root files to AP400 we use Redboot. Our final product will include the existing files of the kernel plus our scripts, and the user will reload the kernel by Redboot and it will be ready to use.

## **PROGRAMMING ISSUES**

Since our project is building a model of an improved AP400, we will add new functions to it in order to aim more customers. This “adding new functions” part will be done by our scripts. AP400 runs Linux and it has its own scripts (some of which can be useful for us) and we will add new scripts next to them (transferring this scripts from computer to AP400 is discussed in “Special Software for Software Design” section). These scripts will get the needed information about the network traffic when it is wanted (the related button on the PCB is pressed). Since many of our goals need to keep the related information since the time AP400 was opened, also we will keep this information, by a special algorithm in order to minimize the size of it, in AP400. As all other embedded systems, memory management is an important criterion for us, because the capacity of AP400 for running the scripts and keeping the information is relatively small for a normal computer program. So we will be careful about our scripts and use some algorithms to minimize the memory space we need to run the scripts and keep the information

The programming language we are going to use may change during the project. But we believe two of them can fulfill our demand. We will use C Programming and Bash Scripting. When we use C for the programs, we will compile it with MIPS gcc cross compiler and put its executable into one of the directories. On the other hand, Bash scripting is relatively faster than C; we will just put the script into the directory we want.

## A Simple Pseudo-Code for Out Main Script

*mainscript:*

*if IdleTime is bigger than IdleTimeConstant*

*DisplayDateandTime*

*else*

*WaitforInterrupt*

Here *WaitforInterrupt* is responsible for the interrupts. As mentioned in the hardware design chapter, interrupts will be used when user wanted one of the information about the network traffic to be displayed. This will be done by buttons on the PCB. According to the type of the button *WaitforInterrupt* will call the suitable script to get the needed information. *DisplayDateandTime* function is for the displaying the current date and time, when the user did not press any buttons for some time.

## How to Get the Information

As mentioned above AP400 runs Linux and has its own scripts. Some of these scripts, for example, are the result of a project, The Linux Wireless Extension and the Wireless Tools, an open source project. These project codes may be example of writing wireless tools with C. We may write some programs like these ones to get the information, compile it, and transfer it to the AP400; or we can use the outputs of these tools with a suitable script (that we write with Bash Scripting). These are two examples of getting the network information. For example wlanconfig command returns the user mac addresses, with some other information and display them on the screen. Since we need to display only the mac addresses of the users, we can do this in two ways; first we can rewrite the C code of wlanconfig in order to get only mac addresses, or second we can get the output of wlanconfig with a Bash script and choose only the mac addresses.

## **Pic Programming**

After getting the information and transferring it to the PCB, a program is needed to display the information. We are going to use C for this program and it will be suitable with our microcontroller, PIC16F877.

## **Special Software for Software Design**

For the communication between the computer and AP400, we use a graphical serial terminal program, cutecom. The file transfer, e.g. transferring a script from the computer to AP400 or reloading the root files to the AP400 can be done by entering the commands from cutecom and by help of a tftp server (for a fast transfer). The communication between the computer and AP400 is established with an RS232 cable.

Some features of cutecom:

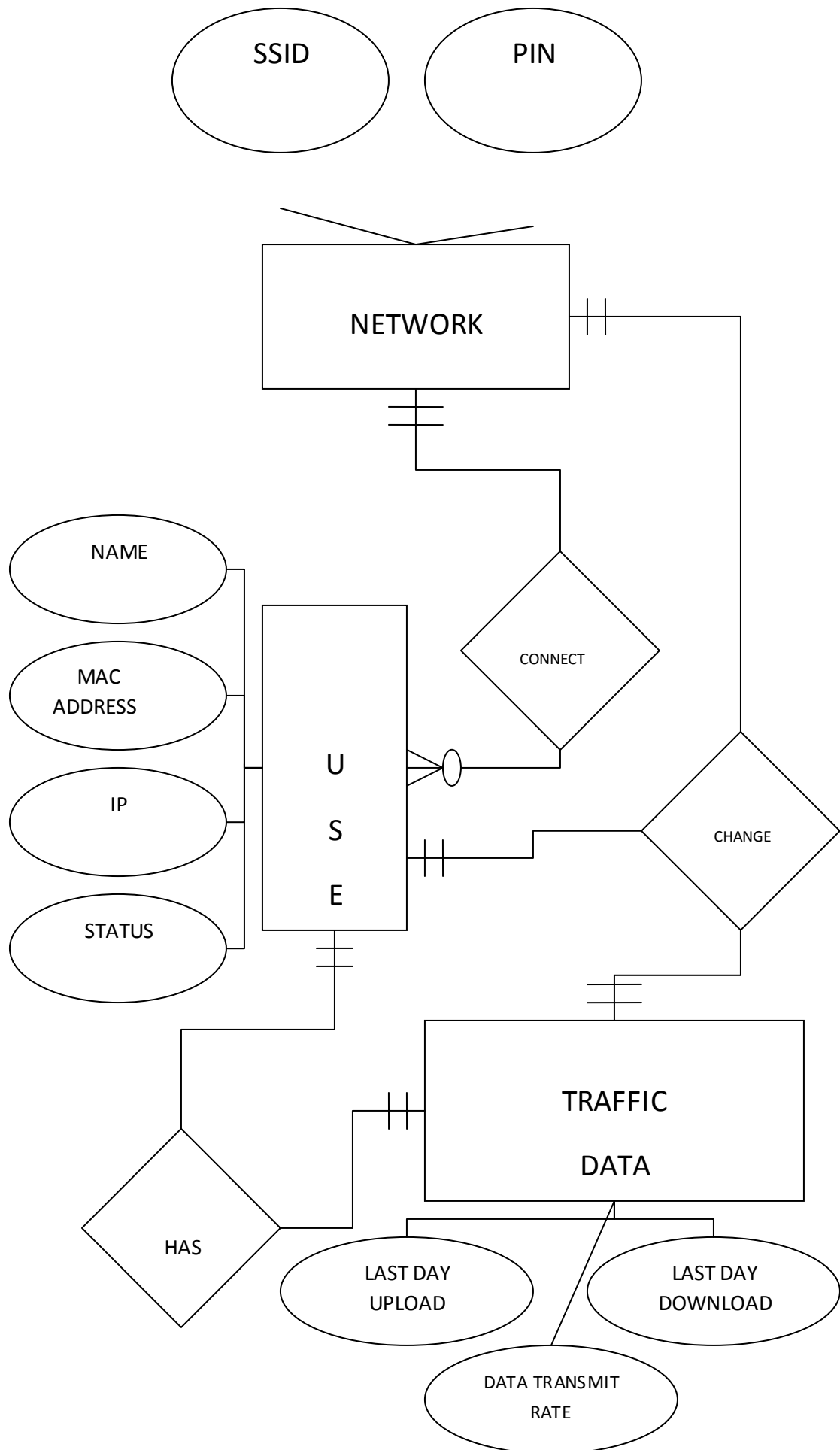
- Ctrl+C, Ctrl+Q and Ctrl+S control sequences work,
- input history,
- xmodem, ymodem, zmodem support (requires the sz tools)
- hexadecimal input and output

In order to compile the pic program we are going to use SDCC, a retargettable, optimizing ANSI - C compiler. SDCC supports PIC16 and PIC18 series. It is free open source software.

To load the .hex file to the pic, we are going to use winpic800. We will use it because we are familiar to it from ceng336 and it is easy to use.

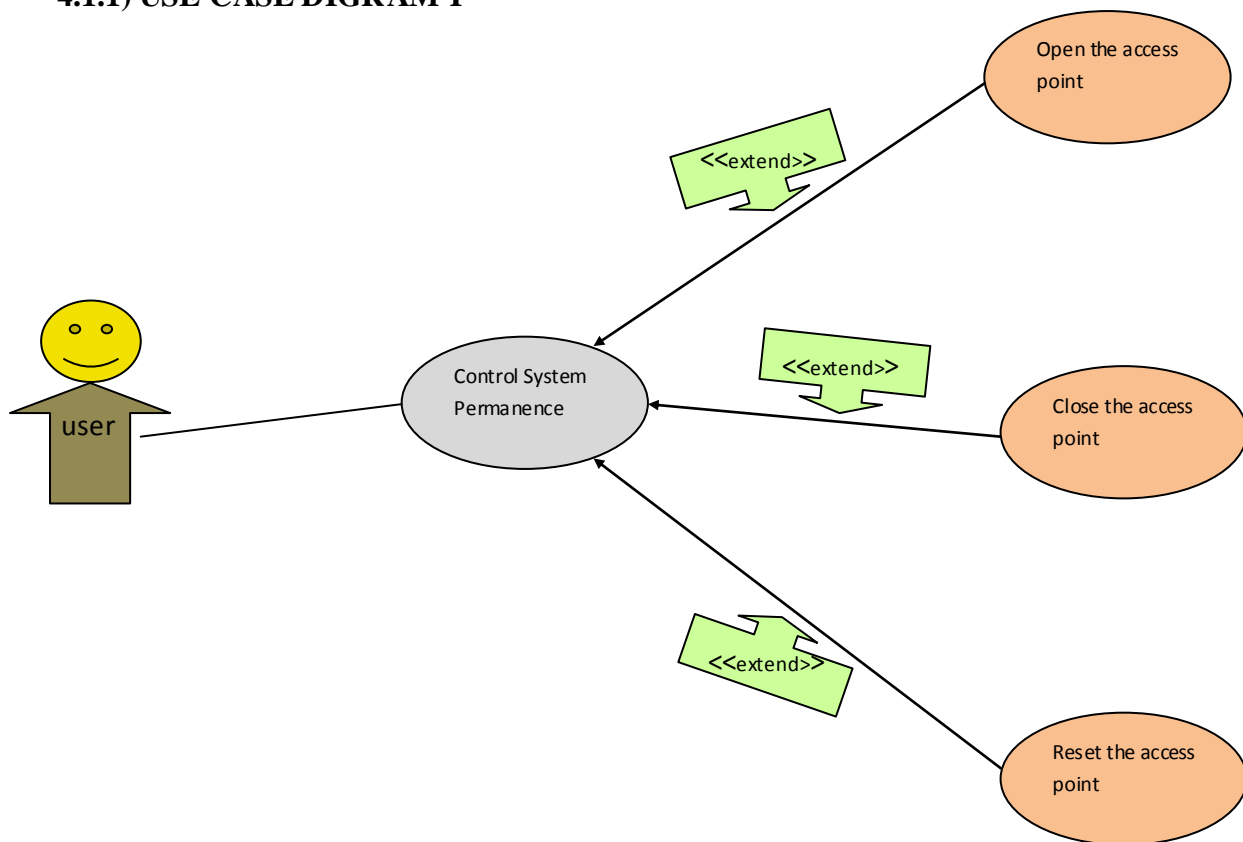
## **4) MODELING**

The related network traffic, network properties, and user data will be kept in AP400. The data diagram is above. User has his name, mac address, IP, and status of the connection. One user can connect exactly one network, but the network can accept zero or more user connections. The traffic data consists of last day upload, last day download, and data transmit rate. Exactly one user can change the network properties and traffic data information. The below diagram shows the data relationships.



## 4.1) USE CASE DIAGRAMS

### 4.1.1) USE CASE DIGRAM I

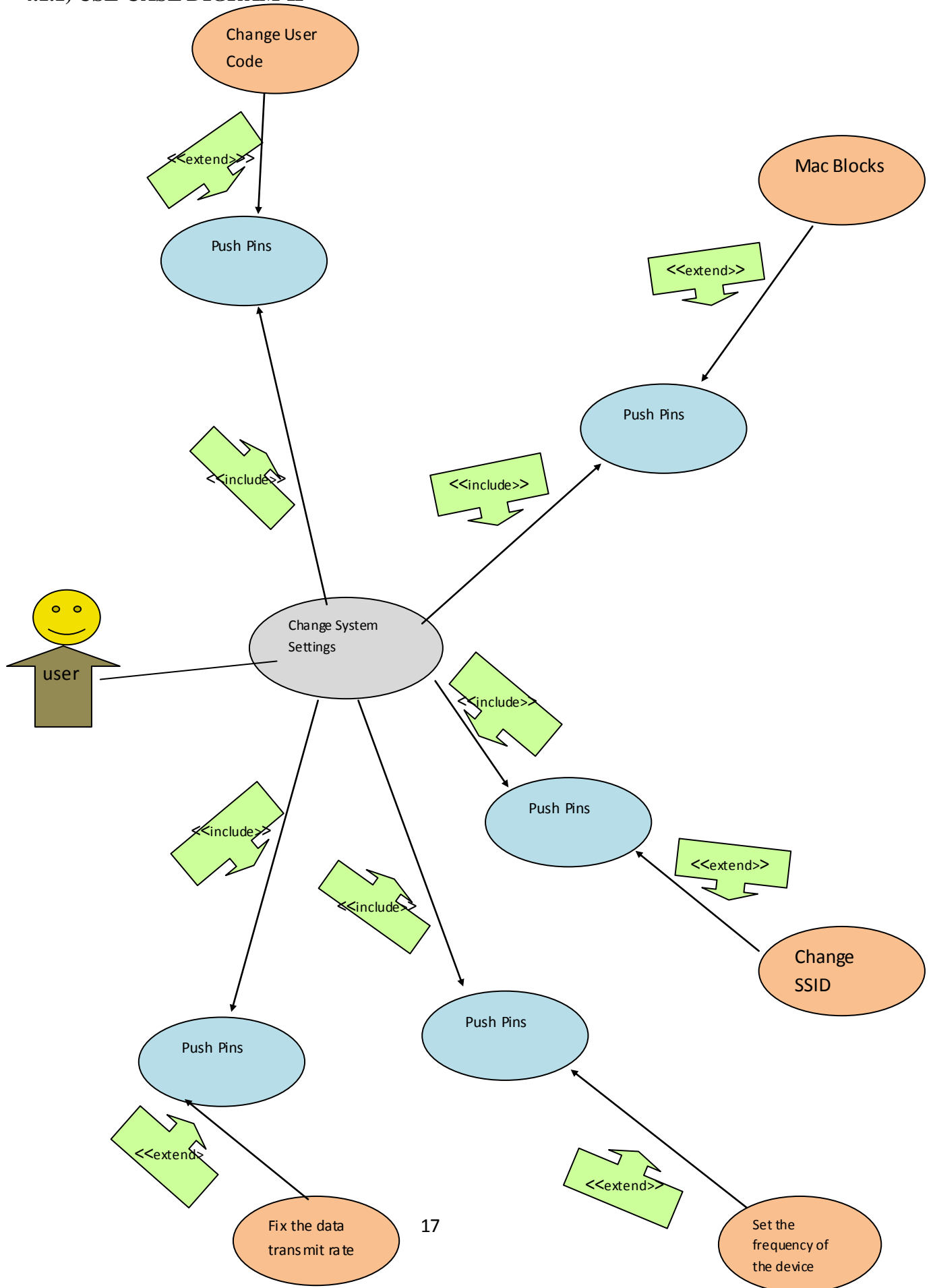


Managing system is one of the vital point of a project . In our project user must also act as admin of the system . He or she must be responsible for the permanence of the system . To make the system available every time user must take some action . This actions can be listed as followed :

- **Open the access point** : User must open access point before clients connect to it . This may be seen as very easy role but it is important to initialize access point properly .
- **Close the access point** : User must close access point when needed . Because it will be harmful to be open every time .Due to its structure it must be closed when it is not used .
- **Reset the access point** : To avoid the clustering of data user must reset the system periodically . Also this action will raise the system performance .



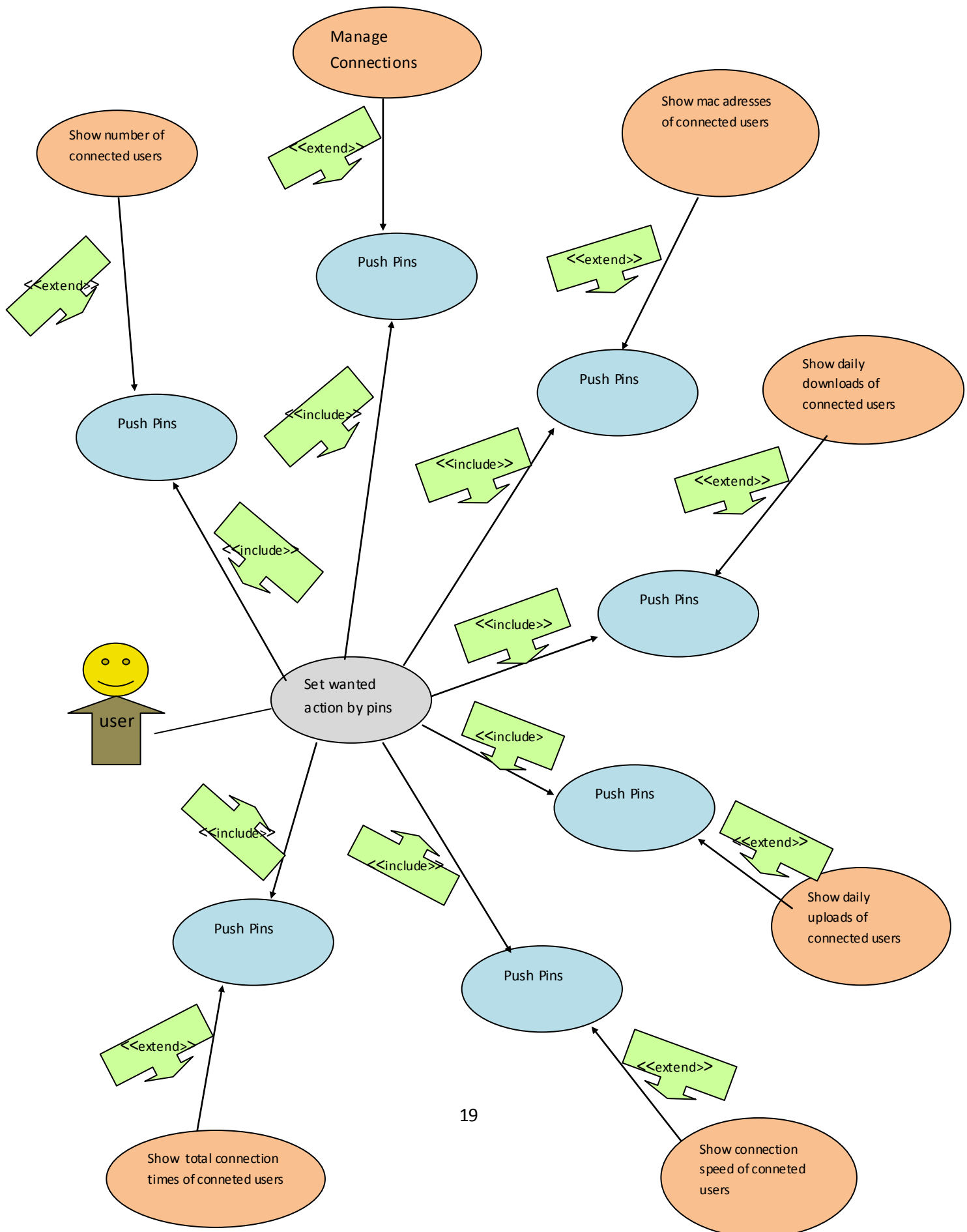
#### 4.1.1) USE CASE DIGRAM II



Another important point of our project is changing system settings. User must configure the system for maximal performance and security. Because every action performed due to these settings . In our project user can change the pin and SSID of the system , manage the connections , set frequency of device , fix the data transmit rate . Also if user wants he or she can block some macs . All these actions are very beneficial to the system security and performance .

- **Change User Code:** User can change the pin to increase the security level of the system . Because a stable pin can be easily stolen . Changing pin regularly will block the unauthorized user to connect the system .
- **Manage Connection :** User can manage the connections to avoid over-using of the system . When user notices that some users make a lot of jam in the network he or she can warn the user or do something to avoid this . At this point our system will only show other connected user's connection status to the user. Then , immediately he or she can take precautions to overcome this problem.
- **Mac Block :** If a connected user makes unwanted action then it will be possible to block his or her mac . Disturbing other connected users , keeping network busy can be examples of unwanted action .
- **Change SSID :** User can change the SSID of the system to make network more user friendly . For example a company can change the SSID of the system to their company name .
- **Set Frequency and Channel Of The Device :** Which channel to use and its frequency can be set by user. User can coordinate network traffic by this action . It will be very beneficial to maximize the network usage.
- **Fix The Data Transmit Rate :** User can fix the data transmit rate in order to have a single transmit rate . For example, all the rate other than the user specified will automatically become equal to it .

#### 4.1.1) USE CASE DIGRAM III



In our project user will select what he or she wants to do by pressing pins of the board. Initially in the lcd display date will be seen. When user wants to do something he or she will press to the pin and a selection menu will be appear. Again pressing to the pins user will select action to do. Then selected action will be done by AP400 with the help of pic 16f877 . All this process is the goal of our project. Because user will see what he or she wants. So this process is very important part of out project.

- **Show Number of Connected Users :** By pressing pins user can see the connected number of users when he or she wants. According to this information user can change the system settings. For example user can fix the data transmit rate to adequate value with network traffic. Also user can change the device frequency according to the number of connected users .
- **Show Mac Addresses of Connected Users :** User can see the mac addresses of connected users again pressing pins of board. Mac addresses are unique so this information is very important. Because by mac addresses every user will be found very easily. User can do whatever he or she wants by only knowing the mac address of connected user. For example only knowing mac addresses of connected user will be enough to block him or her.
- **Show Daily Downloads Of Connected Users :** User can see the daily downloads of the connected users by pressing pins of board. Knowing daily downloads will be important due to use of project. For example in the internet cafe's user may be have quota on download size. So he or she can take precautions according to this information .
- **Show Daily Downloads Of Connected Users :** User can see the daily uploads of the connected users again by pressing pins of the board. Like knowing daily downloads, knowing daily uploads will be important due to use of project. This information will be very important to the internet users who have quota on their accounts .
- **Show Connection Speed Of Connected Users :** User can see the connection speeds of connected user by pressing pins of the board. This information will be beneficial to give feedback to connected user about his connection. If the

connected users' speed is low then user can offer him to reset or repair, his or her connection. This process will help connected users to increase their benefits from device.

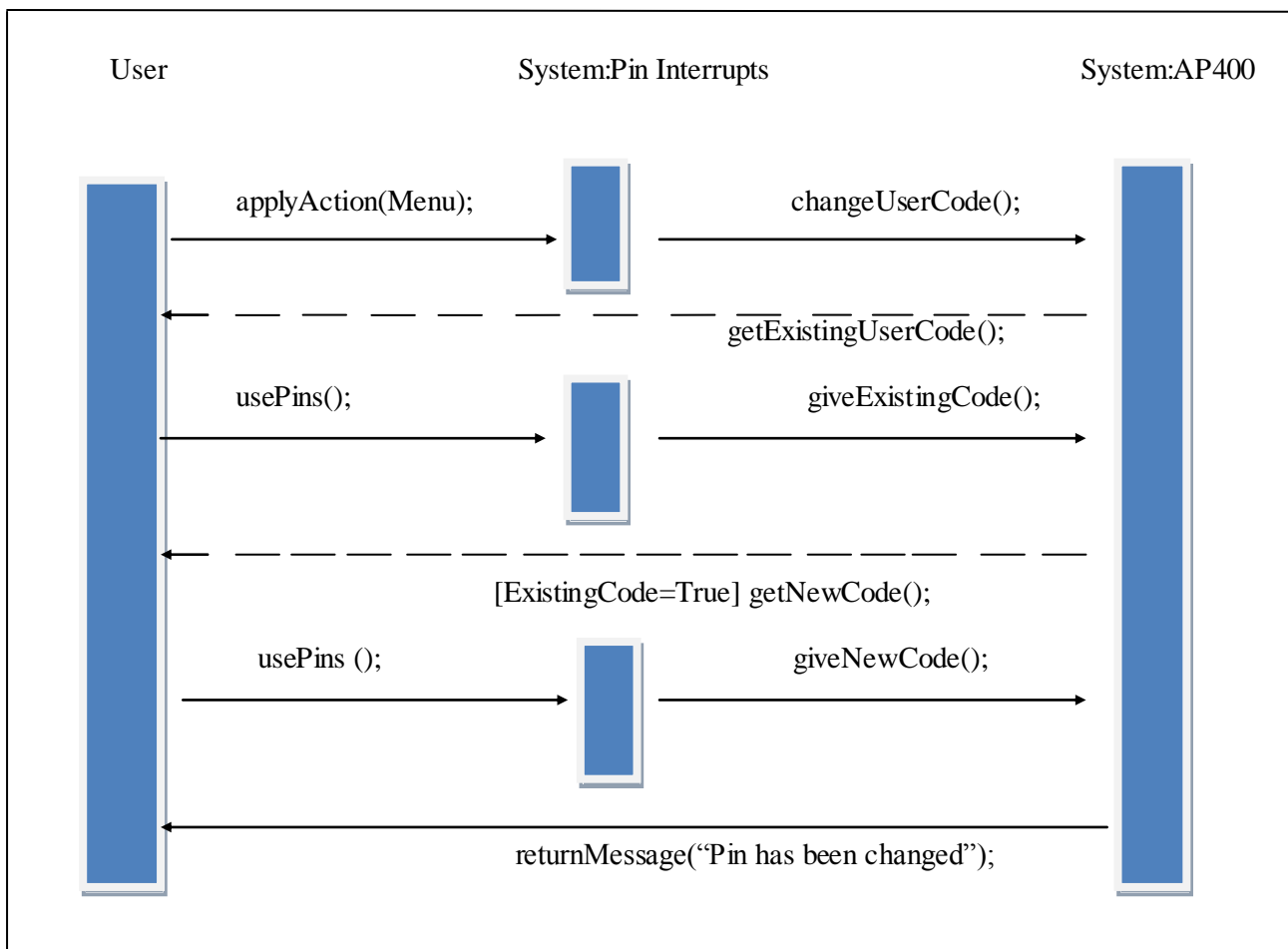
- **Show Total Connection Time Of Connected Users** : User can see the total connection time of the connected users by pushing pins of the board. This info again can be useful due to use of project. This information will be very beneficial in big companies or internet cafes.

## 4.2) SEQUENCE DIAGRAMS

### 4.2.1) Change Settings Module

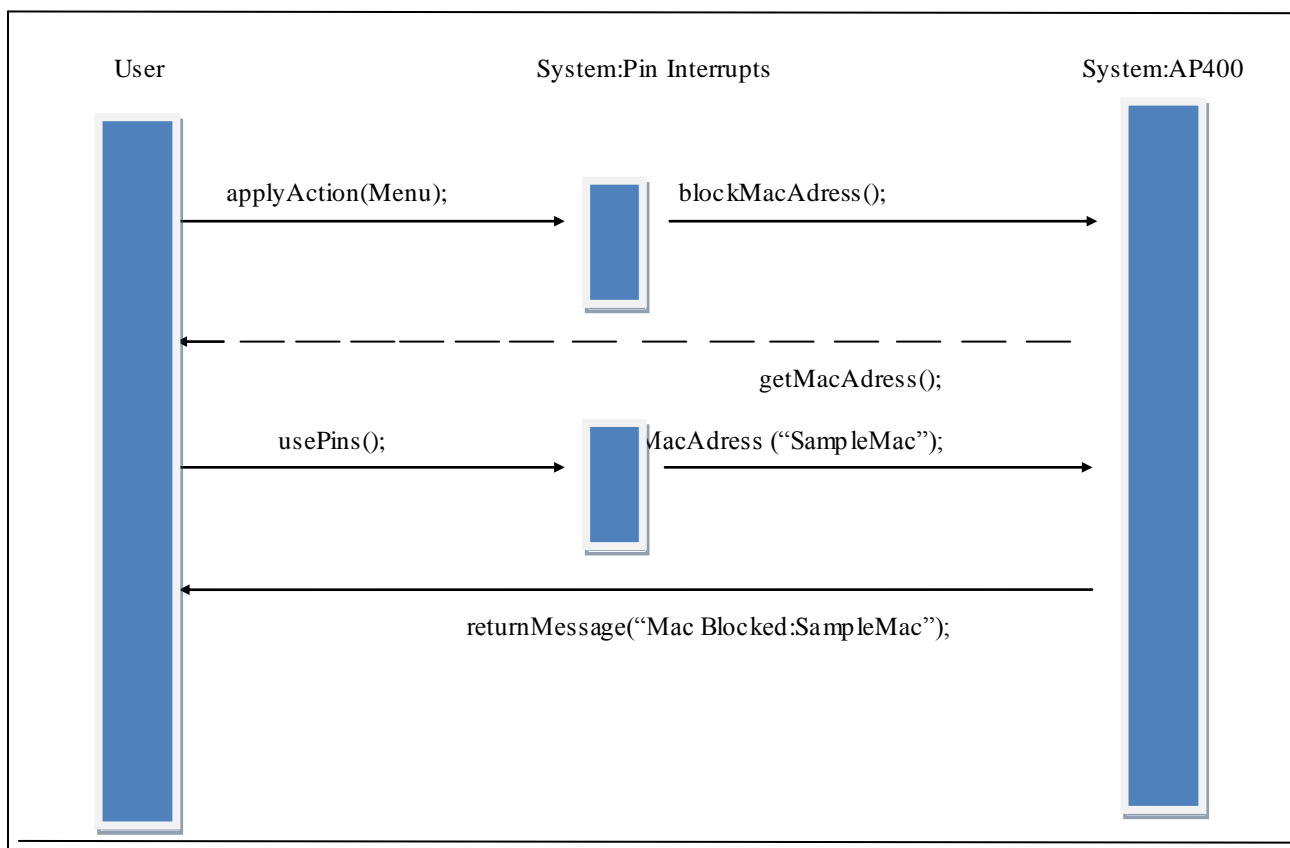
Change settings module consists of some sub-modules.

#### Change Settings: Change User Code Module



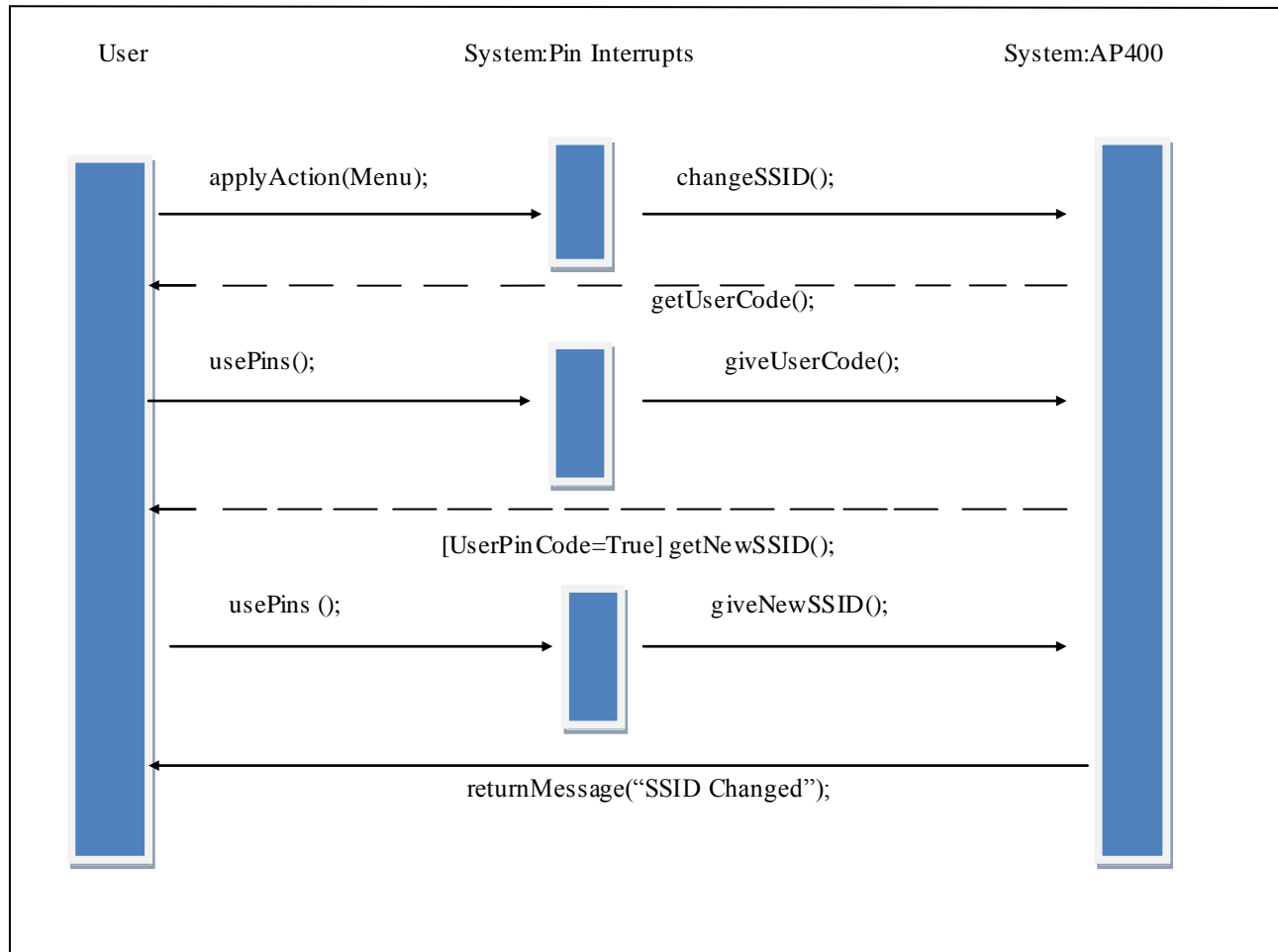
The above sequence diagram shows the interaction of the objects of our project when the user wants to change his pin code. Firstly, the user selects to change the pin code from the menu. When this action is selected, `changeUserCode()` function is called. Then, AP400 calls for the existing Pin code, in order to check it, because only the owner can change the pin code. After that, by using pins, the user gives an input of his existing pin code. Now, AP400 is ready to compare the input with the existing code. If this comparison returns true, a new input wanted. This input will be the new pin code. Again by using pins, the user gives his new pin code. Lastly, AP400 sets the new pin code and returns a message.

### Change Settings: Block Mac Adress Module



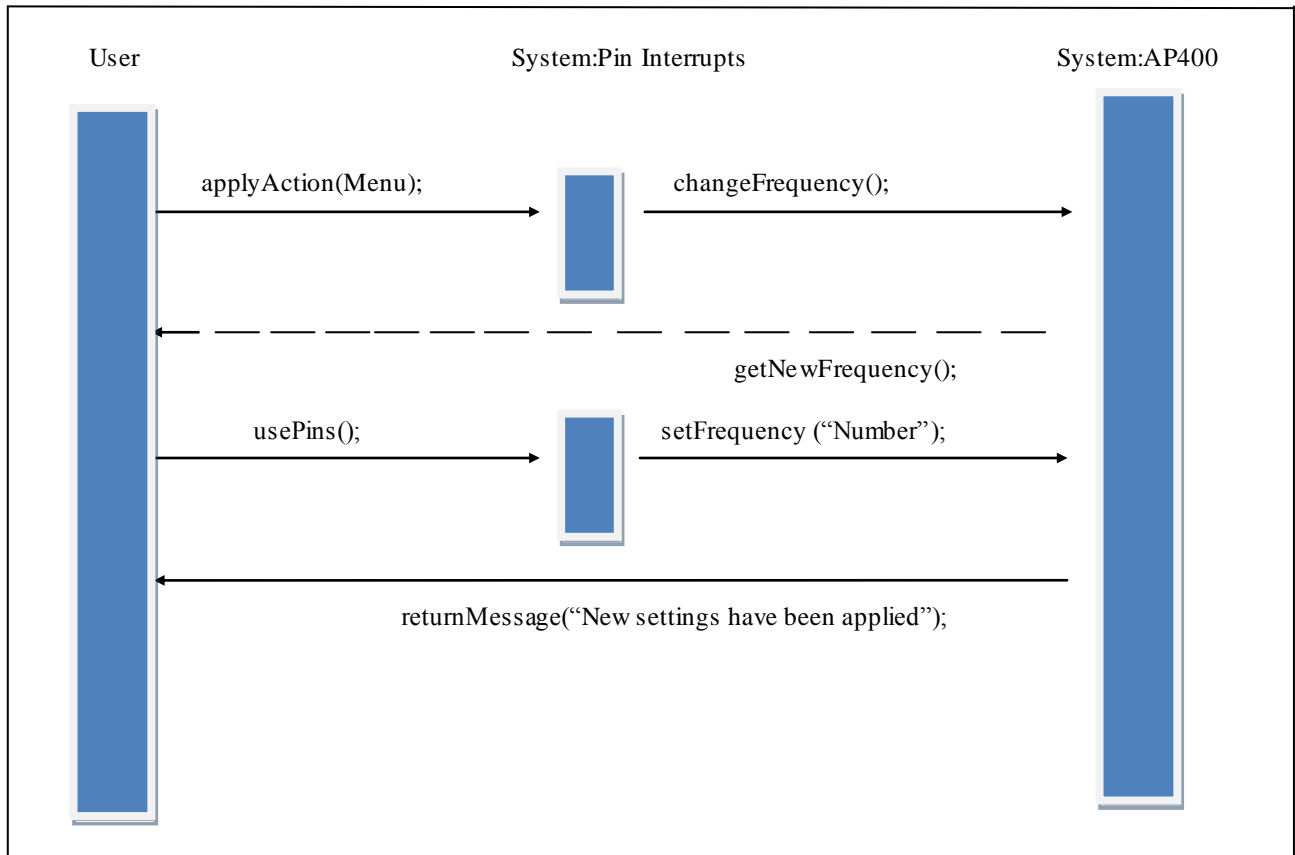
When the main user wants to block another user, he should block his mac adress. In order to do that, the user selects the action from the menu. When AP400 warned with the function, `blockMacAdress()`, the user should type the mac adress with the help of the pins. When the user finishes typing the mac adress, AP400 blocks this mac adress and returns a message.

## Change Settings: Change SSID Module



The above sequence diagram shows the sequential order of actions when the user wants to change the SSID. Firstly, the user selects to change the SSID code from the menu. When this action is selected, changeSSID() function is called. Then, AP400 calls for the user pin code, because only the owner can change the SSID. AP400 checks the user pin. If it is true, AP400 calls for the new SSID. By using pins, the user gives an input of wanted SSID. Then AP400 changes the SSID and returns a message.

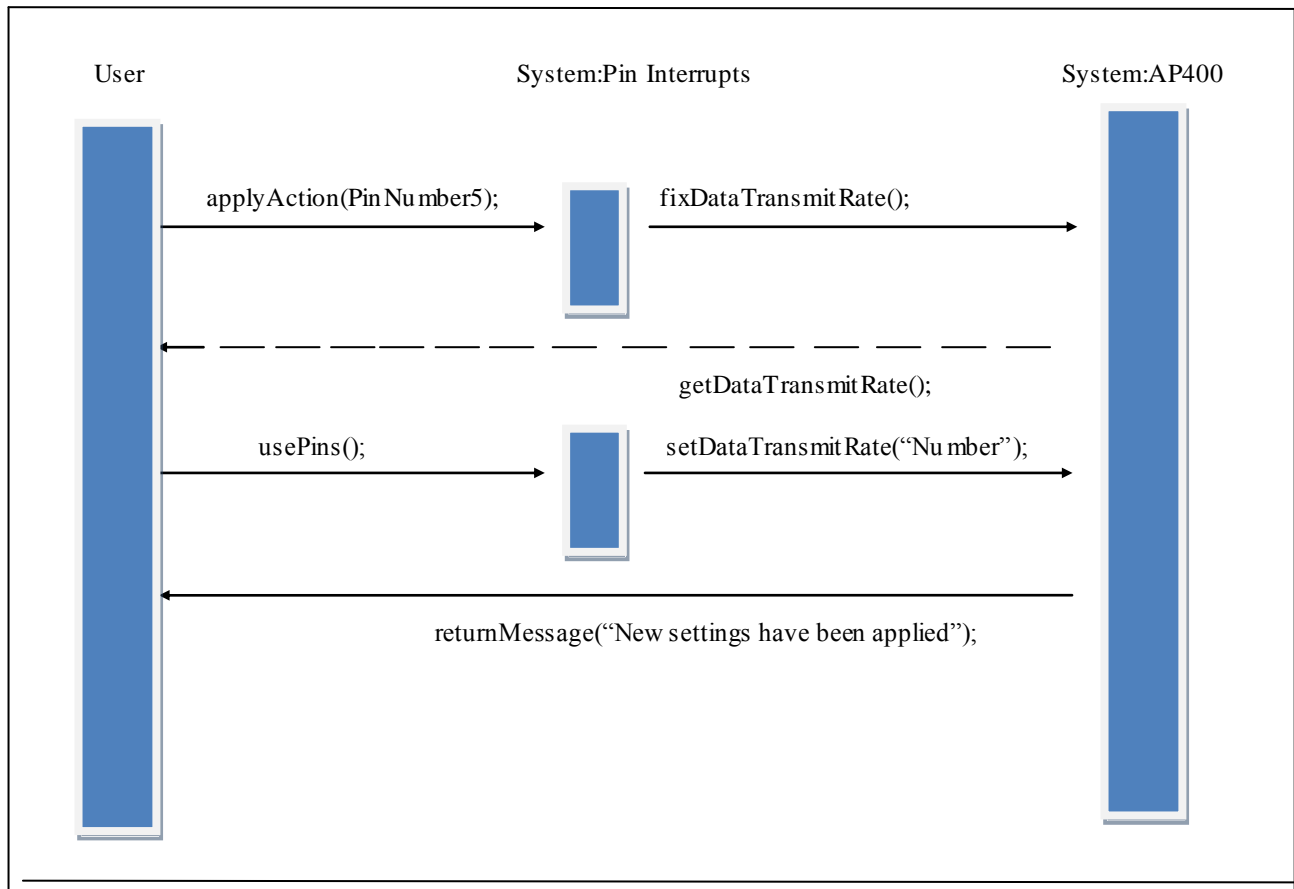
## Change Settings: Change Device Frequency Module



When the user wants to change frequency of the device, he will firstly choose this action from the menu. When the action is selected, `changeFrequency()` function is called. After that; AP400 asks for the new frequency value. By using pins, the user will give an input of new frequency value. Then AP400 sets the new frequency value, and return a message.



## Change Settings: Fix Data Transmit Rate Module

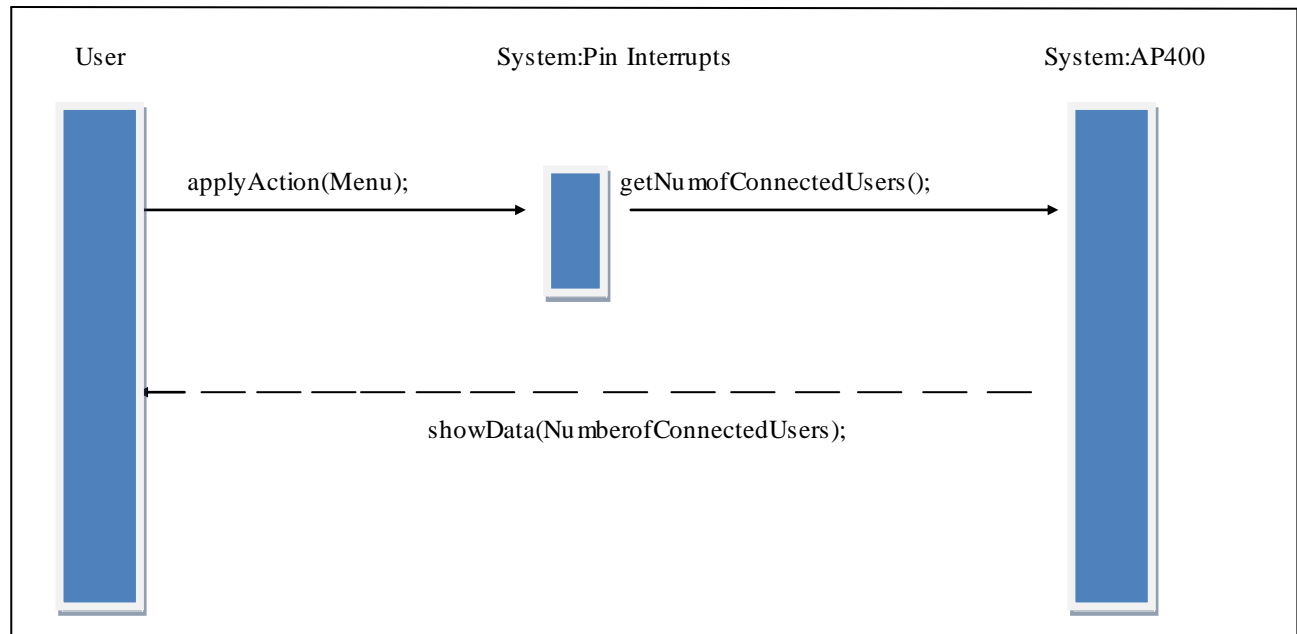


In order to fix the data transmit rate; the user should select the related action from the menu. Then, `fixDataTransmitRate()` function called. Then, AP400 returns with `getDataTransmitRate()` function. Now, it is expected from the user to enter the new data transmit rate value. If the user enters the new frequency value, AP400 returns a success message.

#### 4.2.2) Display Module

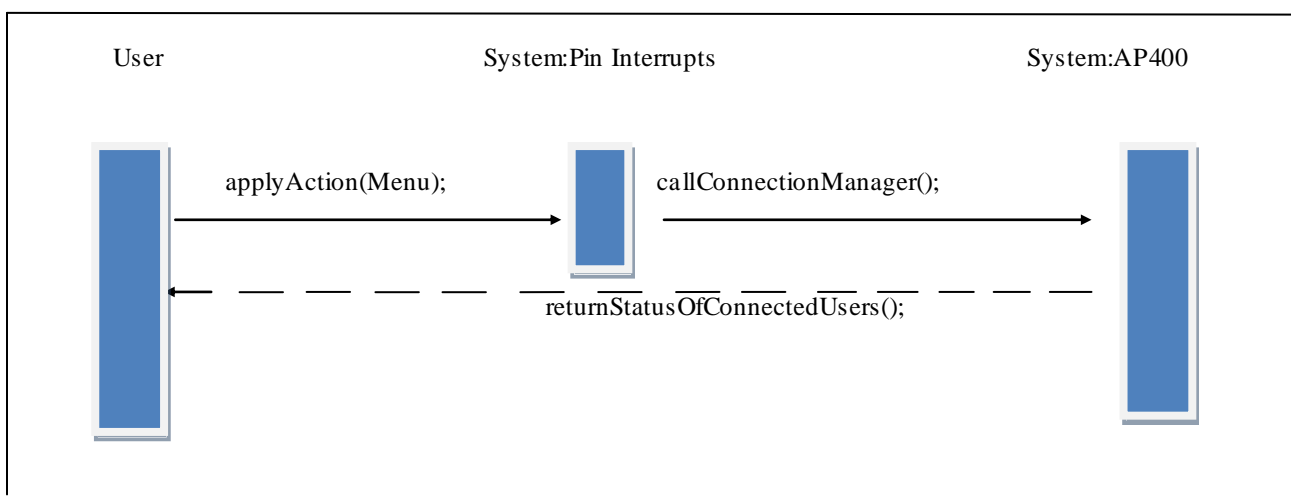
Display module consists of some sub-modules.

##### Display: Number of Connected Users Module



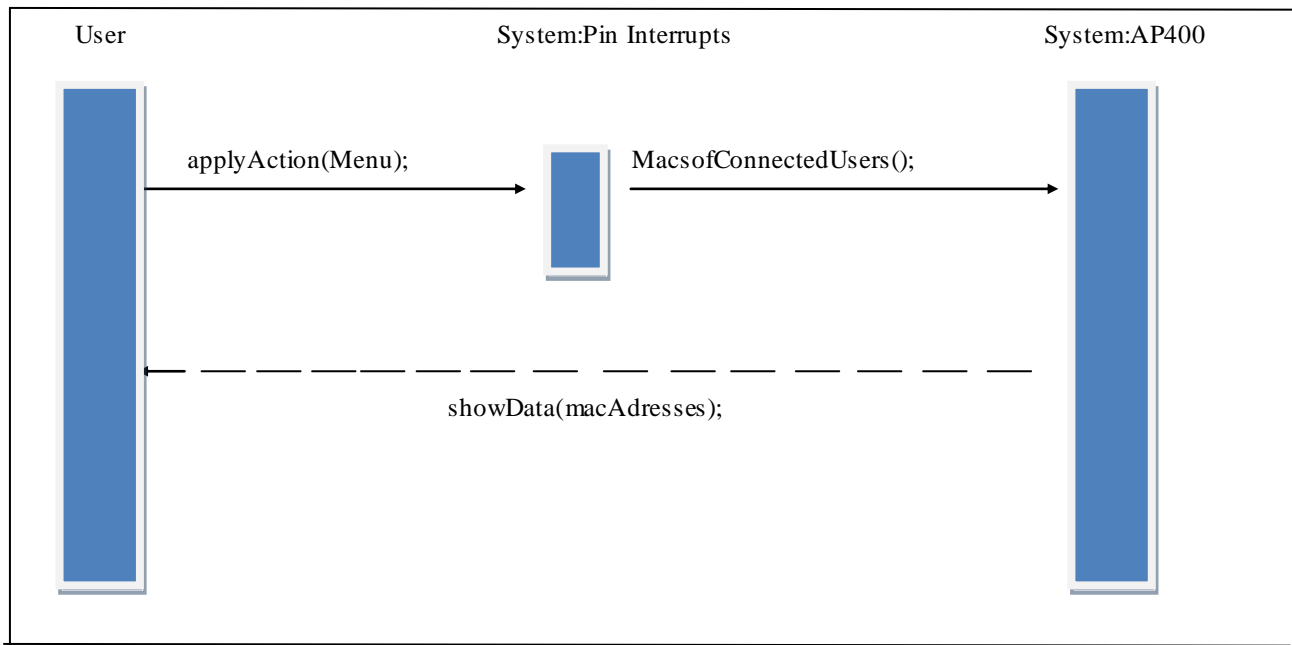
In order to see the number of connected users, the main user will set this action from the menu. Then, `getNumofConnectedUsers()` function is called. Then AP400 returns with the function `showData()` with the number of connected users.

##### Display: Connection Manager Module



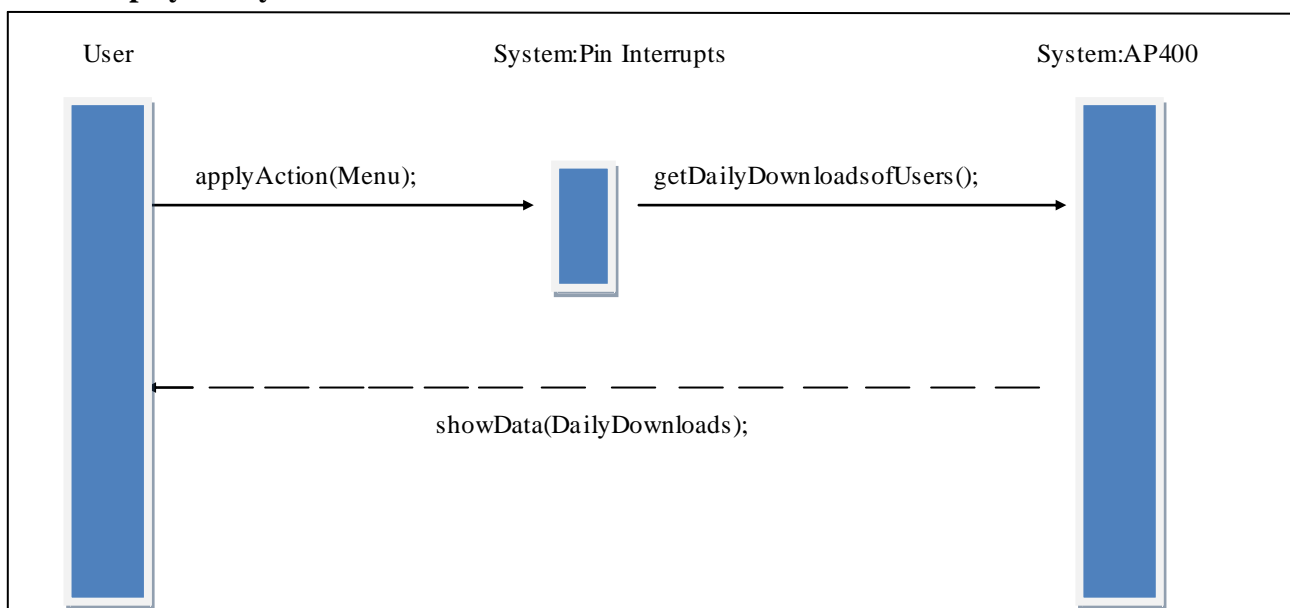
According to the diagram, when the user selects connection manager from the display menu, callConnectionManager() function is called. Then, AP400 returns a list of connected users.

### Display: Mac Adresses of Connected Users Module



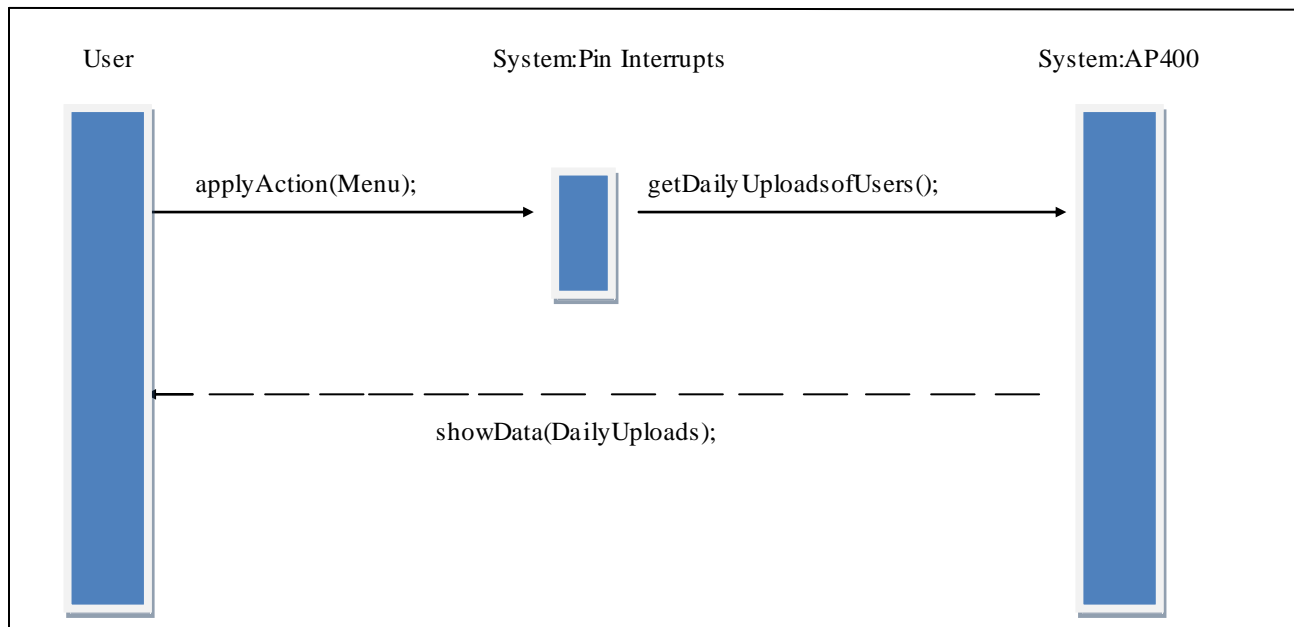
`getMacsofConnectedUsers()` function helps to get the mac adreeses of the connected users to AP400. Here, `showData()` function returns the mac addresses of the connected users.

### Display: Daily Downloads of Connected Users Module



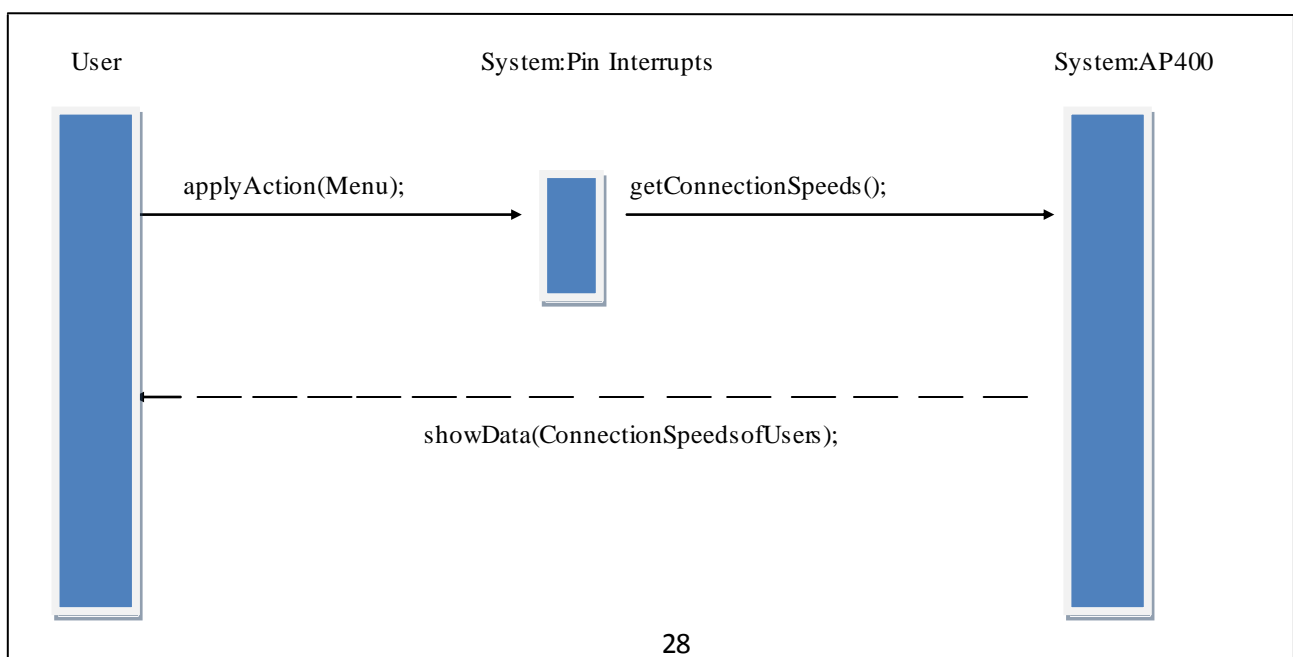
From the display menu; the user will select the action to see daily downloads of connected users to AP400. When AP400 is interrupted with the function `getDailyDownloadsofUsers()`, it will return the list of daily downloads.

### Display: Daily Uploads of Connected Users Module



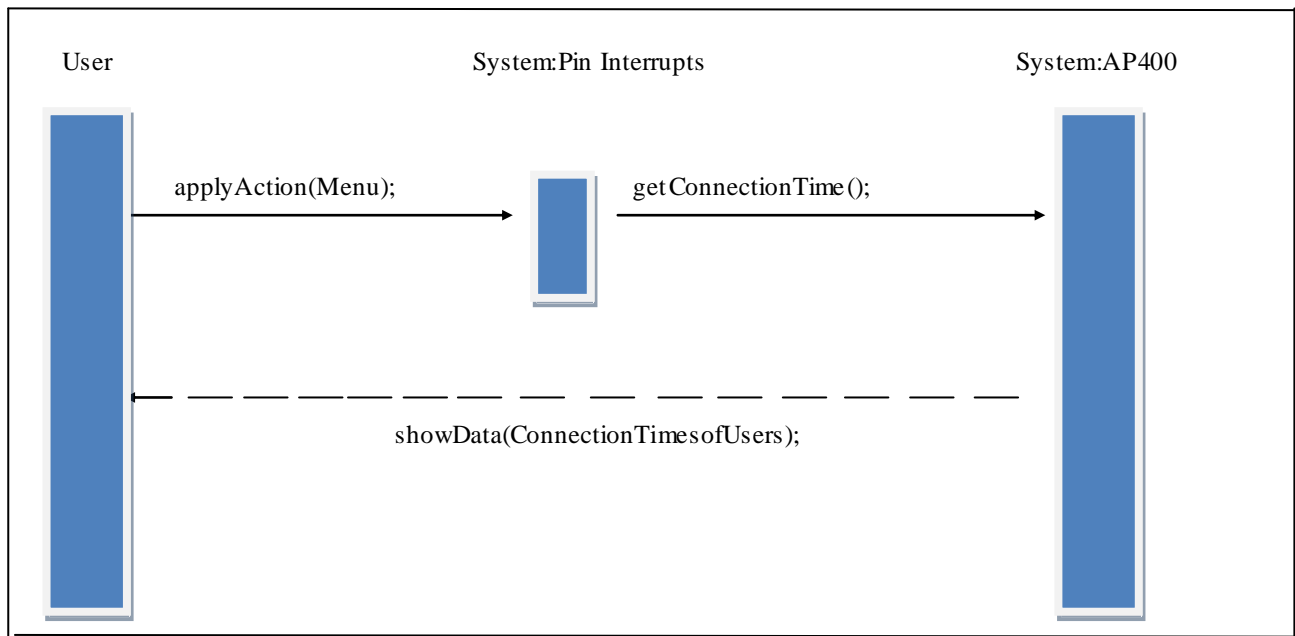
This module will work similiar to the previous module. Only wanted data is daily uploads of the users.

### Display: Connection Speeds of Connected Users Module



When the main user activate pins to see the connection speeds of the users, `getConnectionSpeeds()` function is called. Then, AP400 returns connection speed of users by the help of the `showData()` function.

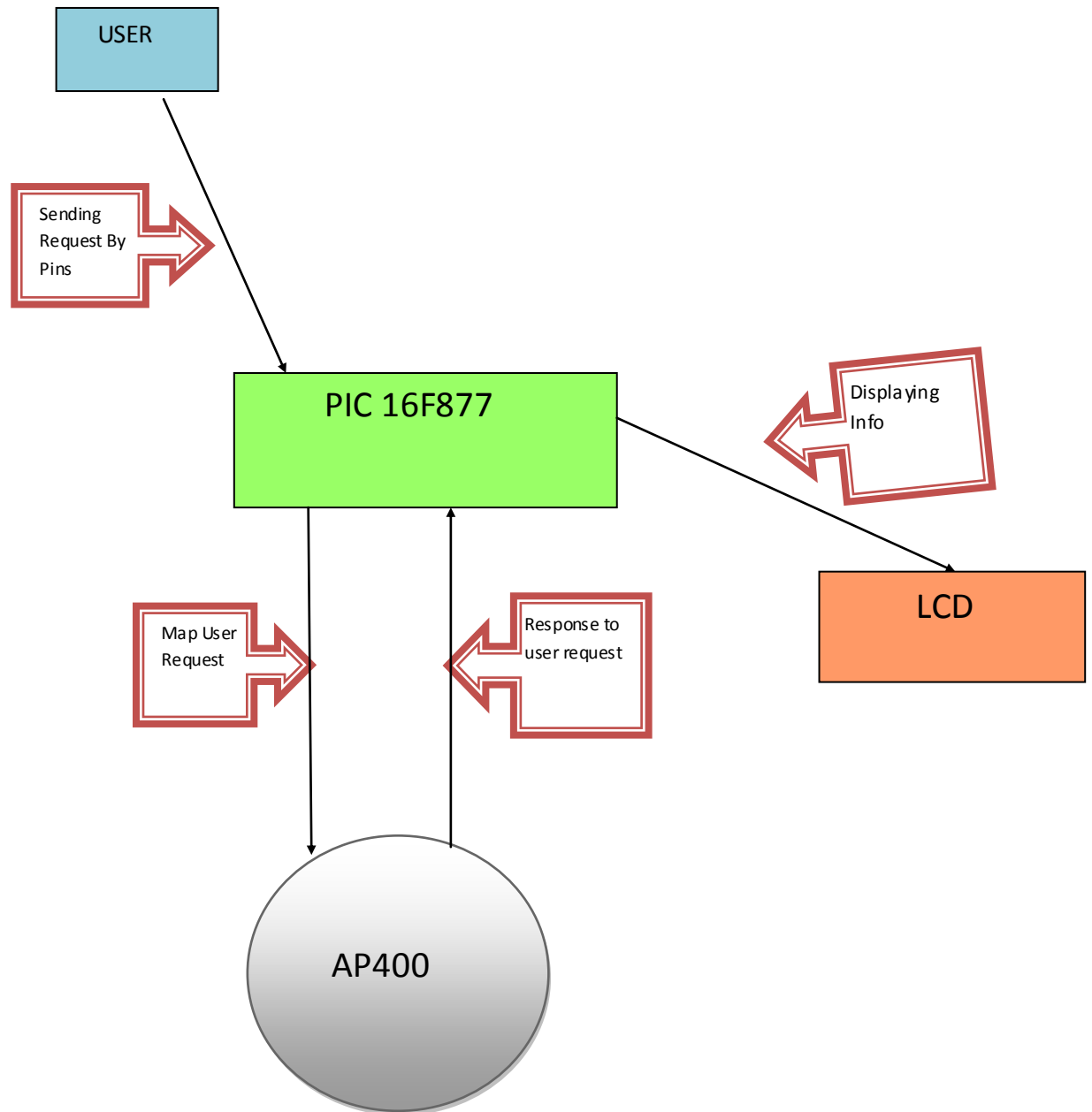
### **Display: Connection Times of Connected Users Module**



`getConnectionTime()` function will work for getting connection times of the connected users. When AP400 is interrupted by this function, it will return connection times of users by `showData()` function.

## 4.3) DATA FLOW DIAGRAMS

### 4.3.1) LEVEL 0



**Data Dictionary:**

**Name :** Sending requests by pins

**Output :** User

**Input :** Pic 16f877

**Description :** User send request to Pic 16f877 by pressing pins of the board . Its format is electrical signal .

**Name :** Map user request

**Output :** Pic 16f877

**Input :** AP400

**Description :** Pic takes user requests and then map them to AP400 . It is in RS232 serial port connection format.

**Name :** Response to user requests

**Output :** AP400

**Input :** Pic 16f877

**Description :** AP400 sends the results of user request to the pic 16f877 . It is in RS232 serial port connection .

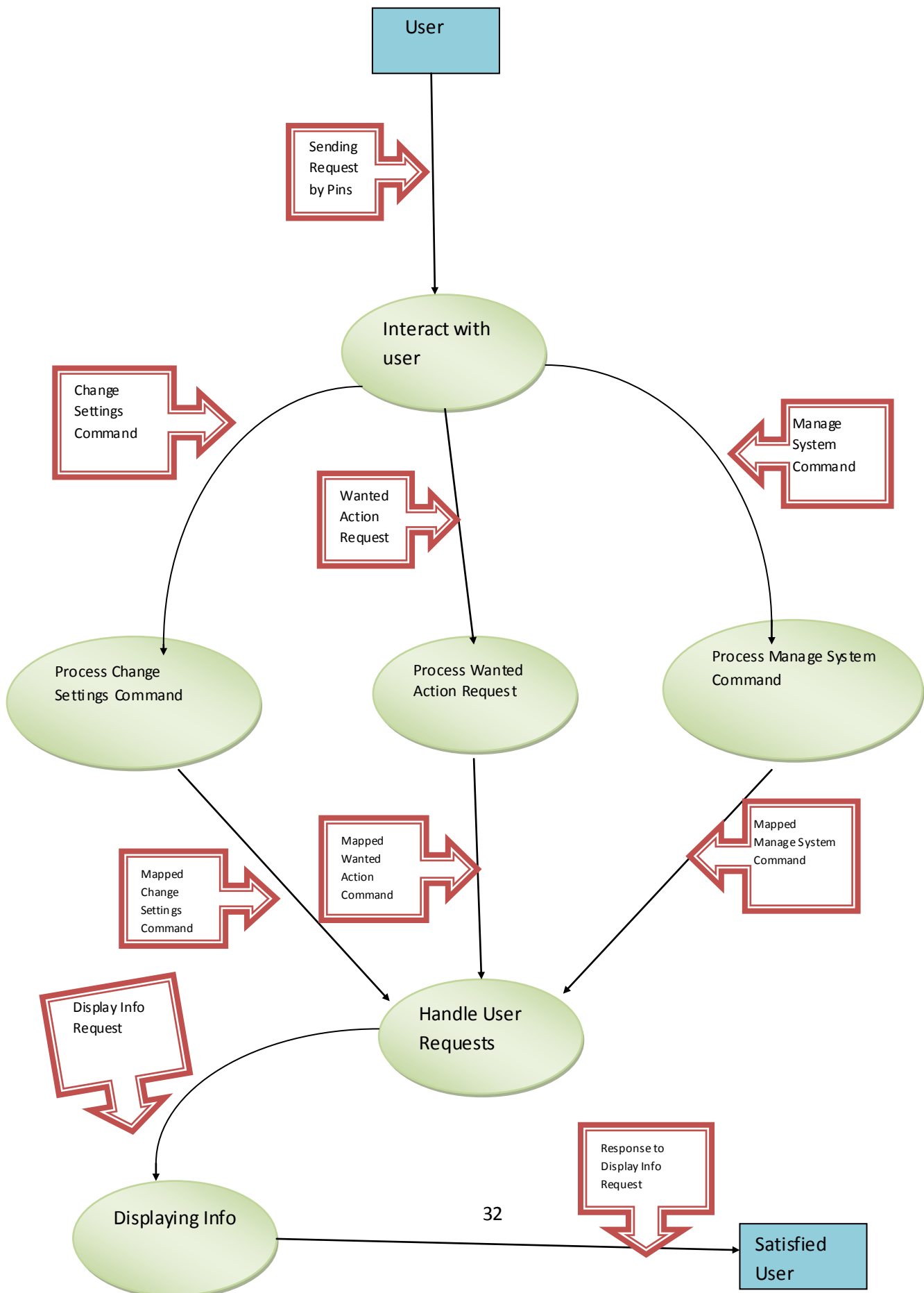
**Name :** Displaying Info

**Output :** Pic 16f877

**Input :** LCD

**Description :** Pic 16f877 sends the data which it receives from AP400 to the LCD display . Its format is electrical signals .

#### 4.3.2) LEVEL 1





**Data Dictionary:**

**Name :** Change settings command

**Output :** Interaction with user

**Input :** Process change settings command

**Description :** The output of the interaction will go to the “process change settings command” and this process will evaluate this info . Its format is electrical signal .

**Name :** Wanted action request

**Output :** Interaction with user

**Input :** Process wanted action request

**Description :** The output of the interaction will go to the “process wanted action request” and this process will evaluate this info . Its format is electrical signal .

**Name :** Manage system command

**Output :** Interaction with user

**Input :** Process manage system command

**Description :** The output of the interaction will go to the “process manage system command” and this process will evaluate this info . Its format is electrical signal .

**Name :** Mapped change settings command

**Output :** Process change setting command

**Input :** Handle user request

**Description :** The mapped request of the user will go to the “handle user request” and this process will adjust the settings as user wants . Its format is electrical signal .

**Name :** Mapped wanted action command

**Output :** Process wanted action request

**Input :** Handle user request

**Description :** The mapped request of the user will go to the “handle user request” and this process will send the info to the LCD display . Its format is electrical signal .

**Name :** Mapped manage system command

**Output :** Process manage system command

**Input :** Handle user request

**Description :** The mapped manage system command will go to the “handle user request” and this process will decide the state of the system . Its format is electrical signal .

**Name :** Display info request

**Output :** Handle user request

**Input :** Display info

**Description :** The info handled by “Handle user request” will be send to the “Display info”. Then this process will display the info on the LCD display . Its format is electrical signal .

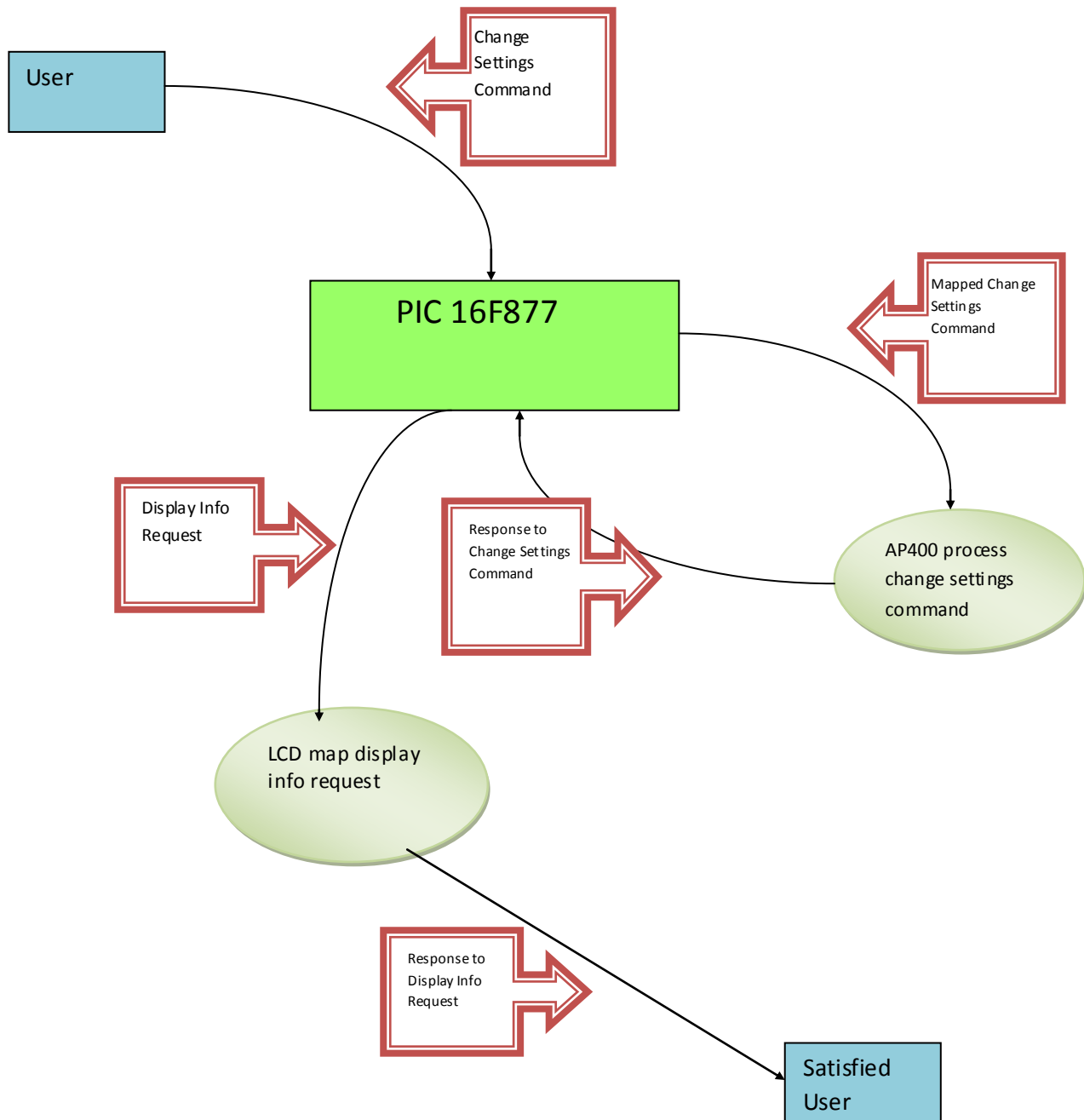
**Name :** Response to display info request

**Output :** Display info

**Input :** Satisfied user

**Description :** The info received by “Display info” will be displayed in LCD display .Its format is electrical signal .

### 4.3.3) LEVEL 2



## **Data Dictionary:**

**Name :** Change settings command

**Output :** User

**Input :** Pic 16f877

**Description :** User send request to Pic 16f877 by pressing pins of the board . Its format is electrical signal .

**Name :** Map change setting command

**Output :** Pic 16f877

**Input :** AP400 process change setting command

**Description :** Pic 16f877 will take the request of the user and then it will map to the “AP400 process change setting command” . Its format will be electrical signal .

**Name :** Response to change setting command

**Output :** AP400 process change setting command

**Input :** Pic 16f877

**Description :** AP400 will process on the request it receive from pic 16f877 and than send response to this request to the pic 16f877 . Its format is electrical signal .

**Name :** Display info request

**Output :** Pic 16f877

**Input :** LCD map display info request

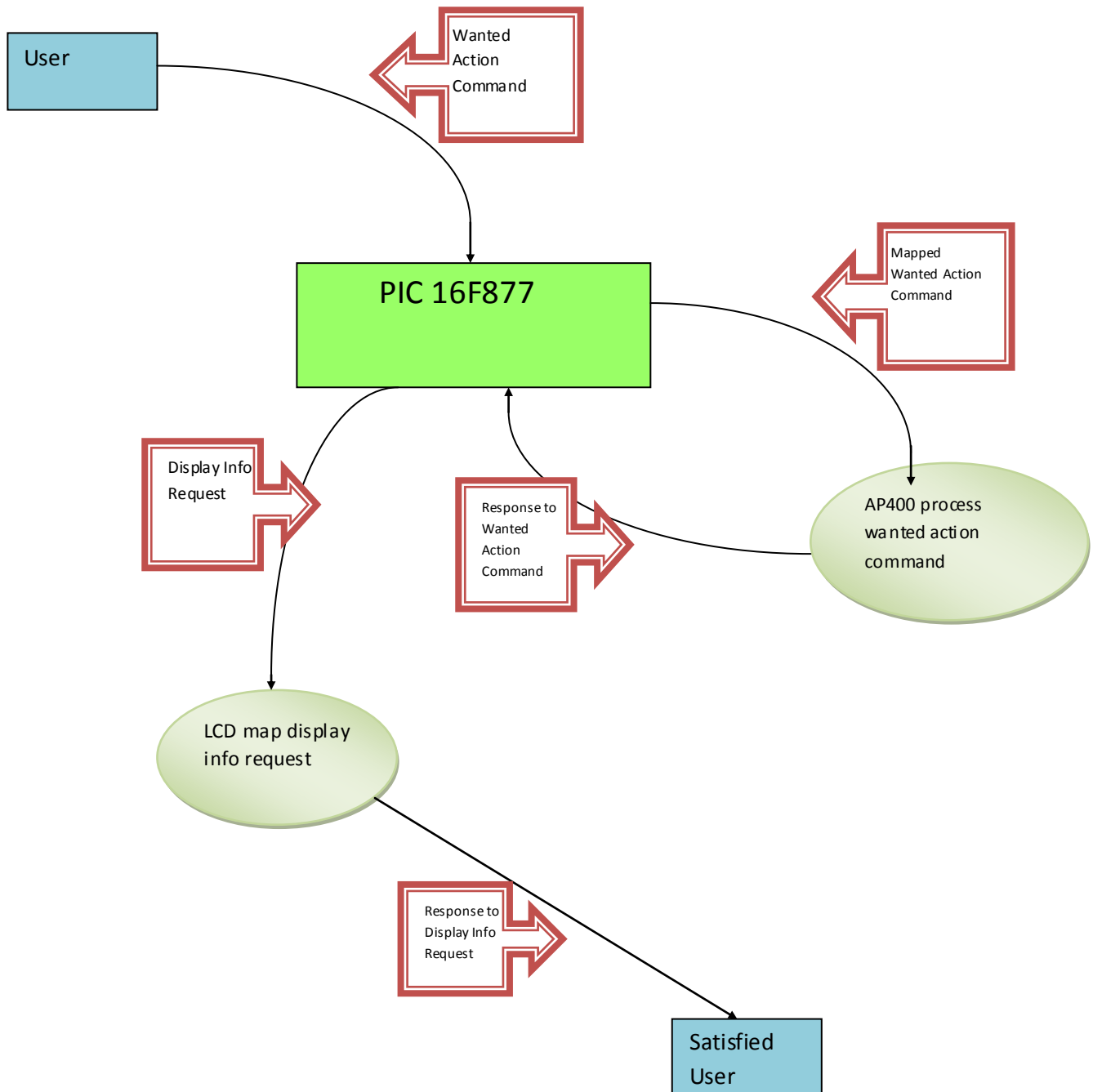
**Description :** The request received from “pic 16f877” will be processed by LCD display and it will be ready to seen on screen .

**Name :** Response to display info request

**Output :** LCD map display info request

**Input :** Satisfied user

**Description :** The info received by “Display info” will be displayed in LCD display .Its format is electrical signal .



## **Data Dictionary:**

**Name :** Wanted action request

**Output :** User

**Input :** Pic 16f877

**Description :** User send request of what he or she wants to the Pic 16f877 by pressing pins of the board . Its format is electrical signal .

**Name :** Map wanted action command

**Output :** Pic 16f877

**Input :** AP400 process wanted action command

**Description :** Pic 16f877 will take the request of the user and then it will map to the “AP400 process wanted action command” . Its format will be electrical signal .

**Name :** Response to wanted action command

**Output :** AP400 process wanted action command

**Input :** Pic 16f877

**Description :** AP400 will process on the request it receive from pic 16f877 and than send response to this request to the pic 16f877 . Its format is electrical signal .

**Name :** Display info request

**Output :** Pic 16f877

**Input :** LCD map display info request

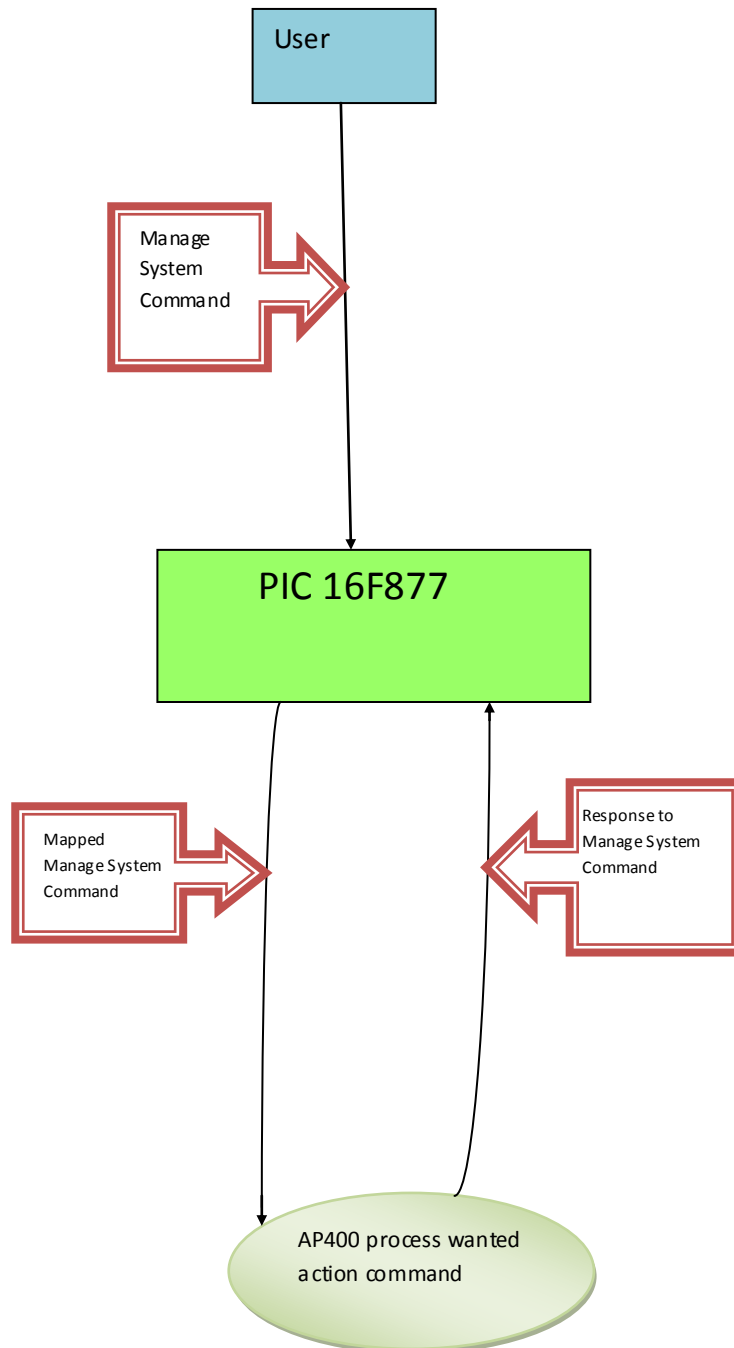
**Description :** The request received from “pic 16f877” will be processed by LCD display and it will be ready to seen on screen .

**Name :** Response to display info request

**Output :** LCD map display info request

**Input :** Satisfied user

**Description :** The info received by “Display info” will be displayed in LCD display .Its format is electrical signal .



## Data Dictionary:

**Name :** Manage system command

**Output :** User

**Input :** Pic 16f877

**Description :** User send request of what he or she wants to the Pic 16f877 by pressing pins of the board . Its format is electrical signal .

**Name :** Map change system command

**Output :** Pic 16f877

**Input :** AP400 process manage system command

**Description :** Pic 16f877 will take the request of the user and then it will map to the “AP400 process wanted action command” . Its format will be electrical signal .

**Name :** Response to manage system command

**Output :** AP400 process manage system command

**Input :** Pic 16f877

**Description :** AP400 will process on the request it receive from pic 16f877 and than send response to this request to the pic 16f877 . Its format is electrical signal .

## 5)USER INTERFACE

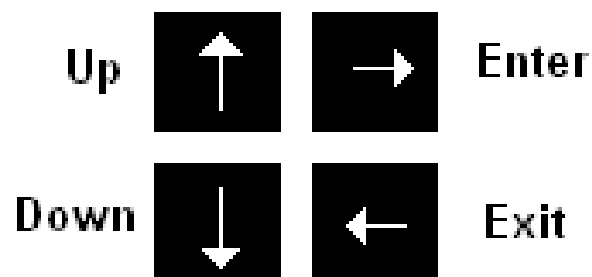


Figure 5.1



There will be 4 buttons on the board. Two of them will be used for going up and down in menus. The other two buttons will be used to for entering to a menu and exiting from a menu.



Figure 5.2

When the user has not pressed any button yet, the time and the date is going to be displayed as in figure 5.2 until the user presses any button on the board.

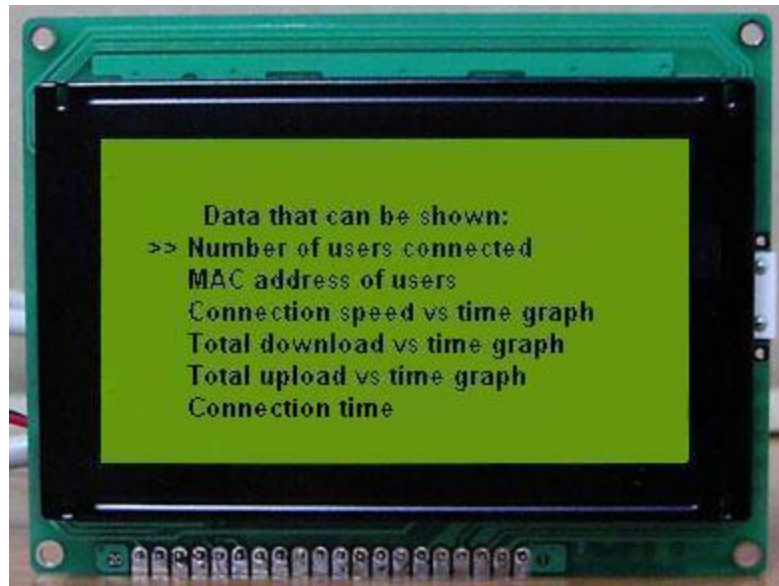


Figure 5.3

When the user presses any button on the board the screen is going to display the list of data that can be shown as in figure 5.3. There are some other actions such as change PIN code and SSID. These items will be shown in the second page when the user presses the down button.



Figure 5.4

When the user selects the number of users connected and presses the enter button, the number of clients who are currently using the device is going to appear on the LCD display with the name of the network name as in figure 5.4. When the user presses the exit button, it will return to the main menu.



Figure 5.5

When the user selects the show MAC address choice from the list and presses the enter button MAC address of the clients that are currently connected to the device is going to be displayed with the network name as in figure 5.5. The MAC address of the other clients is going to be displayed on the LCD display if the user presses the up/down button. When the user presses the exit button, it will return to the main menu.

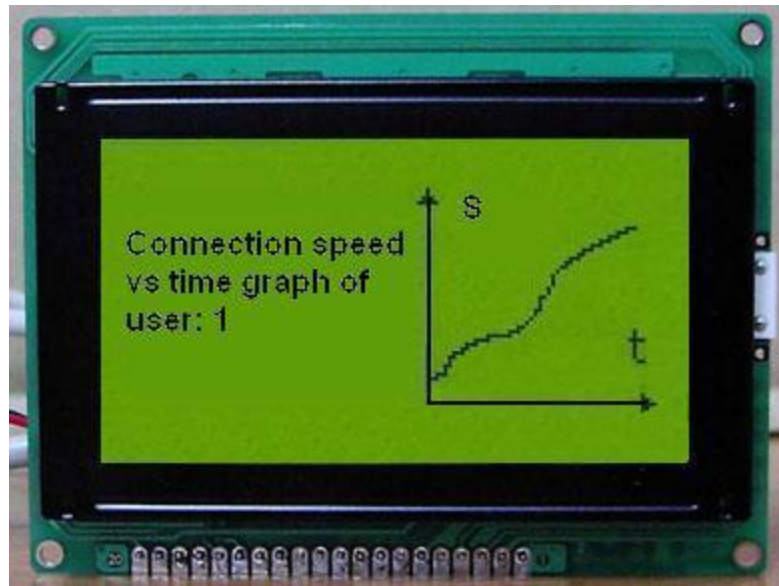


Figure 5.6

When the user selects the show connection speed choice from the list and presses the enter button connection speed vs. time graph of each client that is connected to the device is going to be displayed as in figure 5.6. The graph of the other clients is going to be displayed on the LCD display if the user presses the up/down button. When the user presses the exit button, it will return to the main menu.

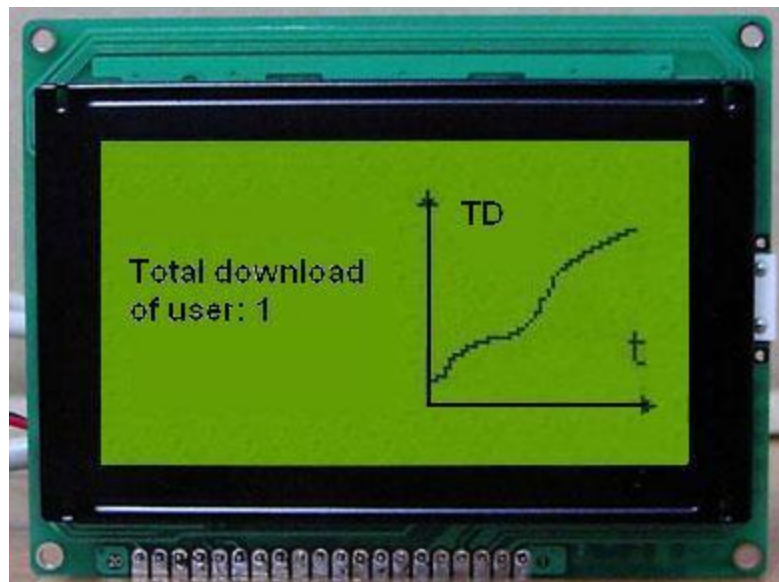


Figure 5.7

When the user selects the show total download of the users choice from the list and presses the enter button total download vs. time graph of each client that is connected to the device is going to be displayed as in figure 5.7. The user is able to see the graph of other clients by pressing the up/down button. When the user presses the exit button, it will return to the main menu.

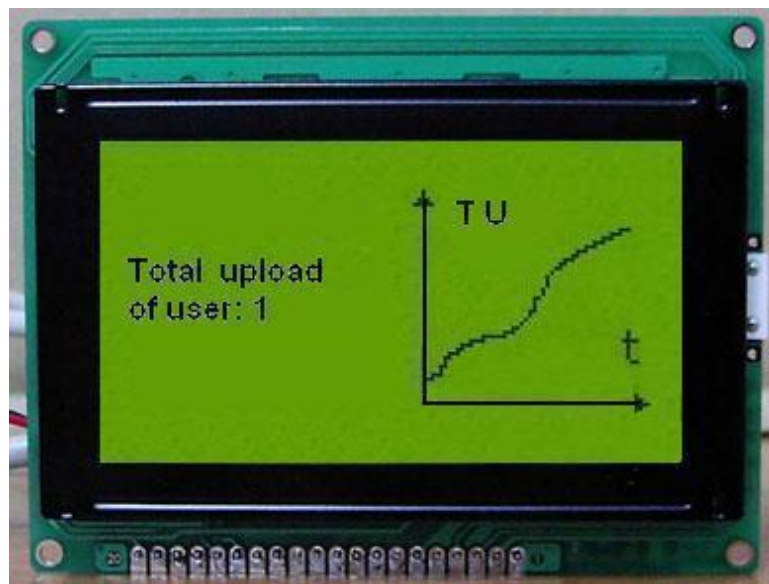


Figure 5.8

When the user selects the show total upload of the users choice from the list and presses the enter button total upload vs. time graph of each client that are connected to the device is going to be displayed as in figure 5.8. The user is able to see the graph of other clients by pressing the up/down button. The user can also return to the main menu by pressing the exit button.



Figure 5.9

When the user selects the show connection time of the users choice from the list and presses the enter button the connection time of each client that are connected to the device is going to be displayed as in figure 5.9. The connection time of the other clients can be displayed on the LCD display by up/down button. The user can return to the main menu by pressing the exit button.

## 6) GANTT CHART

Untitled1 : Gantt Chart : Project

