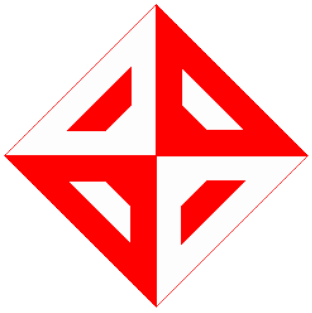


Software Requirements Specification

(IEEE Std 830-1998)[1]

V1.0

NoNET

Prepared by FixIT

Ceyda Tosun-1819580
Gülşah Sabırsız-1881424
Gulnaz Shaidolda-1784578

METU - Department of Computer Engineering
CENG 491 Senior Design Project I
Fall 2015-2016

TABLE OF CONTENTS

1. Introduction

This document is a software requirement specification for Post Emergency Communication System named as NoNET. In this document, firstly we are going to define the problem and introduce purpose and the scope of this document. Secondly, we are going to give an overall description. Then, we are going to state specific requirements. Lastly, we are going to give data model and description.

1.1 Problem Definition

In emergency situations such as earthquake, communication plays a significant role. For instance, people may need to communicate with others to appeal for help. Or they may want to be informed about the situation of their relatives. However, in such situations it is very likely to collapse infrastructure of the Internet and GSM because network encounter trouble if there is a great amount of mobile phone calls or the Internet data usage.

This project will deal with this problem and it will allow communicating with others without using the Internet connection and GSM. As a result of this, after an emergency situation people will be able to continue communicating.

1.2 Purpose

This software requirement specification document is defining the detailed description of the architecture, specifications and functionalities the project NoNET. This document is prepared for both the development team and the users. The purpose of the project is developing an post emergency communication system. Target audience of the NoNET is personnels of the AFAD and every people who may face with an emergency situation and need to take precaution.

1.3 Scope

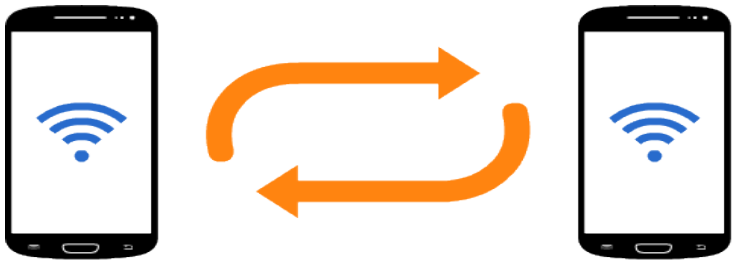
The final product is a mobile application that serves a new communication system especially for emergency situations. The application will be able to set a connection between users without the Internet connection and GSM. Users of the application will be able to send messages to:

- A certain nearby peer,
- Many peers in the communication range,
- A certain distant peer,
- Many peers out of communication range.

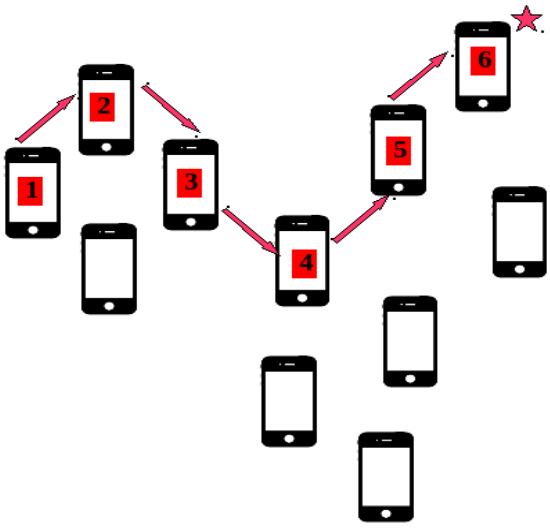
The project will be done by 3 people in 2 semesters.

1.4 System Overview

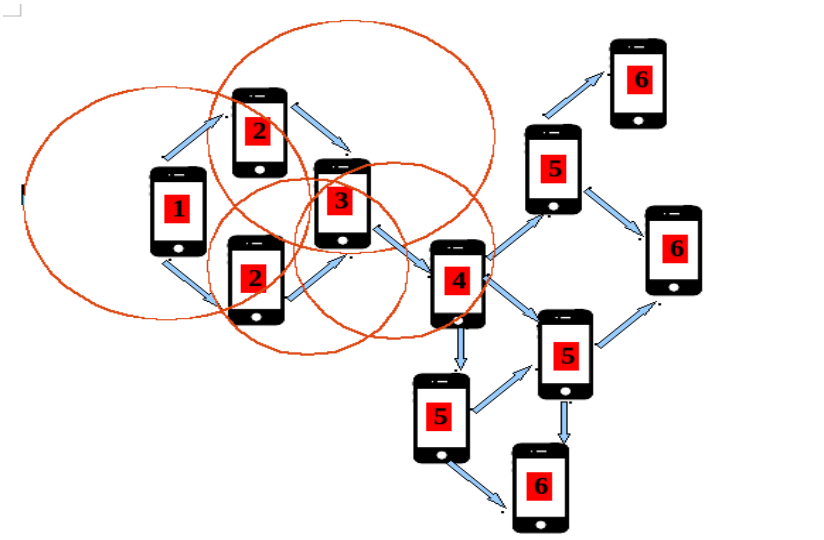
A Post Emergency Communication System - NoNET is a messaging platform for Android and IOS devices which allows users to communicate without using the Internet and GSM infrastructure. In order to explain in more detail, 4 main scenarios are shown below.



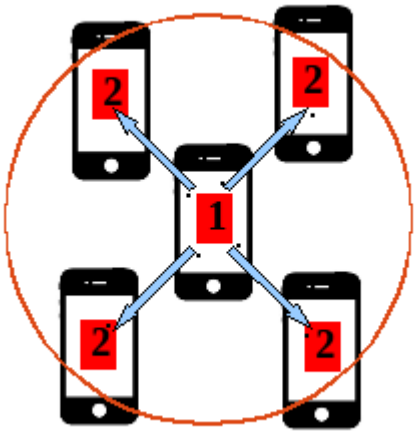
Scenario 1 : Users will be able to connect a certain nearby peer and communicate with the peer.



Scenario 2 : Users of the application will be able to send messages to a certain distant peer. Middle peers will also take the messages and in the background they will forward it to other peers but middle users will not see this messages.



Scenario 3 : Users of the application will be able to broadcast messages to distant peers. Everybody in the Ad Hoc Network will be able to see the messages.



Scenario 4 : Users of the application will be able to broadcast messages to peers in the communication range.

1.5 Definitions, acronyms, and abbreviations

Term	Definition
IEEE	The Institute of Electrical and Electronics Engineers
GSM	Global System for Mobile Communications (Groupe Spécial Mobile)
AFAD	The Disaster and Emergency Management Authority
NoNET	Name of the application which is explained in this document
SRS	Software Requirement Specification
Java	Java is a general-purpose computer_programming_language that is concurrent, class based, object oriented, and specifically designed to have as few implementation dependencies as possible.
C	C is a high-level and general purpose programming language that is ideal for developing firmware or portable applications.
Android	Android is the name of the mobile operating system made by American company; Google.
IOS	IOS is a mobile operating system created and developed by Apple Inc. and distributed exclusively for Apple_hardware.
Wi-Fi Direct	Wi-Fi Direct® is a certification mark for devices supporting a technology that enables Wi-Fi devices to connect directly, making it simple and convenient to do things like print, share, sync and display.
Multi-peer connectivity	The Multi-peer Connectivity framework provides support for discovering services provided by nearby IOS devices using infrastructure Wi-Fi networks, peer-to-peer Wi-Fi, and Bluetooth personal area networks and subsequently communicating with those services by sending message-based data, streaming data, and resources.
Scrum	Scrum is an agile software development model based on multiple small teams working in an intensive and interdependent manner.

1.6 Assumptions and dependencies

- **Operating System:** The application will be developed for IOS and Android platforms.
  - IOS 7.0 and upper versions, (Multi-peer Connectivity have to be supported.)
  - Android 4.0.3 and upper versions. (Wi-Fi Direct should be supported.)
    - According IDC [2], Android dominated the smart phone market with a share of 82.8% and IOS is in 13.9%. Therefore covering IOS and Android devices will be result in covering 95% of the smart phone market.

- **Communication range:** For near field messaging scenarios, communication range is about 20-30 meters.(It is restricted by Wi-Fi Direct and Multi-peer Connectivity infrastructure range). However, with the help of the hop to hop network, we will be able to send ordinary users' broadcast messages to 15th hop users. Moreover, we will broadcast AFAD personnels' messages to 30th hop users.

- **Battery:** In emergency situations, energy is also becoming a very important issue. Most probably our application will drain battery very fast. Therefore, receiver with battery under 30% will only accept AFAD messages and messages comes into one's own.

- **Bandwidth:** Bandwidth has a fixed size because in emergency situations people do not need to send too long messages. And content of the message is restricted as text message. Time information of the messages will also be sent. Moreover, users may choose to send their location information. It will be optional.

These assumptions may change during the implementation and new features may be added. SRS document will be updated according to these changes.

2. Overall description

This part of the SRS is about general factors that affect the system and requirements of the system.

2.1 Product functions

Register

After the user downloaded our application, s/he must register to the system in order to use it. There will be register button in the login page which redirects user to register page.. After the user registered to the system, s/he can login and use the system. e. After filling required fields with information, s/he can register to the system with using register button.

Send message to a near peer

User sends a short message to ask help to a person near to him/her. As the person is close to our user , he can come to help faster than anyone else.

Broadcast message

User broadcasts message to the surrounding people to ask help in emergency situations such as an earthquake. In such a case reaching more people is important since it increases the chance of getting rescued.

AFAD broadcasts message to people in communication range to learn if anyone needs help and to learn the location of people who is under destructed buildings.

Send message to a distant user

User sends message to a specific user after s/he discovered the other user devices. He send message to distant user AFAD to ask help or her/his family member to inform the about his situation or to receive information from them.

Send location

User of our application sends the location of him attached to the message to the surrounding people. There will be location button. If user clicks on this button, location of him will be sent.

Discovering devices

User discovers the near devices and a distant specific devices if s/he wants to communicate them. When application is started, it begins to discover devices without clicking any button.

Connect devices

User connects to the near devices or distant devices after s/he discovered the certain device. S/he can connect with other people by clicking to the other devices' name.

Write Message

User types a message by clicking on the text field. The message includes time of sent message. Message will be short and understandable since it is an emergency situation. Message content can include asking help or information about the user's situation.

Accept Message

User accepts the message automatically in the background of the application. If the user's device's battery is low, it will not attend to the ad-hoc network.

Reject to Message

User can reject to transferring the received messages to other users. Because the user may be under destructed buildings, waiting for help and his device's charge may be low.

Offer a Service

User have to offer a service to be discovered by surrounding devices.

Select device

User select the device before sending message to a certain person by clicking to the device name. After that the device will selected and user can communicate the selected user.

2.1.1 Use-case model survey

Ordinary person: Use case diagram

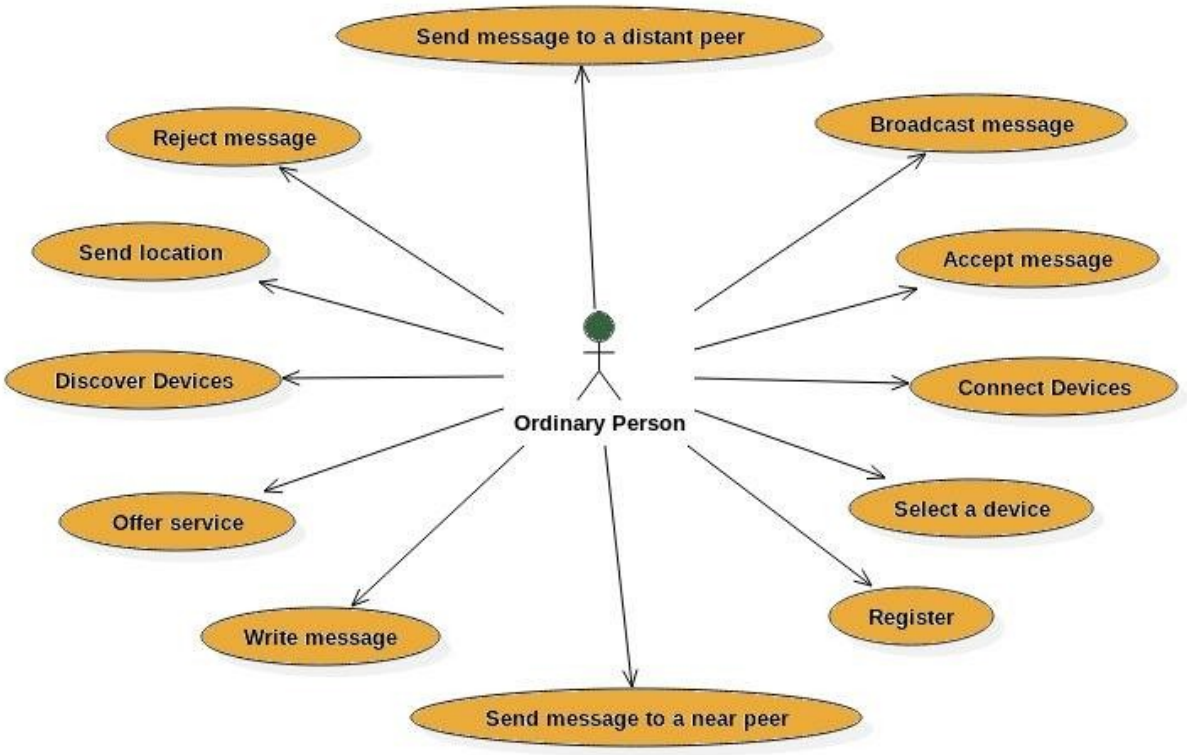


Figure 1 : Use case Ordinary Person

Use Case	Description
Register	Ordinary person has to register in order to use the application.
Offer a service	Ordinary person offers a service to be discovered by other devices.
Discover Device	Ordinary person lists the information of devices and device users.
Connect Devices	Ordinary person establishes communication with all devices around.
Select a device	Select only one device and connect to it
Write message	Ordinary person types a short message.
Send message to distant peer	Ordinary person can send message to a distant specific peer.
Send message to near peer	Ordinary person can send message to a near peer.
Broadcast message	Ordinary person can broadcast message to surrounding users.
Accept message	Ordinary person accepts message
Reject message	Ordinary person can reject messages.
Send Location	Ordinary person sends his location within the message

AFAD: Use case diagram

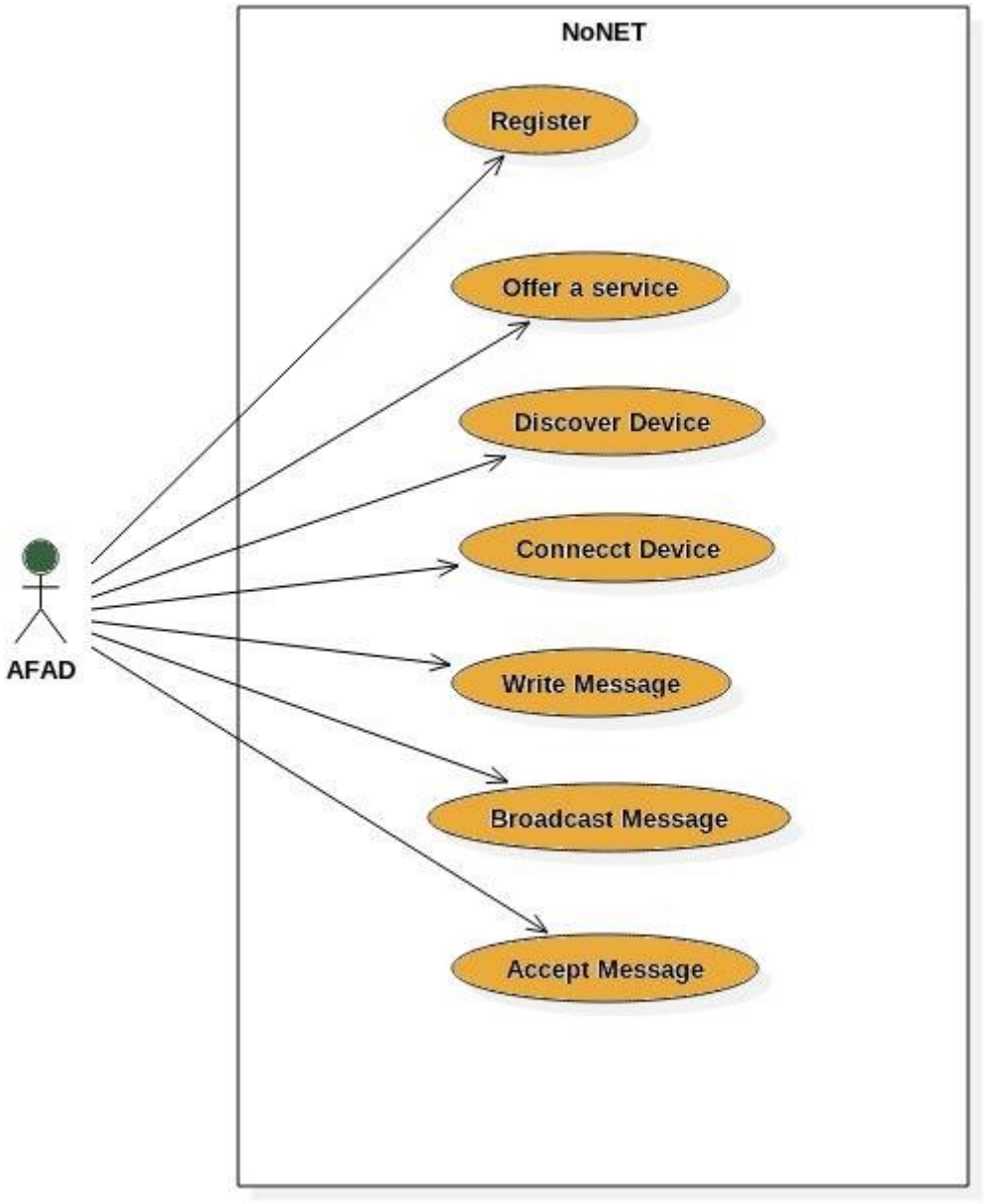


Figure 2 : Use case - AFAD

Use Case	Description
Register	AFAD has to register in order to use the application.
Offer a service	AFAD offers a service to be discovered by other devices.
Discover Device	AFAD lists the information of devices and device users.
Connect Devices	AFAD makes connection with the surrounding users.
Write Message	AFAD can write a message.
Broadcast Message	AFAD can broadcast messages to surrounding people.
Accept Message	AFAD accepts all the messages received.

2.1.2 Actor survey

AFAD

AFAD is an organization that helps people and provide their needs in an emergency situations. They can accept messages from the users and broadcast messages to the surrounding people.

Ordinary person

Ordinary user of our application can send message to surrounding people, broadcast messages and send message to distant specific user. They can also accept messages from other ordinary users and AFAD.

2.2 Interfaces

The application will be implemented in Java and it will use Android SDK. Moreover,. Eclipse will be used as IDE while implementing.

2.2.1 User Interfaces

User interface of our application will be easy to use and understandable. It uses Turkish language since our application is designed for Turkish people at any age. Moreover,the user is expected to know how to use Android or IOS mobile devices and to be able to write and read messages and use buttons.

User interfaces are explained in details below:

2.2.1.1 Login Interface

In this interface, there will be a button register. If user have not registered to the application, s/he will use register button and register to it.

2.2.1.2 Register Interface

In this interface user register to the system by giving information of himself in a provided text field. There will be a button register. After user filled the required fields with his/her related information (username, password, gender etc.), click to the register button and be able to login the application.

2.2.1.3 Browsing device interface

In this interface,there will be text ‘Cihaz aranıyor’ and if the device get the surrounding device’s user names ,MAC addresses, the device names will be listed. Thus, user can choose one of them to communicate or broadcast message to all of them.

2.2.1.3 Messaging interface

In this interface,user types message and click to send button. Also if any message is received there will pop up accept and reject buttons.

2.2.2 Hardware Interfaces

This application works on Android, IOS mobile devices and tablets. No other hardware is required.

2.2.3 Software Interfaces

Since this application is a mobile application, it will only need an Android version 4.0 or higher in order to perform.

2.2.4 Communications Interfaces

This application uses Wi-Fi Direct to communicate Android devices, Multi-peer connectivity to communicate IOS devices. We will use the Internet to communicate between IOS and Android devices since Bluetooth and Wi-Fi Direct communication are not supported between this two different platforms.

2.3 Constraints

- Users have to download the application on their phone in order to use it.
- NoNet will operate on Android v4.0 and IOS 7 or higher operating system.
- There should be at least 3 MB of free space on the device.
- Java should be the programming language used in implementation with Android SDK and IOS SDK.

3. Specific Requirements

With this section and later, we will describe the requirements of the software in detail. We will categorize requirements in 2 which are namely functional requirements and nonfunctional requirements.

3.1 Interface Requirements

3.1.1. Register Page

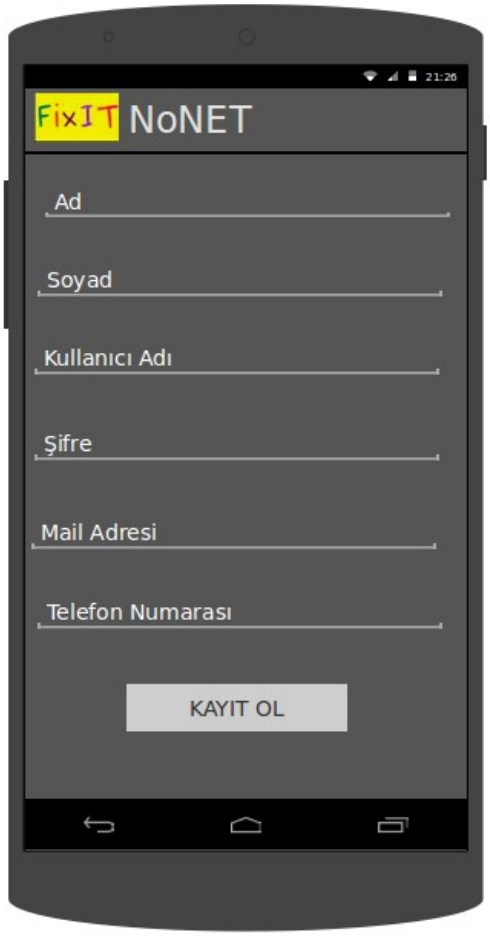


Figure 3 : Register Page

3.1.2. Main Page

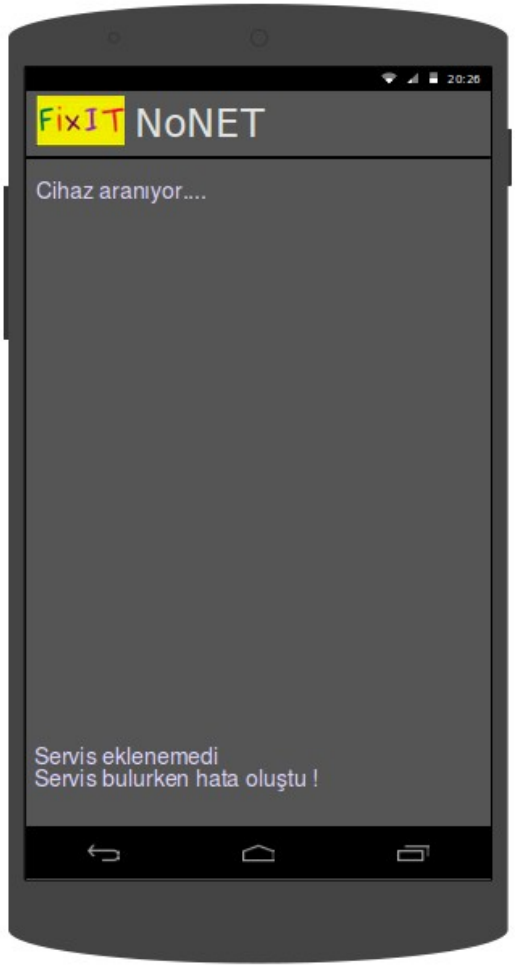


Figure 4 : Peer Registration Page

3.1.1. Available Peer List

Figure 5 : Available Peer List

3.1.2. Chat Page

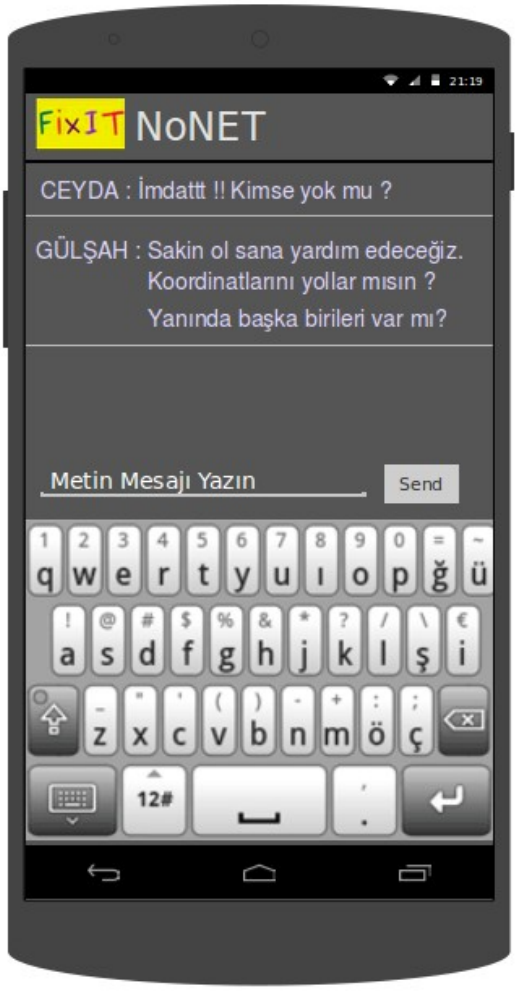


Figure 6 : Chat Page

3.2 Functional Requirements

3.2.1. User Register

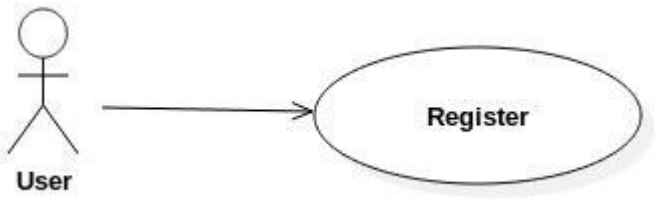


Figure 7 : Use Case Diagram - Register

Use Case Number	1
Use Case Name	Register
Summary	Ordinary person needs to register to the system
Actor	Ordinary person, AFAD
Trigger	Register Button
Precondition	Downloaded the NoNET application.
Scenario	• After user download the NoNET, s/he must register to system in order to use application. • Register button in the login page redirects user to the register page. • After filling required fields with information, s/he can register to the system with register button. • After user register the system, s/he can start to use the application.
Postcondition	After user registers the system, s/he can start to use the application.

3.2.2. Discover Devices

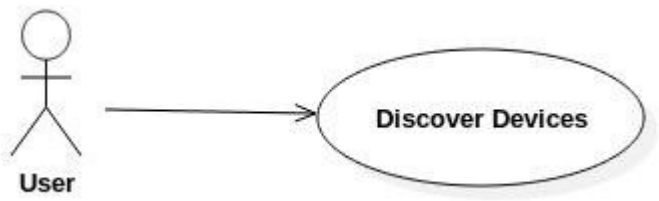


Figure 8 : Use Case Diagram - Discover Devices

Use Case Number	2
Use Case Name	Discover Devices
Summary	Discover other devices.
Actor	Ordinary person, AFAD
Trigger	Clicking to the NoNet Application
Precondition	Initiate the NoNET application .
Scenario	• When application is started, it discovers the near devices and distant certain devices without clicking any button.
Postcondition	After application discovers devices, it communicates and sends messages them.

3.2.3. Connect Devices

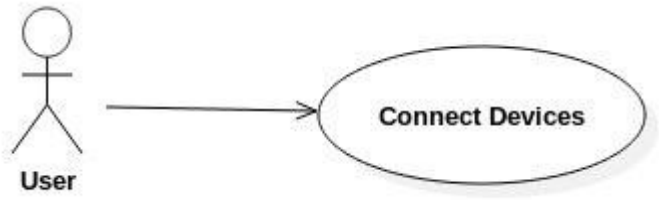


Figure 9 : Use Case - Connect Devices

Use Case Number	3
Use Case Name	Connect Devices
Summary	User connects to other devices.
Actor	Ordinary person, AFAD

Trigger	Clicking to other devices' name.
Precondition	User discovers to the near devices or distant certain devices.
Scenario	• When user sees the devices, s/he can connects with other people.
Postcondition	User broadcasts messages.

3.2.4. Select a Device

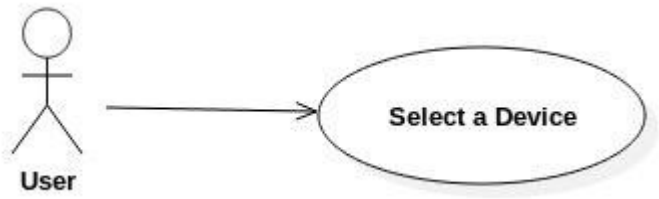


Figure 10 : Use Case - Select a Device

Use Case Number	4
Use Case Name	Select a Device
Summary	Select a device before sending message
Actor	Ordinary person
Trigger	Clicking to the other device' name.
Precondition	User discovers to the near device or distant certain device.
Scenario	• User select the device before sending message to a certain person.
Postcondition	User sends message.

3.2.5. Write Message

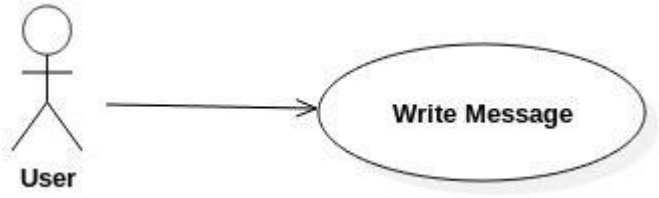


Figure 11 : Use Case - Write Message

Use Case Number	5
Use Case Name	Write Message
Summary	Type a text message
Actor	Ordinary person, AFAD
Trigger	Send
Precondition	Connect the other devices.
Scenario	• User writes a message by clicking on the text field. • Also, message content includes time of sent message and information about the user's situation.
Postcondition	Contact other devices.

3.2.6. Send Messages to a Distant Peer

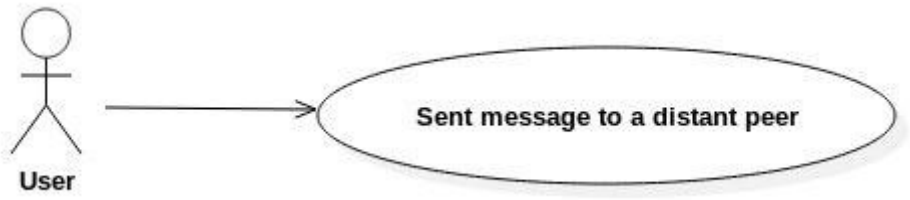


Figure 12 : Use Case - Sent message to a distant peer

Use Case Number	6
Use Case Name	Send Messages to a Distant Peer
Summary	User sends messages to a specific peer.
Actor	User
Trigger	Send
Precondition	Selected to the peer and found the correct path to send messages to the peer.
Scenario	• User sends messages to a specific peer. • S/he sends messages to her/his family members to inform about their situations or to receive information from them.
Postcondition	Receive information from the distant peer.

3.2.7. Send Messages to a Near Peer

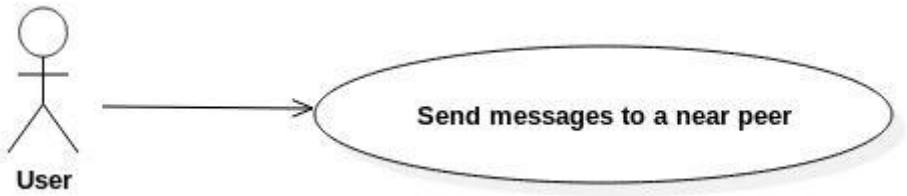


Figure 13 : Use Case - Send Messages to a near peer

Use Case Number	7
Use Case Name	Send Messages to a Near Peer.
Summary	User sends a short message.
Actor	Ordinary person
Trigger	Send
Precondition	Selected device and connected to it.
Scenario	• User sends a short message to ask help to a person near to her/him.
Postcondition	The receiver can reply this message.

3.2.8. Broadcast Messages

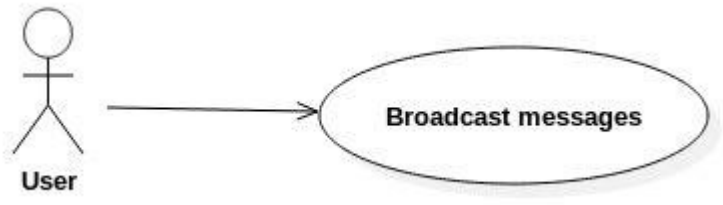


Figure 14 : Use Case - Broadcast Messages

Use Case Number	8
Use Case Name	Broadcast Messages
Summary	User broadcasts message to the surrounding people.
Actor	Ordinary person, AFAD
Trigger	Send
Precondition	Discovered devices.
Scenario	● User broadcasts a message to the surrounding people to ask help in emergency situations. ● AFAD broadcasts a message to people in communication range in order to learn if anyone needs help.
Postcondition	People will learn the location of people who is under the debris.

3.2.9. Offer a Service

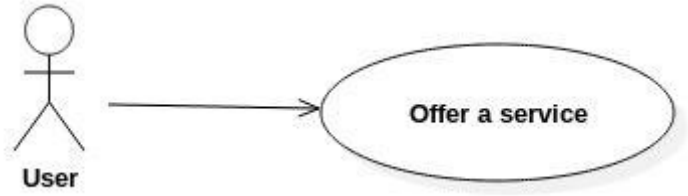


Figure : Use Figure 15 : Use Case - Offer a Service

Use Case Number	9
Use Case Name	Offer a Service
Actor	Ordinary person
Trigger	Clicking to the NoNet Application
Precondition	Wi-Fi Direct or Multi-peer Connectivity must be active.
Scenario	• User have to serve a service to be discovered by surrounding devices.
Postcondition	When user offers a service, other user can see the device and connect to it.

3.2.10. Accept Messages

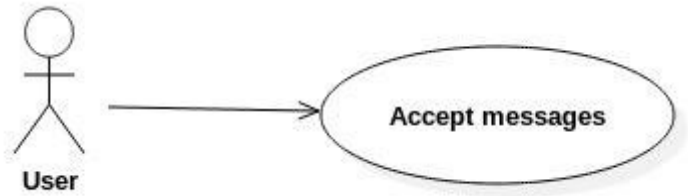


Figure : Use Figure 16 : Use Case - Accept Messages

Use Case Number	10
Use Case Name	Accept Messages
Actor	Ordinary person
Precondition	Other devices send messages. Also, user's device's battery should be higher than 30%.

Scenario	• Device accepts the message automatically in the background of the application.
Alternative Scenario	• If the user's device's battery is low, it will not attend to the ad-hoc network.
Postcondition	Message is displayed on the user screen.

3.2.11. Reject Messages

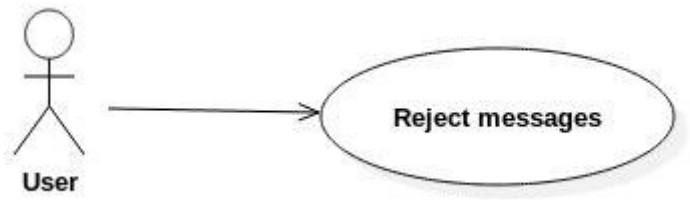


Figure 17 : Use Case - Reject Messages

Use Case Number	11
Use Case Name	Reject Messages
Actor	Ordinary person
Precondition	Other devices send messages. Also, user's device's battery should be lower than 30%.
Scenario	• When the user may be under debris and waiting for help, s/he can reject to transferring the received messages. Because, her/his device's battery may be low.
Postcondition	User be rescued before end of the battery.

3.2.12. Sent Location

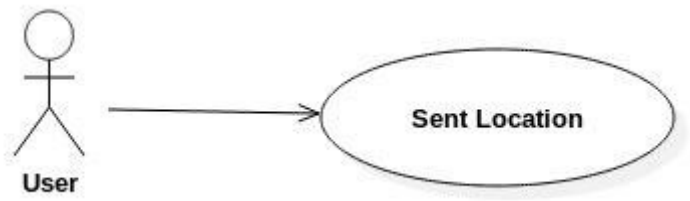


Figure 18 : Use Case - Sent Location

Use Case Number	12
Use Case Name	Sent Location
Summary	User sends the location of her/him.
Actor	Ordinary person
Trigger	Location Button
Precondition	Discovered devices and connected to them.
Scenario	• User of our application sends the location of her/his attached to the message to the other people. • User clicks location button and location of her/him will be sent.
Postcondition	People who come to the rescue learns her/his location.

3.3 Nonfunctional Requirements

In this section, last group of the requirements which is nonfunctional requirements will be explained in detail. Nonfunctional requirements are separated into usability, reliability, performance, supportability requirements subsections.

3.3.1 Usability

- This application is mobile application designed for Android and IOS users. Moreover, it is using Turkish language so that everyone literate with Turkish can use it. Furthermore, users have to know how to use mobile device and how to read and write messages. Thus, our application is easy to use and understandable so that no time is required for users to become productive at the usage of NoNET.

3.3.2 Reliability

- Application will be installed on mobile devices, it does not need to Internet connection to run. It shall be available in 24 hours a day, 7 days of a week on mobile device.

- Our application will deliver messages to the intended recipient(s), and then it will check if it is taken or not by intended recipient(s).If it is not taken by the users it will resend the message. In other words, it uses Reliable Data Transfer.

- NoNET is working in communication range (local area) and it is not dependent on GSM , so that in emergency situations even if too many people used the application at the same time, the failure of it shall be less than other operators and applications.

3.3.3 Performance

- The messages will be delivered to the users in a very short time less than 1 second and there will be delay which is mentioned in detail below:

- **Delay:** It is an important issue especially for broadcasting messages out of communication range. We are assuming that receiving peer will take the message at most in 30 seconds.

- The message will be very short since in emergency situations, communicating with surrounding people as fast as possible is important. That is the capacity is small.

- In our first scenario where the user send message to a peer,there will be 2 customers. For the case, sending message to a certain user ,the customer's number is again 2.

- For broadcasting scenarios,the number of customers depends on the users who downloaded the application NoNET. However, since NoNET allows ordinary users to hop messages to the people around until 15th hop, there will 15* (user number for 1 hop) customers. For AFAD, there will be 30*(user number for 1 hop) customers.

- Users should have 3MB free space on their mobile devices. CPU speed or RAM of the device is not a big concern.

3.3.4 Supportability

- Programming language used in this project is object-oriented so that the tasks are independent of each other and therefore easy to maintain.

- The codes must be easy to understand and readable.

- All design architecture should be well documented.

3.3.5 Security

- In order to specify the sender of the message, our application need an authentication system. We will develop a GSM number or Email/Facebook account based authentication system for the application.

- Sent and received message should be transferred via Reliable Data Transfer connection.

3.3.6 Integrity

Our application will certificate that whether a message comes from AFAD or not. And for ordinary users there will be an authentication system.

3.3.7 Priority

As we mentioned above, we have 2 kind of users. One of them is AFAD personnels. We think that their role in emergency situations is more vital so we will assign them high

priority than others. If more than a message is trying to reach a receiver at the same time, the message with high priority will reach more immediately.

4. Data Model and Description

UML Class Diagrams will be presented in this section. We have 5 classes which are User, AFAD personnel, Ordinary User, Message and Location.

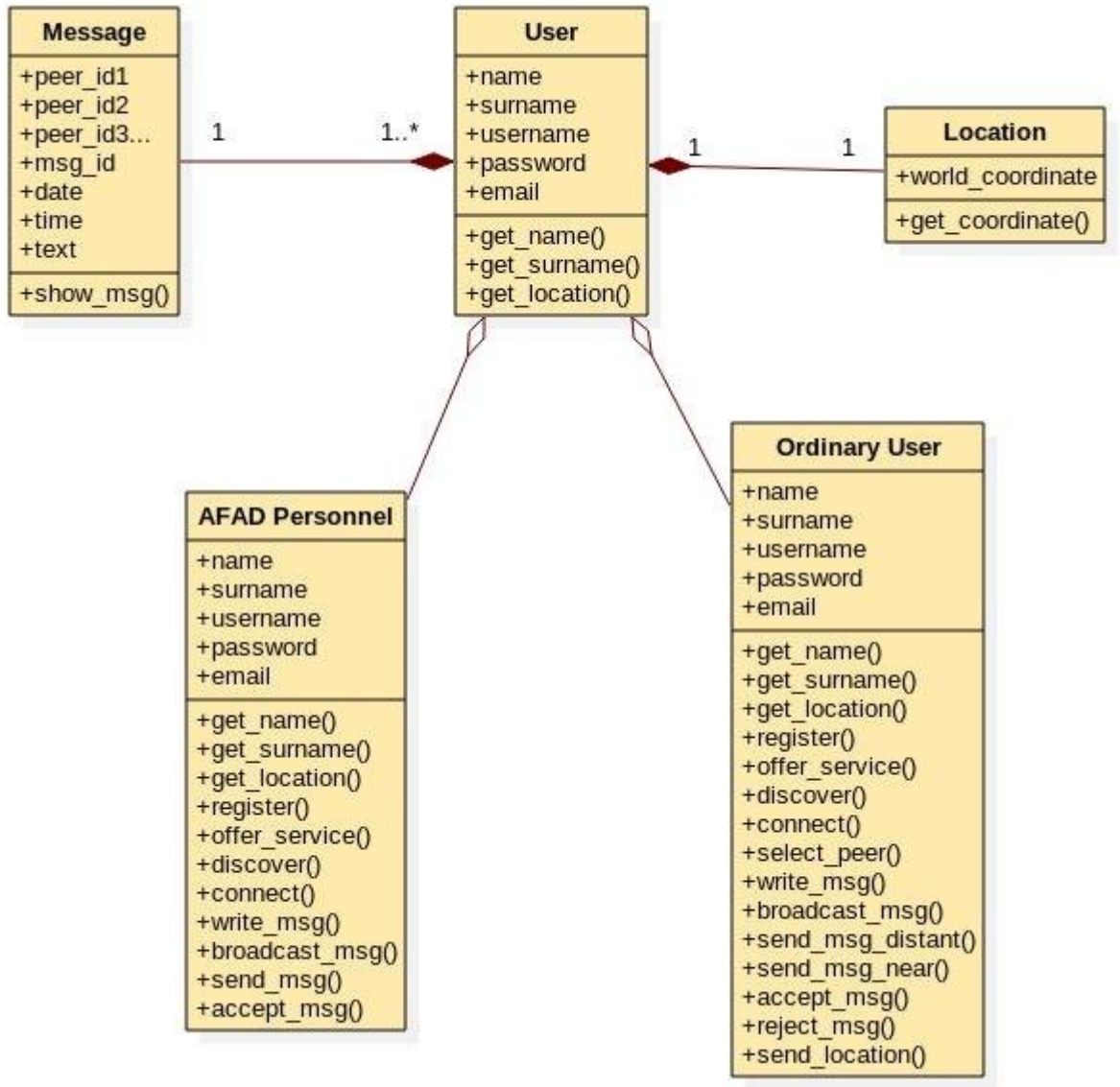


Figure 19 : Class Diagram

Message Data Attributes

Data Item	Type	Description
peer_ID	String	Unique ID of the peer device.
message_ID	String	Unique ID of the messages.
text	String	Message content.
time	Date	Sending time of the message.
date	Date	Sending date of the message.

User Data Attributes

Data Item	Type	Description
name	String	Name of the user.
surname	String	Surname of the user.
username	String	Username of the user.
password	String	Password of the user.
email	String	Email address of the user.

Location Data Attributes

Data Item	Type	Description
world_coordinate	String	Gives the location information of the user.

AFAD Personnel Attributes

Data Item	Type	Description
name	String	Name of the user.
surname	String	Surname of the user.
username	String	Username of the user.
password	String	Password of the user.
email	String	Email address of the user.

Ordinary User Attributes

Data Item	Type	Description
name	String	Name of the user.
surname	String	Surname of the user.
username	String	Username of the user.
password	String	Password of the user.
email	String	Email address of the user.

Figure 20 : Component Diagram for Android and IOS communication

We are planning to communicate Android and IOS devices in our project. However, these devices could not be able to communicate via Bluetooth or Wi-Fi Direct so we will set this connection with the help of the Wi-Fi/GSM. These technologies will only be used for communicating different platforms.

5. Behavioral Model and Description

5.1 State Transition Diagram

The state transition diagram in Figure shows all the states, triggers and conditions for each transition and related events.

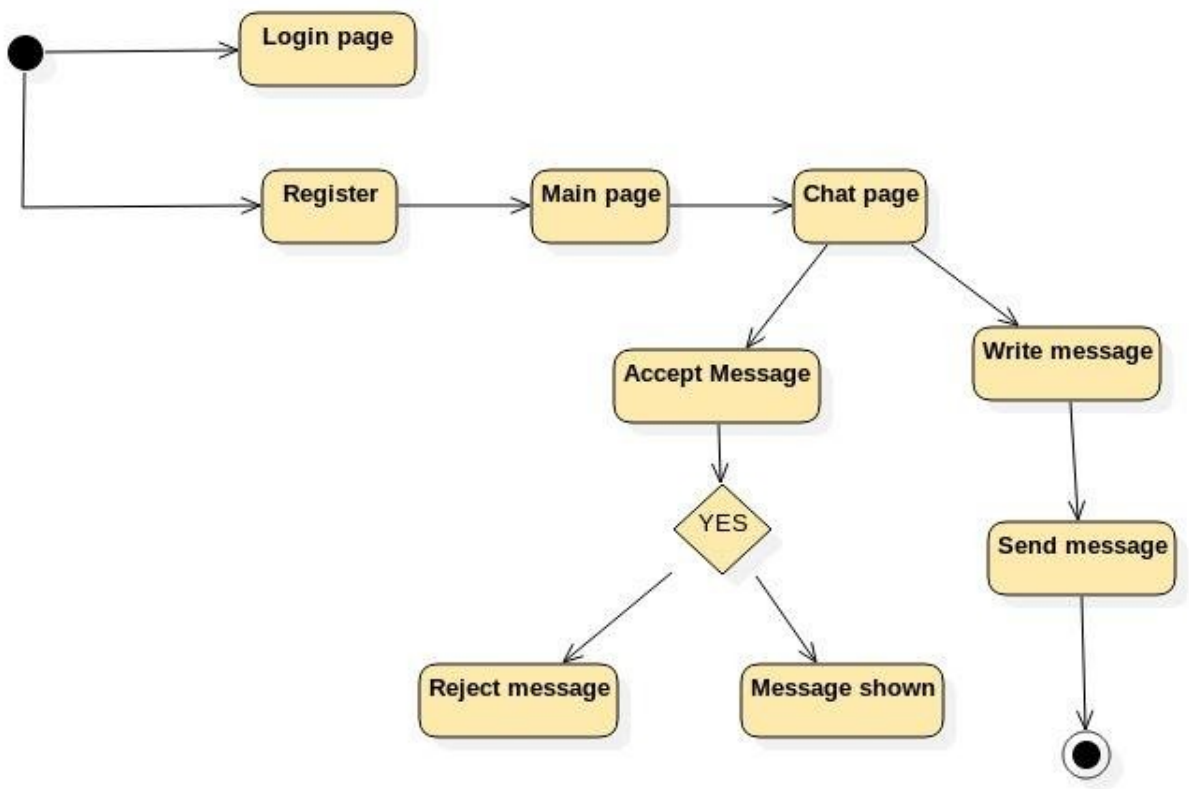


Figure 21 : State Transition Diagram

6.Planning

6.1 Team Structure

We periodically meet with Dr. Ertan Onur for consultancy. In our group,everyone contributes to each part of the work. Our members are below:

- Ceyda Tosun
- Gülşah Sabırsız
- Gulnaz Shaidolda
-

6.2 Process Model

We use Agile Scrum software development modeling this project. Agile method is based on an iterative approach. Each iteration takes 2-4 weeks.

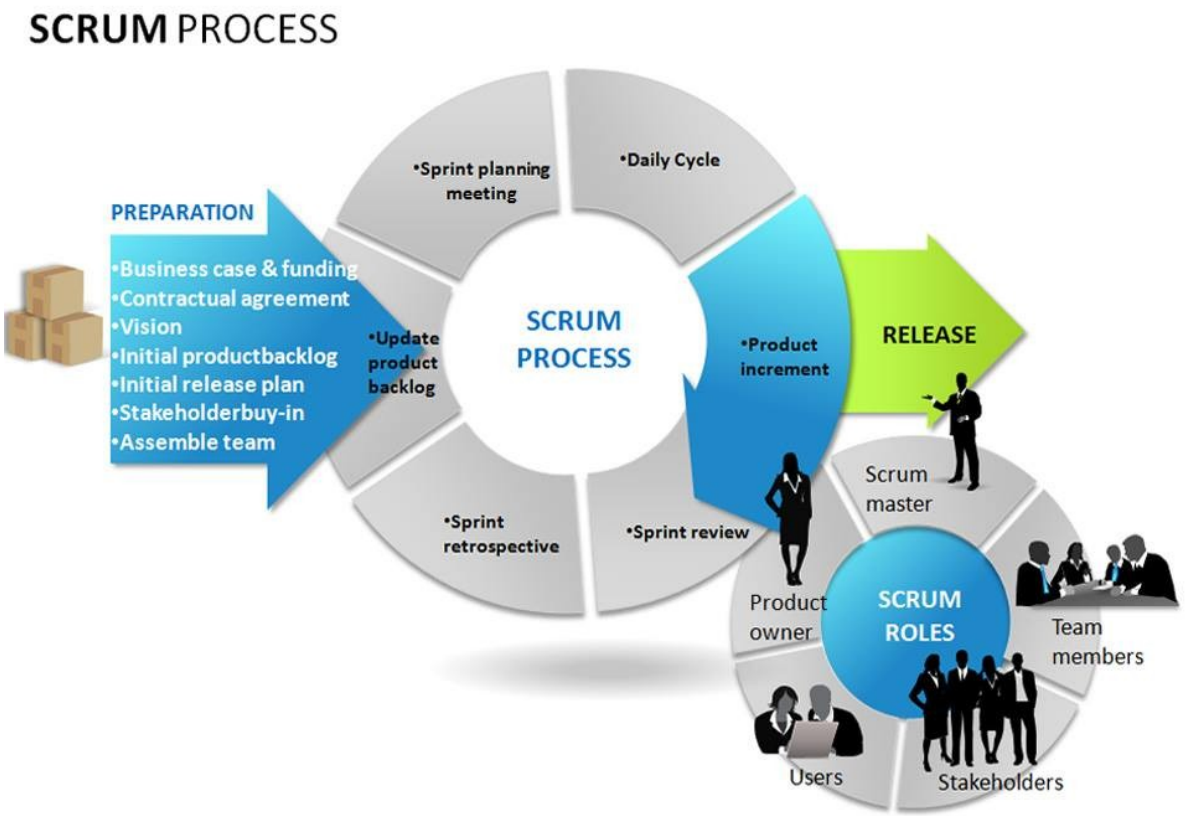


Figure 22: Process Model (Source: Newman, 2013.) Available at: https://www.inqbation.com/agile-methodology-of-web development/ [Viewed 13 January 2016].[3]

7.Conclusion

In this project, we aim to help people to communicate in emergency situations. The Software Requirement Specification states system interfaces, user interfaces, software and hardware interfaces,communication interfaces and functional and nonfunctional requirements in detail. Developer of the project will make use of this document throughout the implementation process.

8. References

[1] IEEE Guide for Software Requirements Specifications," in IEEE Std 830-1984 , vol., no., pp.1-26, Feb. 10 1984, doi: 10.1109/IEEESTD.1984.119205, URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=278253&isnumber=6883
[2] IDC, 2015. Smartphone OS Market Share. Available from: http://www.idc.com/prodserv/smartphone-os-market-share.jsp [Accessed 13 January 2016].

[3] Newman, 2013.Process Model.Available at: https://www.inqbation.com/agile-methodology-of-web-development/ [Viewed 13 January 2016].