Technical document

Note: I generally don't write pseudocode code, I prefer to describe the problem and the solution in plain English. I wrote pseudocode code for this assignment as it was required but it may not be the best

Components:

The project is broken down into 3 main components with further sub components

1. Desktop program
    a. PDF reading and display
    b. Arduino communication
    c. GUI layer
2. Receiver program
    a. Receiving and processing information
    b. Sending information to computer
    c. Sending information to other Arduino devices
3. Device program
    a. GPS communicator and translator
    b. Information passing protocols
    c. Power consumption manager

External libraries

The RXTX library for serial communication with the Arduino. It is an open source library with a standard MIT license.

The PDF Reader Library used for reading PDF files and displaying those PDFs. MIT license

Break down of components

<u>Desktop Program</u>

Reading and display:

```
Reading and Display:

Load pdf()
{
    if pdf is not there
    {
        throw error
    }
    else
    {
        Ask for GPS coordinates
        run PDF Reader load function and get map from coordinates
        if we cant find that map
        {
            WELL CRAP. Sorry but cant load map. Load everything else
        }
        run PDF Reader display to Java element, or to show placeholder if no map
    }

    Run first set up
    prompt user for information about groups
    keep asking until they say okay

    Load and display message area
}
```

Arduino Communication

```
Arduino Communication
{
	Establish connection with the reciever
	if no connection
	{
		warn user
	}

	Send test signal to arduino to send to devices
	wait for comfirmation from devices
	if comfirmation timeout
	{
		list device timed out
	}

	go to standby and wait for information from receiver

	When we get information from receiver process it
	Make sure the message is a proper one
	if cant decipher message
	{
		if we at least got the ID of the device who sent it
		{
			warn user whos device failed to send info properly
		}
		else
		{
			just tell user got message but couldnt decipher
		}
	}
}
```

GUI layer

```
GUI layer
{
	After pdf loader, lets now keep track of colors
	Assign teams colors based on order they were checked in
	Load up buttons and side bar

	When displaying new message, highlight it

	Handle all warnings that should be given to the user
	warning function
	{
		HEY USER SOMETHINGS WRONG
	}

	Popup layer for when the user is typing in names
	{

	}

	popup layer for when the user is sending a message
	{
		if failed
		{
			warn user
		}
		else
		{
			tell them it worked
		}
	}
}
```

Receiver

Receive and process info

```
Receive and process info
{
	We got a message
	if message has no ID
	{
		guess its not for us, so ignore
	}

	if message has no end code
	{
		We didnt get the whole message. Lets try to make sense of it
		Can we get GPS data?
		Can we get special text data?

		If no to any of these, warn the user where the problem is

		We prob cant decipher it, so maybe send to user to see if they can
	}
}
```

Send info to computer

```
Send info to computer
{
	Using RXTX send this string along with if we found errors to the computer
	if failed to send
	{
		Save that information and wait for a reconnect
		if out internal storage is full (which happens if we keep getting
		messages but are still not connected)
		{
			Dont save the message a pray the user connects it back soon
		}
	}
}
```

Send info to other Arduino devices

```
Send info to other arduino devices
{
	Keep looking for a message from the computer
	if code to send out message
	{
		Take the information and send it out
	}

	if we got a comfirmation message
	{
		tell user we got the message
	}

	if message comfirmation timed out
	{
		Tell user that we didnt get a message from these devices
	}

}
```

Device Program

GPS communicator and translator

```
GPS communiactor and translator
{
    if getting info from GPS
    {
        if information is a position
        {
            Get position information and translate it
            Save that inforamtion and get ready to send out or to
            display on the LCD screen
        }
        else
        {
            we dont have a satalite fix yet, just wait
        }
    }

    if told to go to low power mode
    {
        Send out code for enter low power mode
    }

    if told to stop processing GPS info
    {
        Go to sleep and only wake when power controller tells you to
    }
}
```

Information passing protocol

```
information passing protocol
{
    if we received a message
    {
        if it has the code for the command base receiver
        {
            read message
        }
        else
        {
            probably another device
            if code to pass info along
            {
                if already got it
                {
                    ignore
                }
                try to establish connection to receiver
                if failed to get comfirmation message
                {
                    send out to another device that hasnt gotten it
                }
            }
        }

    }

    if we are sending a message
    {
        Send that information to the radio module and send it out
        if failed to get confirmation message
        {
            inform user that there was a problm and they should try again
        }
    }
}
```

Power consumption manager

```
Power consumption manager
{
    (Work in progress, still not sure about all aspects)
    Lets keep track of the number of times we are receiving and sending data
    Are we getting a lot of traffic?
        is it GPS traffic or message Traffic?
    if GPs traffic
    {
        Set GPS to high power mode
        Low screen power, keep it off most times
        Be redy to send out messages more
        Set low estimate on power consumption
    }

    if its message data
    {
        set GPS to low power mode
        Keep screen on, but contrast low
        Set moderate estimates on power consumption
    }
```

```
How often are we using certain devices?
if GPS readings are above a threshhold
{
    keep it in high power mode
}

if after time it goes below threshhold
{
    Set to low power mode
}

if consistatnly using GPS every so many minutes
{
    Set a timer and turn GPS off. Turn on 5 minutes before possible check in
    then get reading
    if not on during check in
    {
        tell user its getting reading
        wait till GPS gets satelites
        once it has reading send out
    }
}

if using screen a lot
{
    if during night
    {
        Keep the screen on but keep contrast low
    }
    else during day
    {
        Keep screen on and contrast high
    }
}
```

```
if screen use below threshhold
{
    lets lower screen contrast if we can
}

if gettings lots of signals from others to pass on
{
    if its the same device multiple times
    {
        if you get his code just immedietly relay, dont think
    }
}

if not sending out a lot of signals for what ever reason
{
    Only check for signals every few seconds
}
```