Report Stress

İngilizce olarak seçilmiş Report Stress ismi stres raporlama anlamına gelmektedir. Projem en basit haliyse çocukların stres durumlarını ebeveynlerine otonom bir şekilde bildirmeyi hedeflemektedir. Bu bildirimler hem acil durumlarda anlık bildirim olarak bulunmaktadır, hem de uzun vadeli stres verisi depolanarak bu veriden anlamlı bir sonuç çıkarmaya yaramaktadır.

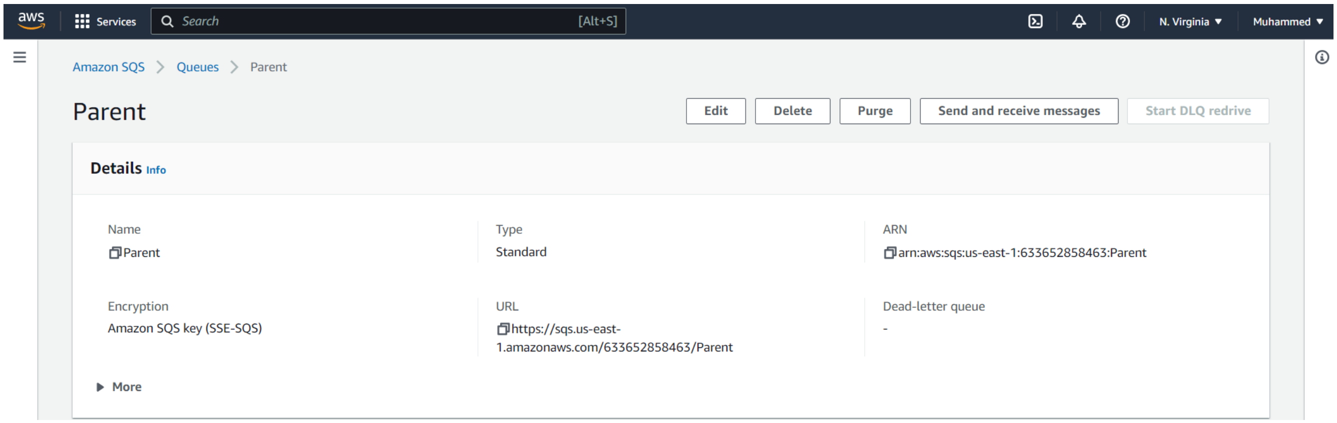
Teknik Detaylar

Projenin arayüzü mockup olarak tasarlanmış halde bulunmaktadır. Projenin şu anda hayata geçebilen kısmı, acil durum kısa mesajları (mobil hatlara gönderilen) ve anlık stres bildirimlerini cloud sistemi kullanarak aktarma ile sınırlı kalmıştır.

Amazon Web Services(AWS)

Projemizin cloud sisteminin temelini oluşturan araç. Bu araca ait Simple Queue Service (Basit Sıra Hizmeti) ve Simple Notification Service(Basit Bildirim Sistemi) projenin temel alt yapısını oluşturmaktadır.

Simple Queue Service (Basit Sıra Hizmeti) (SQS)



Parent

Details Info

Name: Parent

Type: Standard

ARN: arn:aws:sqs:us-east-1:633652858463:Parent

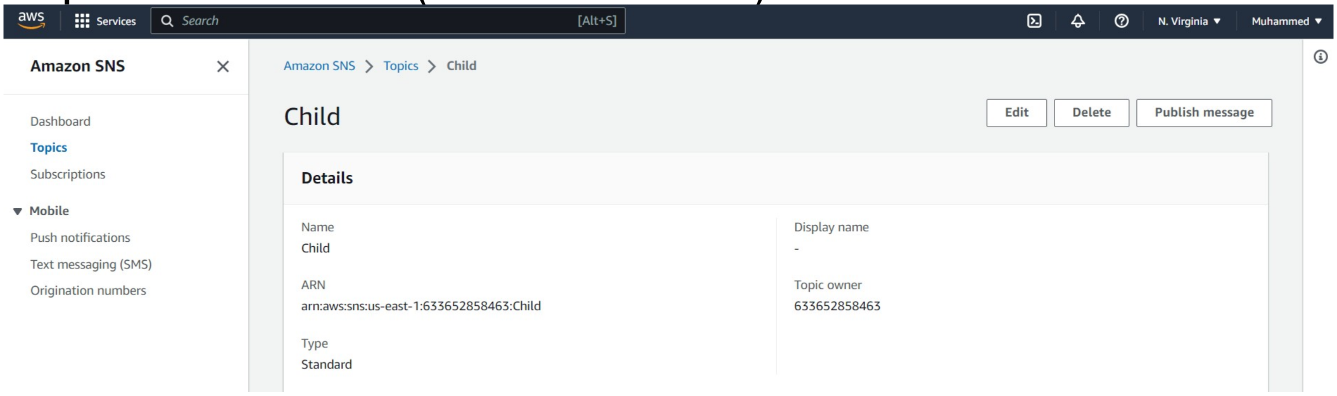
Encryption: Amazon SQS key (SSE-SQS)

URL: https://sqs.us-east-1.amazonaws.com/633652858463/Parent

Dead-letter queue: -

Görselde gösterilen topic(başlık) ebevenyne ait SQS'i göstermektedir görseldeki panelden başlığa erişmek için gereken bağlantılar ve güvenlik protokolüne ait detaylara erişilebilir. Basitçe SQS'in özelliği: veriyi almak için hazırda bekleyen bir birimdir. SQS veriler sürekli olarak yüksek sıklıkla geldiğinde, onları bir sıraya dizerek belirli bir gecikmeyle de olsa sıralı bir şekilde kullanıcıya iletmeyi sağlar.

Simple Notification Service(Basit Bildirim Sistemi)



Child

Details

Name: Child

Display name: -

ARN: arn:aws:sns:us-east-1:633652858463:Child

Topic owner: 633652858463

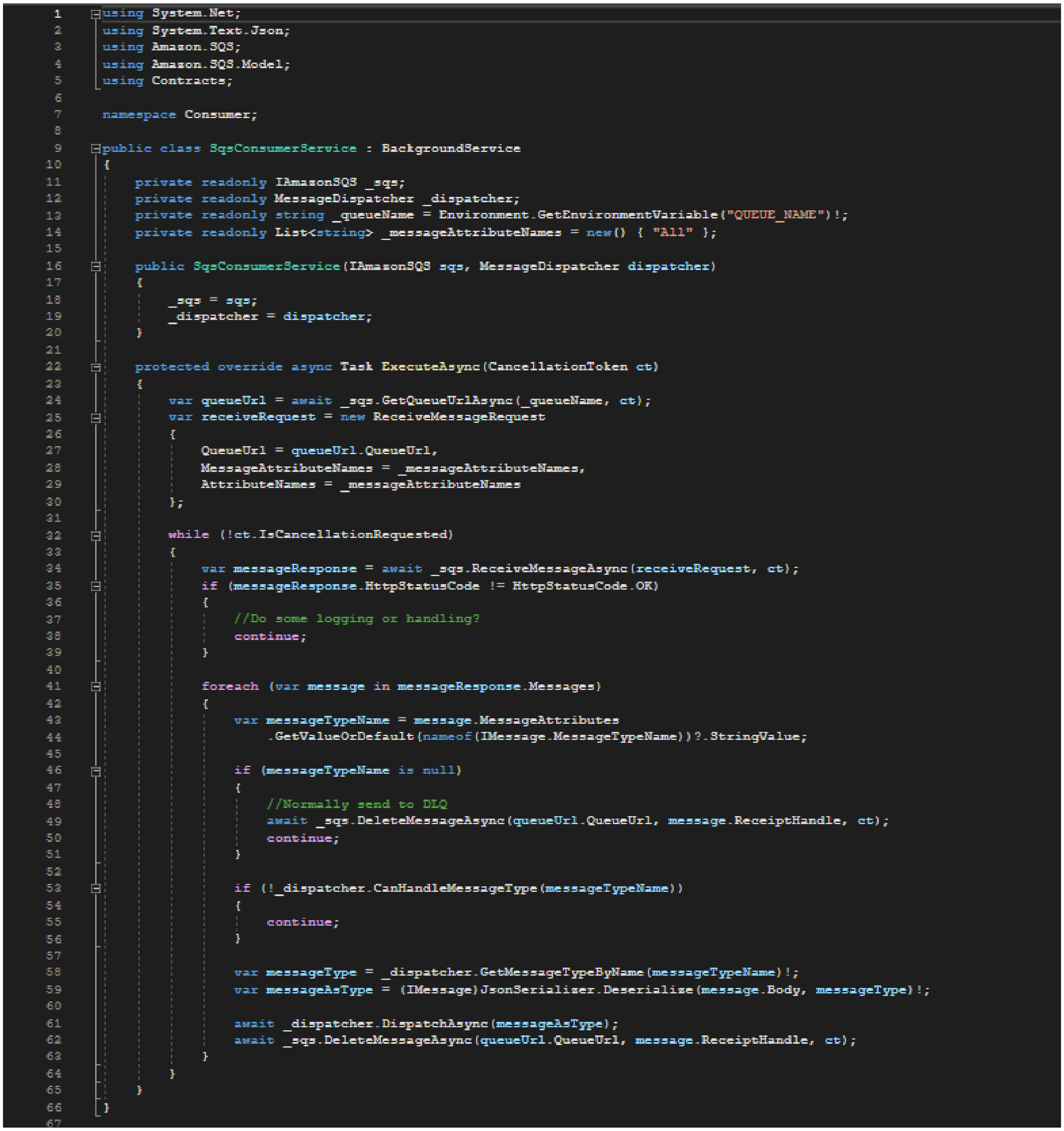
Type: Standard

SNS ise veri almak için hazırda bekleyen SQS'e, veri göndermekten sorumlu birimdir. Çocuğa ait stres verilerinde belirli bir eşiğin üzerinde dalgalanma olduğunda SNS SQS'e yani ebeveyne veri(bildirim) gönderir. Yine görseldeki panel SNS başlığını yönetmek için kullanılan paneli ve başlığın temel özelliklerini göstermektedir.

SNS ve SQS sistemleri FIFO(First In First Out) Queue sistemi üzerine dizayn edilmiştir.

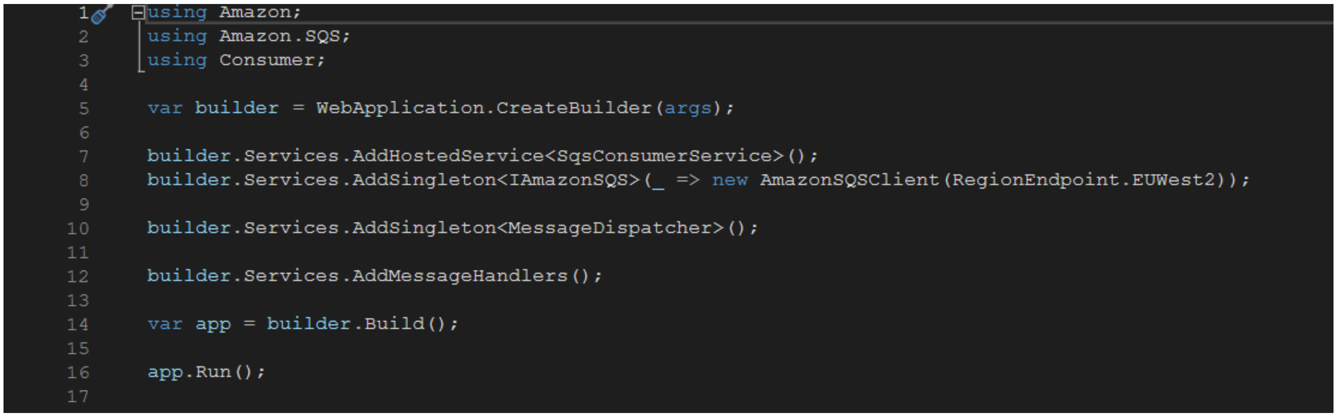
KOD TANITIM BÖLÜMÜ

SQS SİSTEMİ



```
using System.Net;
using System.Text.Json;
using Amazon.SQS;
using Amazon.SQS.Model;
using Contracts;

namespace Consumer;

public class SqsConsumerService : BackgroundService
{
    private readonly IAmazonSQS _sqs;
    private readonly MessageDispatcher _dispatcher;
    private readonly string _queueName = Environment.GetEnvironmentVariable("QUEUE_NAME")!;
    private readonly List<string> _messageAttributeNames = new() { "All" };

    public SqsConsumerService(IAmazonSQS sqs, MessageDispatcher dispatcher)
    {
        _sqs = sqs;
        _dispatcher = dispatcher;
    }

    protected override async Task ExecuteAsync(CancellationToken ct)
    {
        var queueUrl = await _sqs.GetQueueUrlAsync(_queueName, ct);
        var receiveRequest = new ReceiveMessageRequest
        {
            QueueUrl = queueUrl.QueueUrl,
            MessageAttributeNames = _messageAttributeNames,
            AttributeNames = _messageAttributeNames
        };

        while (!ct.IsCancellationRequested)
        {
            var messageResponse = await _sqs.ReceiveMessageAsync(receiveRequest, ct);
            if (messageResponse.HttpStatusCode != HttpStatusCode.OK)
            {
                //Do some logging or handling?
                continue;
            }

            foreach (var message in messageResponse.Messages)
            {
                var messageTypeName = message.MessageAttributes
                    .GetValueOrDefault(nameof(IMessage.MessageTypeName))?.StringValue;

                if (messageTypeName is null)
                {
                    //Normally send to DLQ
                    await _sqs.DeleteMessageAsync(queueUrl.QueueUrl, message.ReceiptHandle, ct);
                    continue;
                }

                if (!_dispatcher.CanHandleMessageType(messageTypeName))
                {
                    continue;
                }

                var messageType = _dispatcher.GetMessageTypeByName(messageTypeName)!;
                var messageAsType = (IMessage)JsonSerializer.Deserialize(message.Body, messageType)!;

                await _dispatcher.DispatchAsync(messageAsType);
                await _sqs.DeleteMessageAsync(queueUrl.QueueUrl, message.ReceiptHandle, ct);
            }
        }
    }
}
```

Görseldeki kod bloğu SQS sisteminin genel mimarisini, başlıklarını kendi arasındaki mesajlaşma(veri iletişimindeki) mesaj uzantılarını ve iletişim protokolünü düzenlemekten sorumludur

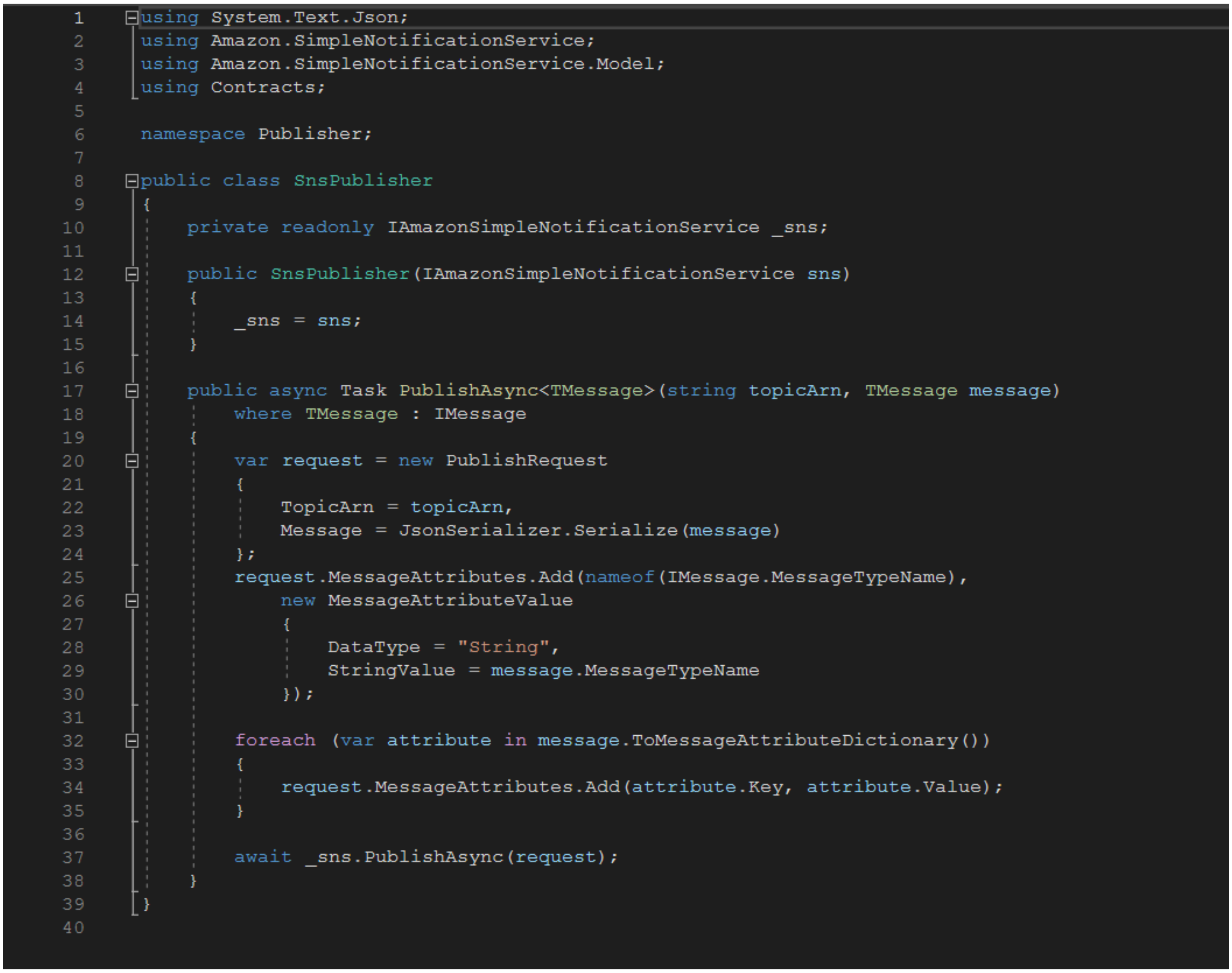


```
using Amazon;
using Amazon.SQS;
using Consumer;

var builder = WebApplication.CreateBuilder(args);

builder.Services.AddHostedService<SqsConsumerService>();
builder.Services.AddSingleton<IAmazonSQS>(_ => new AmazonSQSClient(RegionEndpoint.EUWest2));

builder.Services.AddSingleton<MessageDispatcher>();

builder.Services.AddMessageHandlers();

var app = builder.Build();

app.Run();
```

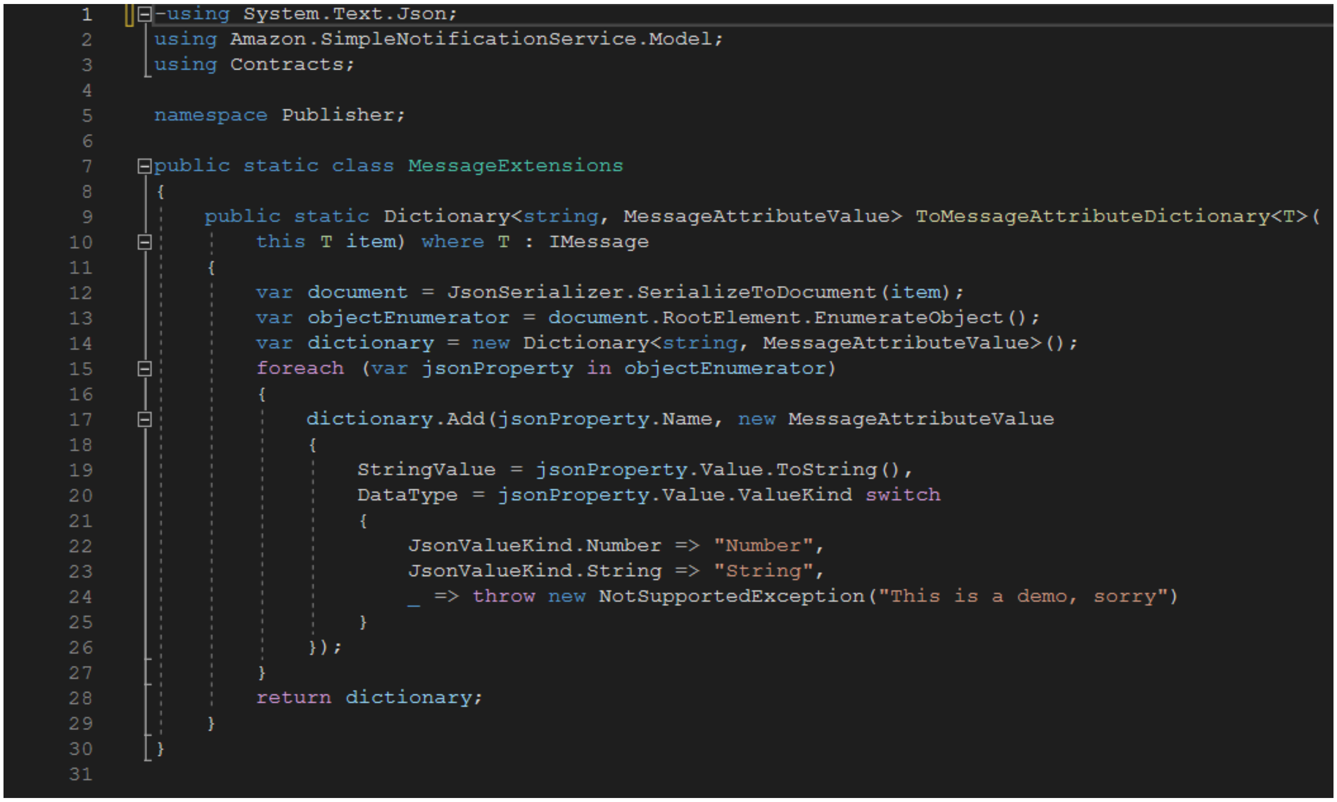
Bu görseldeki kod ise SQS sistemini çalıştırmaktan sorumludur.

SNS SİSTEMİ

Yine görseldeki kod SNS sisteminin hangi SQS sistemine mesaj göndermesi gerektiğine karar veren ve iletişim protokolünü belirlemekle görevlidir.



```
using System.Text.Json;
using Amazon.SimpleNotificationService;
using Amazon.SimpleNotificationService.Model;
using Contracts;

namespace Publisher;

public class SnsPublisher
{
    private readonly IAmazonSimpleNotificationService _sns;

    public SnsPublisher(IAmazonSimpleNotificationService sns)
    {
        _sns = sns;
    }

    public async Task PublishAsync<TMessage>(string topicArn, TMessage message)
        where TMessage : IMessage
    {
        var request = new PublishRequest
        {
            TopicArn = topicArn,
            Message = JsonSerializer.Serialize(message)
        };
        request.MessageAttributes.Add(nameof(IMessage.MessageTypeName),
            new MessageAttributeValue
            {
                DataType = "String",
                StringValue = message.MessageTypeName
            });

        foreach (var attribute in message.ToMessageAttributeDictionary())
        {
            request.MessageAttributes.Add(attribute.Key, attribute.Value);
        }

        await _sns.PublishAsync(request);
    }
}
```



```
using System.Text.Json;
using Amazon.SimpleNotificationService.Model;
using Contracts;

namespace Publisher;

public static class MessageExtensions
{
    public static Dictionary<string, MessageAttributeValue> ToMessageAttributeDictionary<T>(
        this T item) where T : IMessage
    {
        var document = JsonSerializer.SerializeToDocument(item);
        var objectEnumerator = document.RootElement.EnumerateObject();
        var dictionary = new Dictionary<string, MessageAttributeValue>();
        foreach (var jsonProperty in objectEnumerator)
        {
            dictionary.Add(jsonProperty.Name, new MessageAttributeValue
            {
                StringValue = jsonProperty.Value.ToString(),
                DataType = jsonProperty.Value.ValueKind switch
                {
                    JsonValueKind.Number => "Number",
                    JsonValueKind.String => "String",
                    _ => throw new NotSupportedException("This is a demo, sorry")
                }
            });
        }
        return dictionary;
    }
}
```

Görseldeki kod ise SNS sisteminin mesaj uzantısını ve mesajın içerdiği veri tiplerini belirlemekle görevlidir.

Ankara University, Engineering Faculty

ReSt

Stress Reporter

Alternative Mockups

Muhammed Said YETER

2022

Individual research project

saidyeter@gmail.com

Annotation

- In this presentation, all the features from installation to usage of the application will be explained.
- How the user types form the interface will be the main subject.

Contents

1. Introduction

1.1 Project Titles

- ReSt (Report Stress)
- Real time stress reporter for depended individuals.

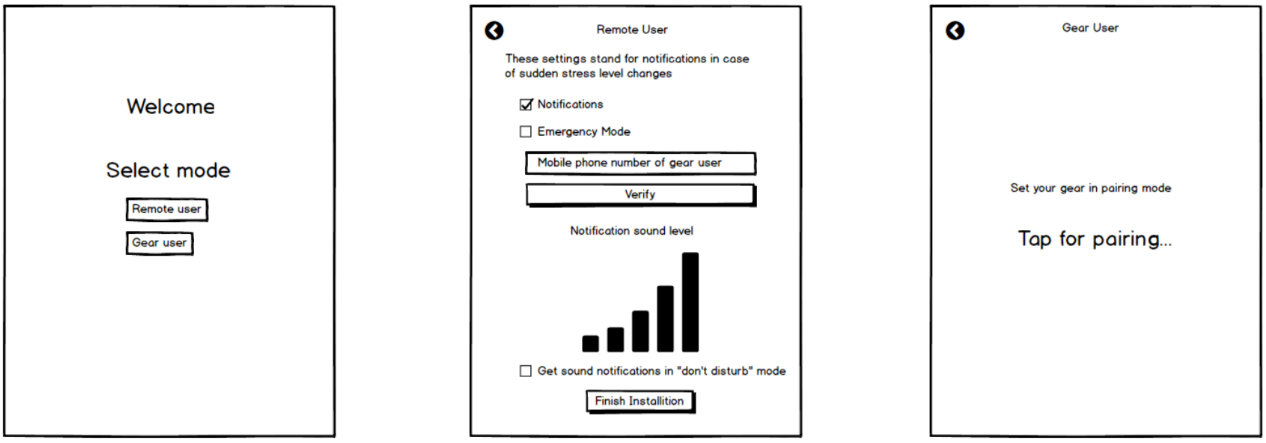
1.2 Problem Description

- Parents generally want to follow their children’s situation when they are not standing by them. Getting sudden notifications when the children are under the risk of any kind of danger is one of parent’s needs. Nonetheless, getting statistical information of children’s mental stress will also help parent’s to learn presence of a problem that bothers children.

2. ReportStress 2 Mockup Muhammed Said Yeter

- This mockup is for the users want to get more statistical information to follow and control their children’s mental health.
- Receiving emergency alerts is optional.

Login Screen



Welcome

Select mode

Remote User

These settings stand for notifications in case of sudden stress level changes

☑ Notifications

□ Emergency Mode

Mobile phone number of gear user

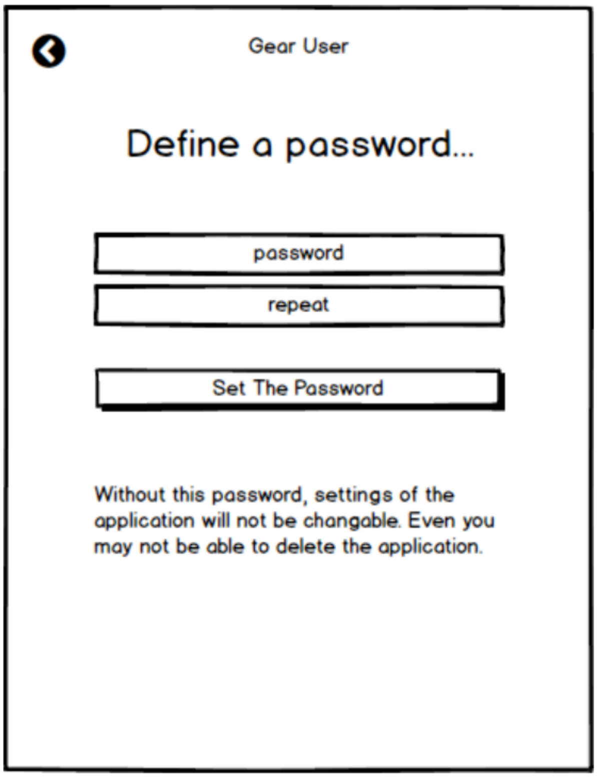
□ Get sound notifications in "don't disturb" mode

Gear User

Set your gear in pairing mode

Tap for pairing...

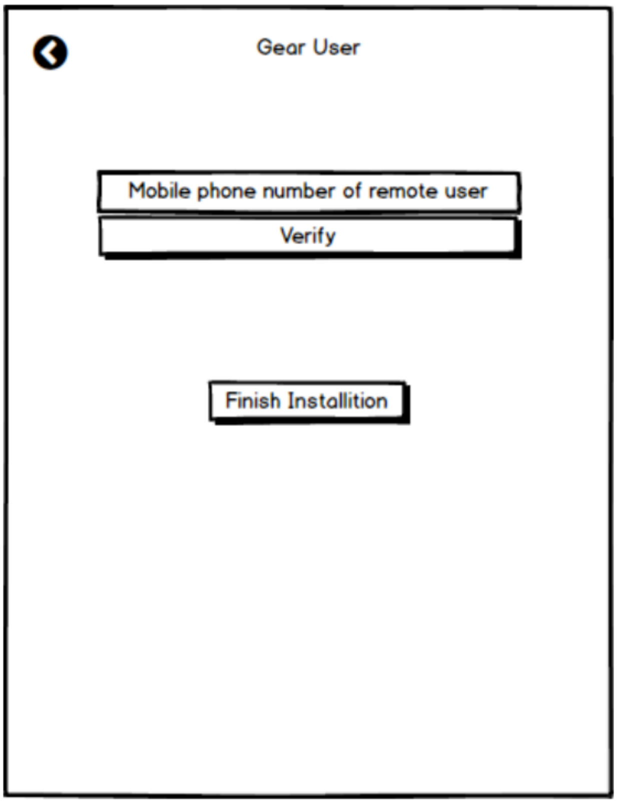
After selecting your mode, you should adjust notification, emergency settings then pair the gear with gear user's device.



Gear User

Define a password...

Without this password, settings of the application will not be changable. Even you may not be able to delete the application.

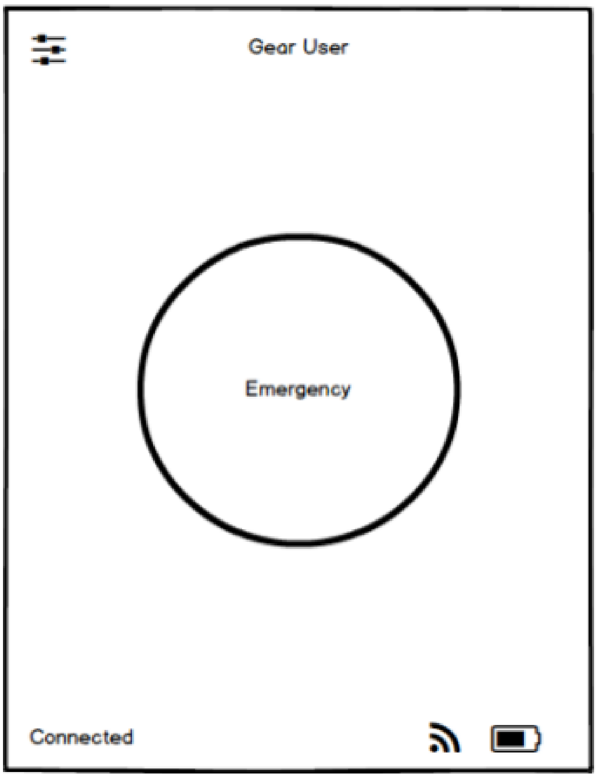


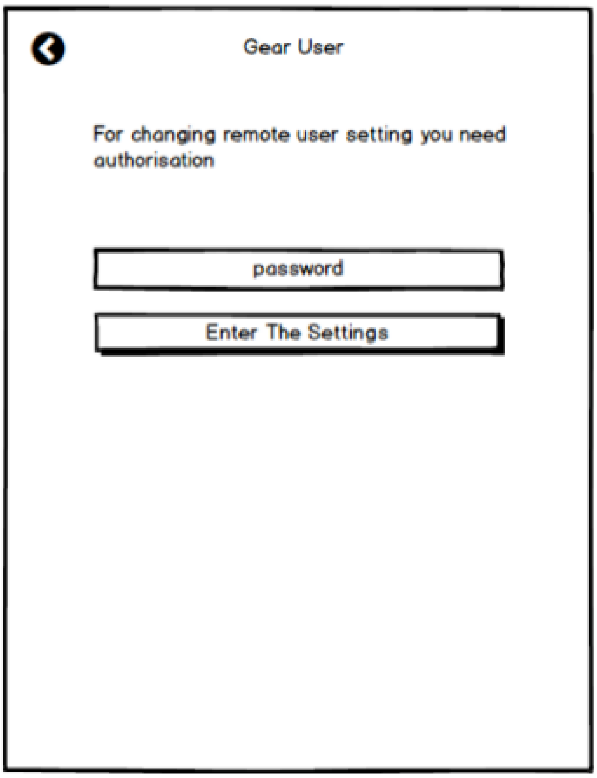
Gear User

Verifying the phone number is important because in case of an emergency situation, emergency message will be sent via this phone number.

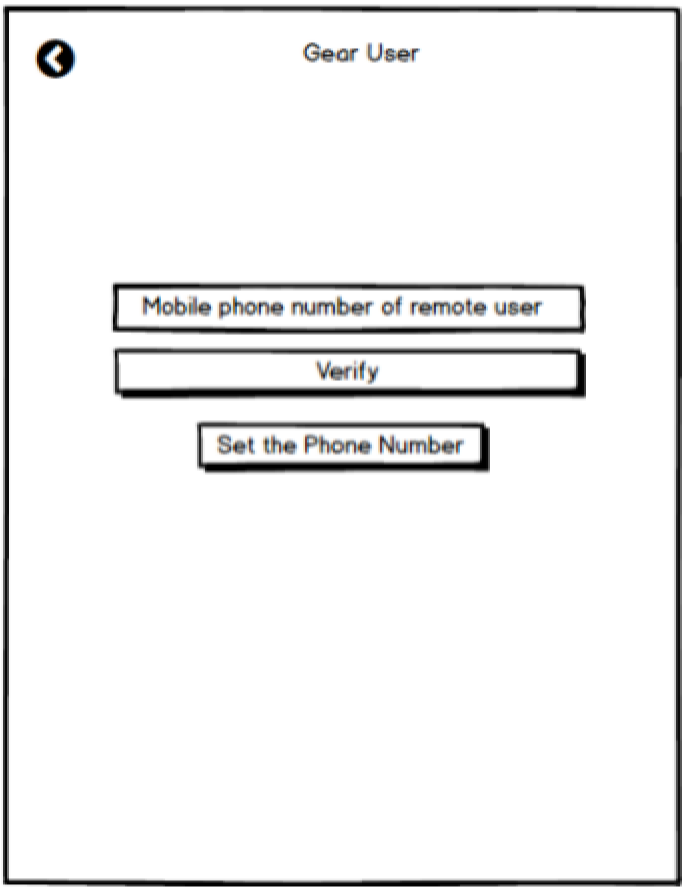
It is important to not forgetting the password. This password had better be known by only remote user.

Gear User’s Settings

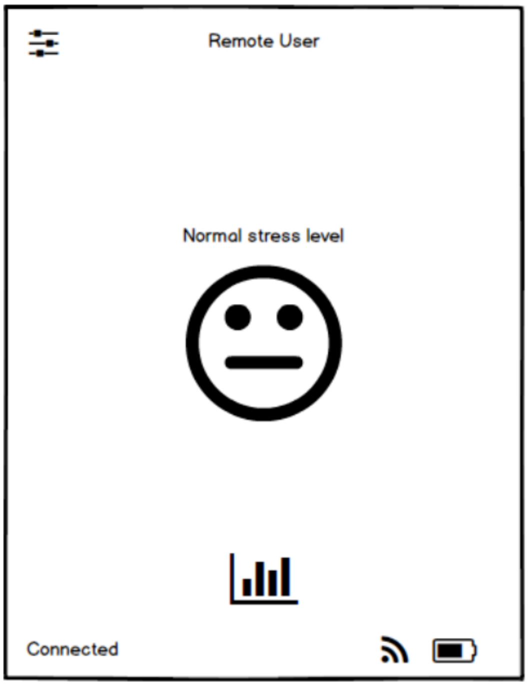


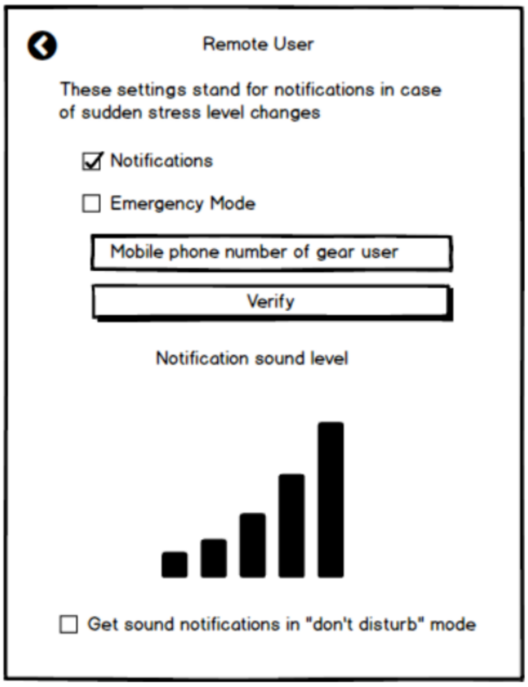


For changing remote user setting you need authorisation



Remote User’s Settings





These settings stand for notifications in case of sudden stress level changes

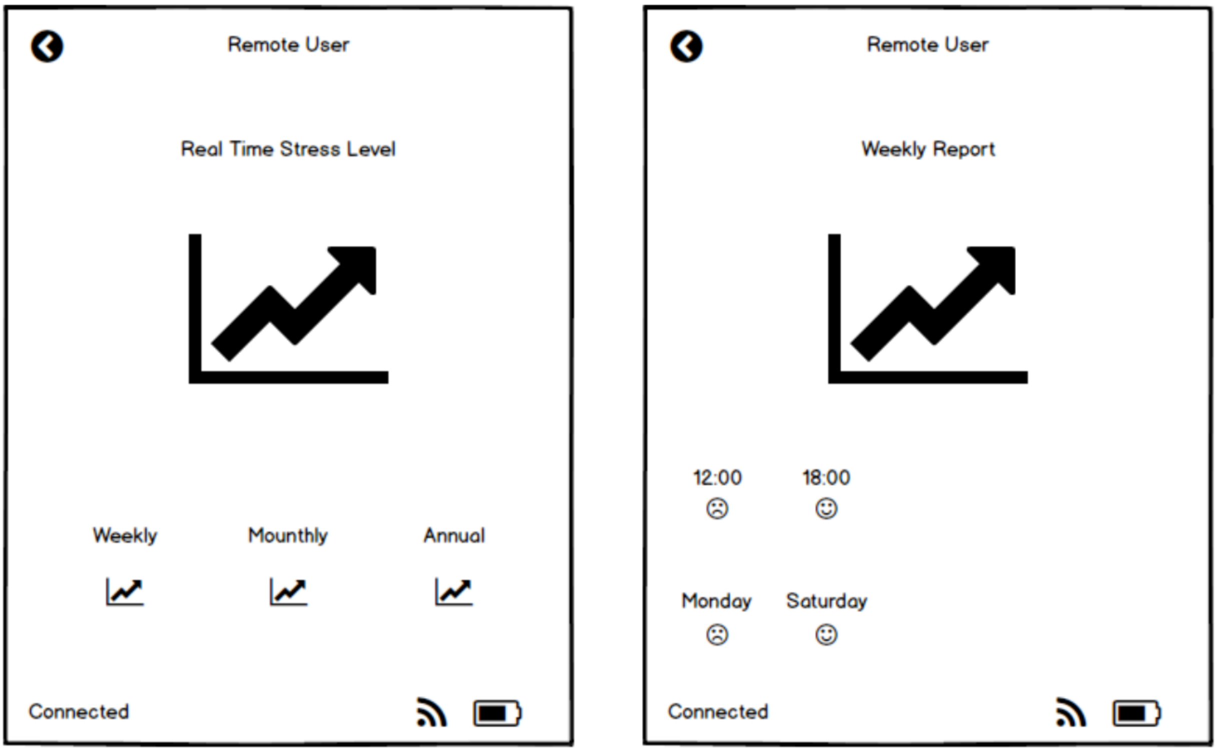
☑ Notifications

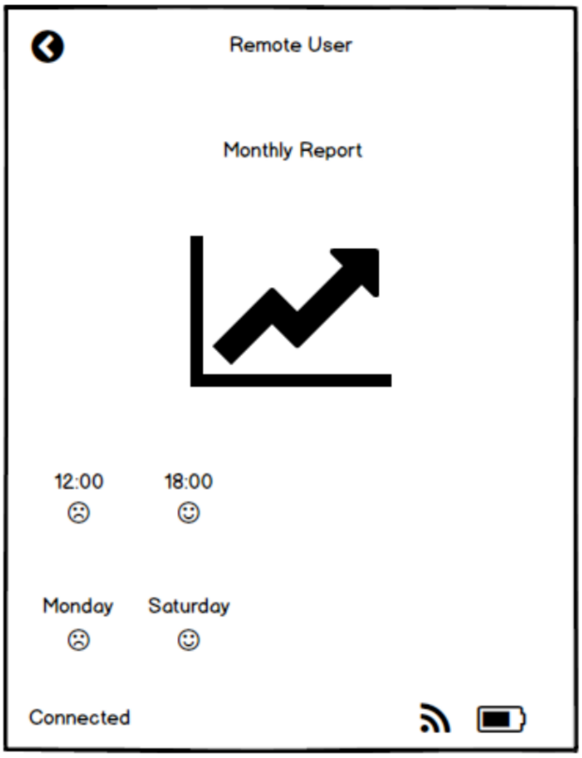
□ Emergency Mode

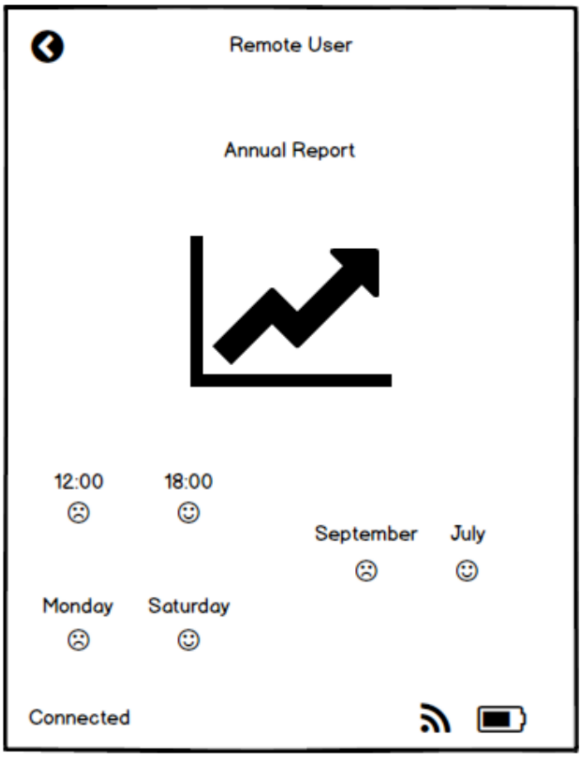
Mobile phone number of gear user

□ Get sound notifications in "don't disturb" mode

Statistics for Remote User



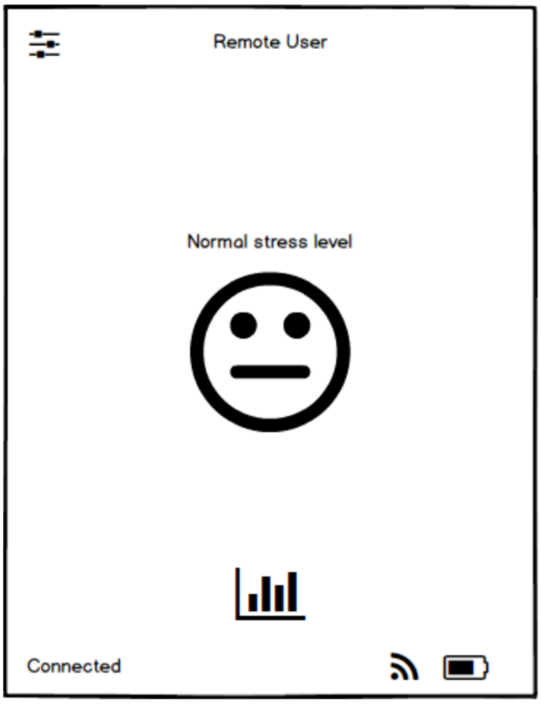


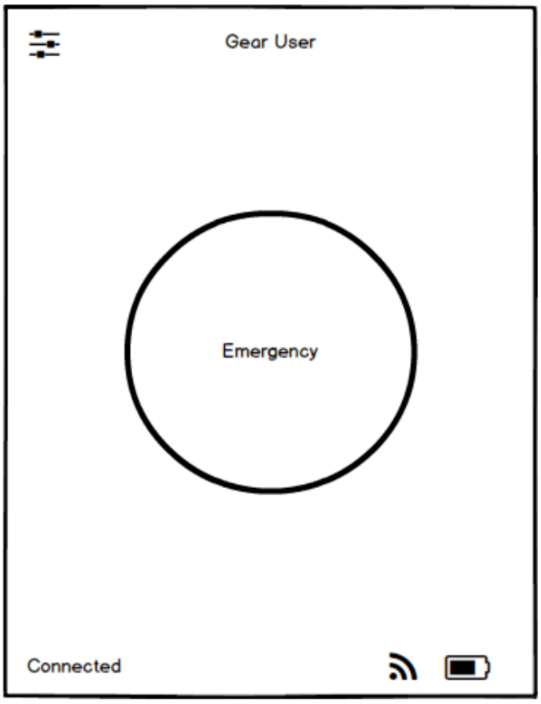


Stress level will be recorded real-time, and be logged hour by hour, day by day and month by month. Visualizations will help user for a better understanding of the data.

These statistics can be exported or shared by holding graph icon. Thus they might be used for helping any psychologically amelioration process.

Home Screen

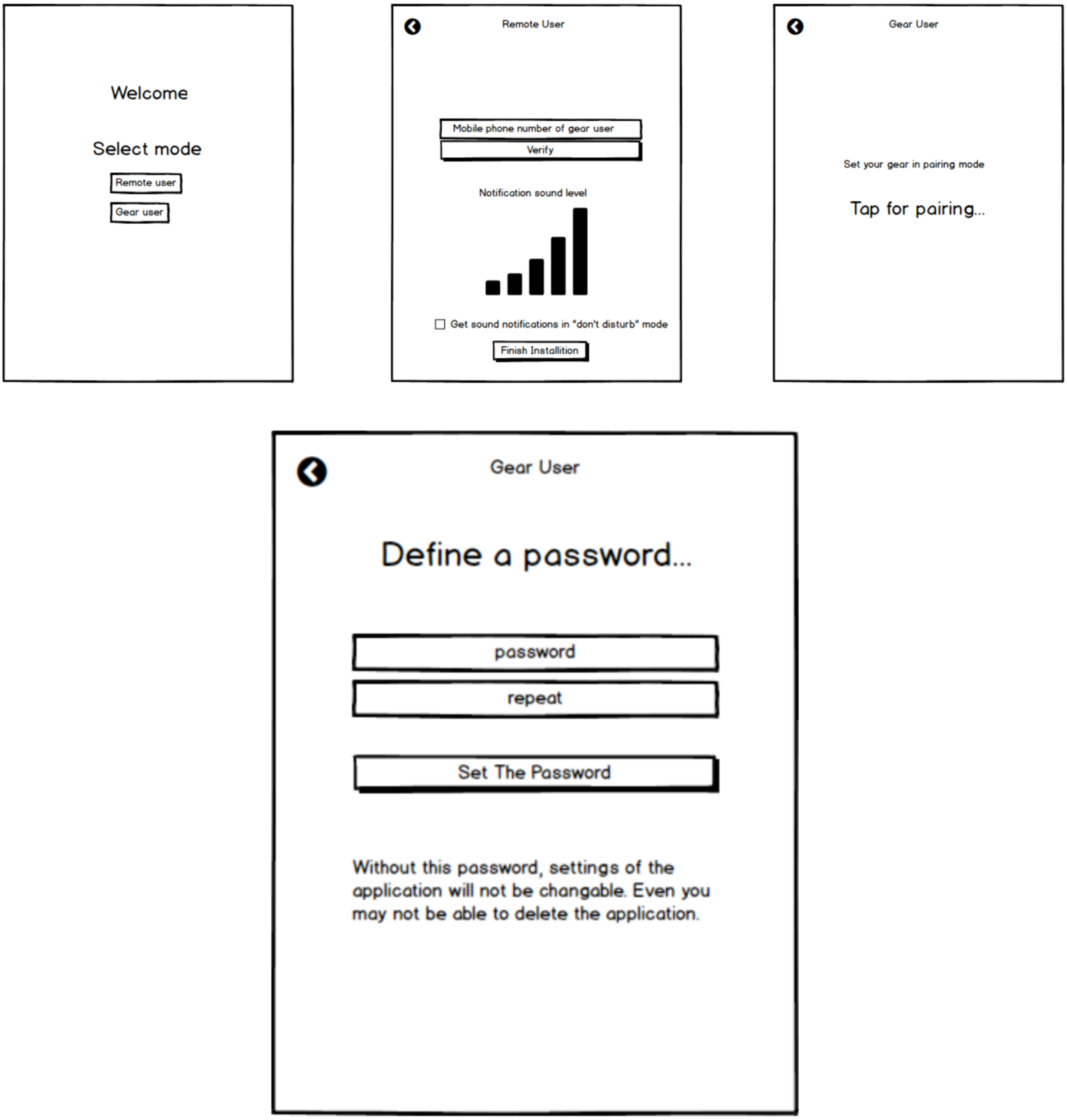

Remote User
Normal stress level
Connected


Gear User
Emergency
Connected

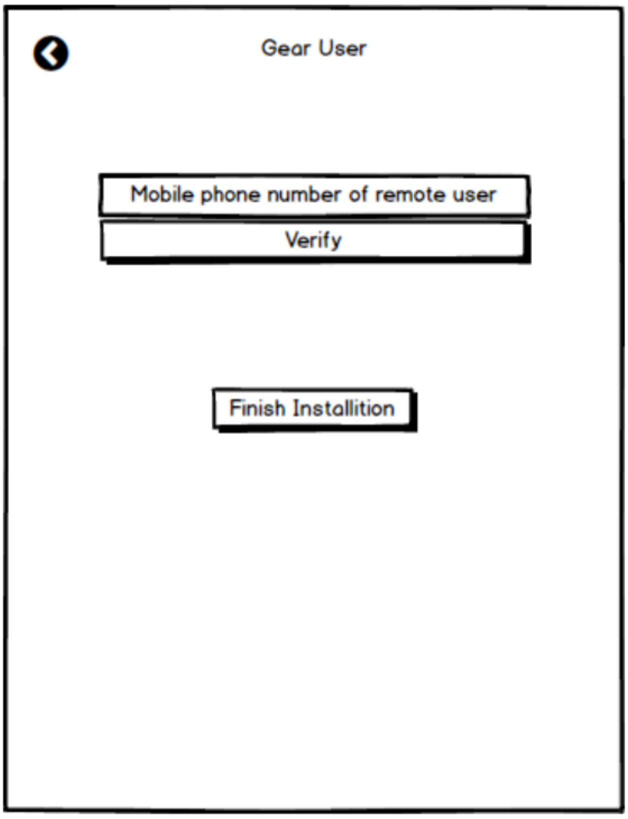
3. ReportStress 2 Second Mockup Muhammed Said Yeter

- This mockup is more simplistic.
- During sudden stress level changes, receiving emergency alerts is the only purpose.

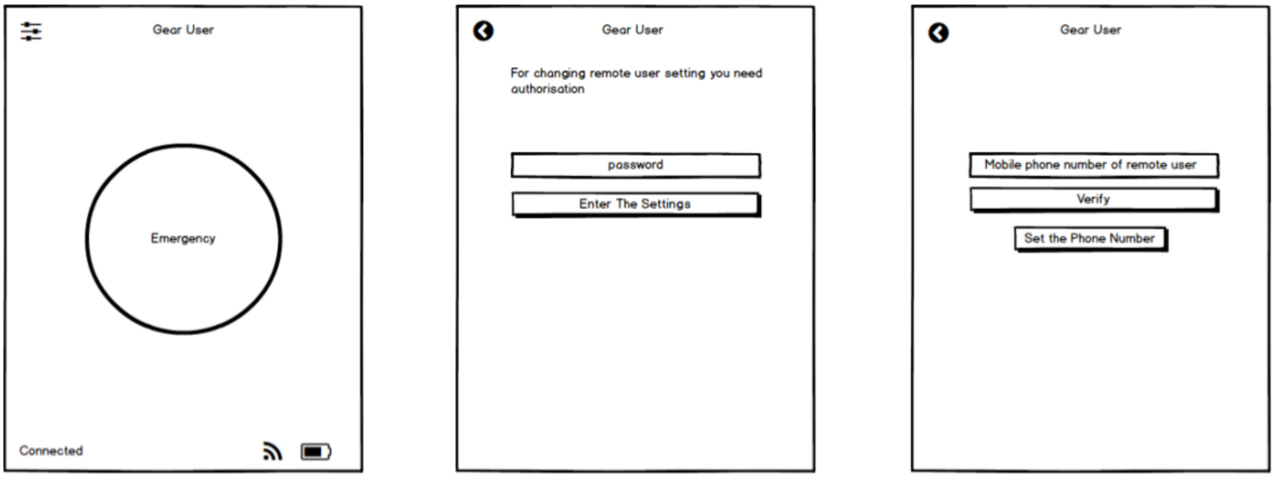
Login and Installation Screen



Without this password, settings of the application will not be changable. Even you may not be able to delete the application.

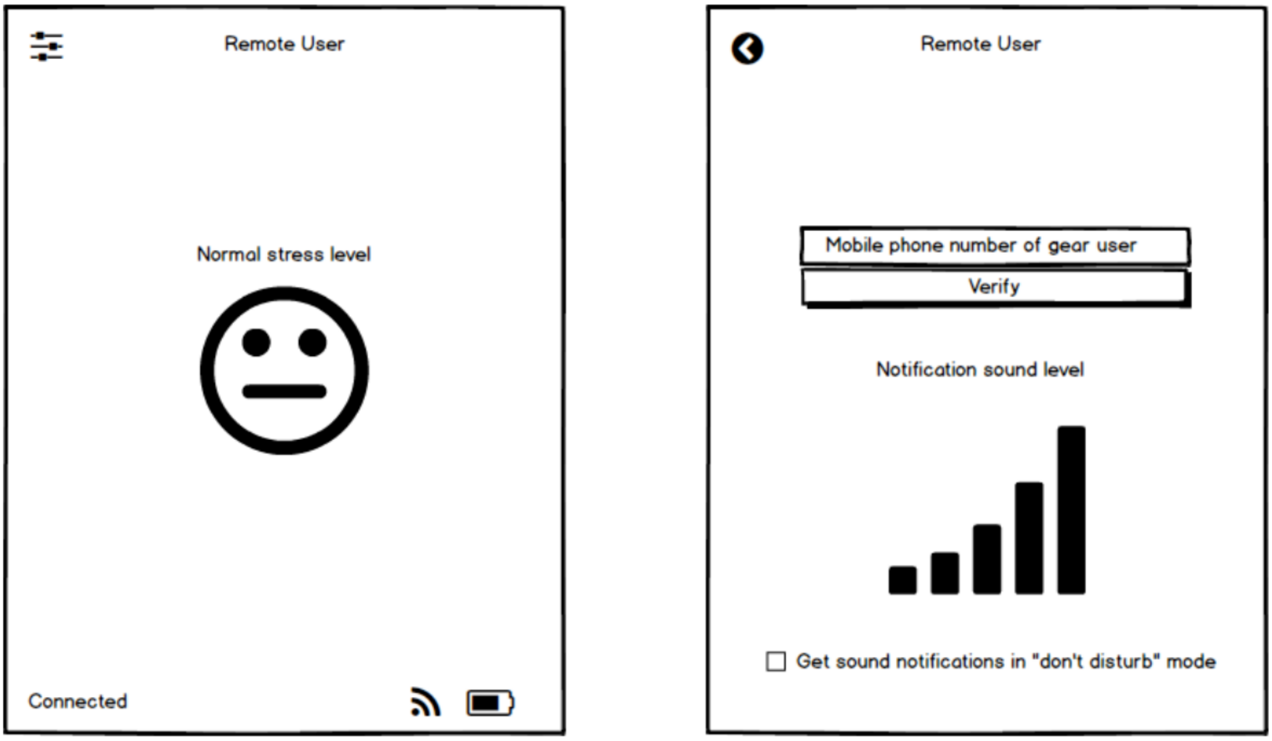

Gear User
Mobile phone number of remote user
Verify
Finish Installition

Settings for Gear User

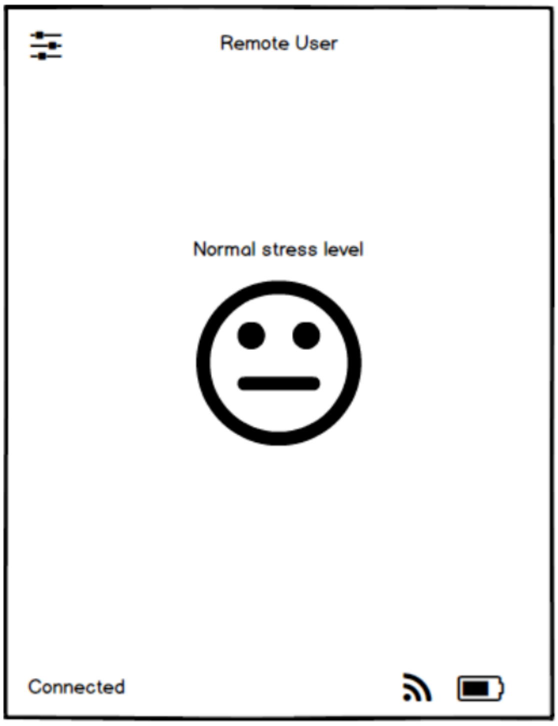


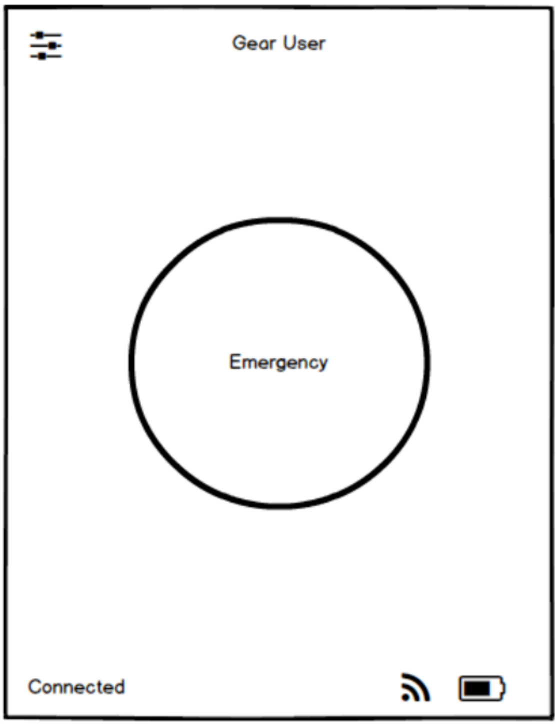
For changing remote user setting you need authorisation

Settings for Remote User


Remote User
Normal stress level
Connected
Remote User
Mobile phone number of gear user
Verify
Notification sound level
□ Get sound notifications in "don't disturb" mode

Home Screen


Remote User
Normal stress level
Connected


Gear User
Emergency
Connected

Ankara University, Engineering Faculty

ReSt

Stress Reporter

Assignment 3 Analytical Evaluation

Muhammed Said YETER

2022

Individual research project

saidyeter@gmail.com

Annotation

- In this presentation, all the features from installation to usage of the application will be explained.
- How the user types form the interface will be the main subject.

Contents

1. Introduction

1.1 Project Titles

- ReSt (Report Stress)
- Real time stress reporter for depended individuals.

1.2 Problem Description

- Parents generally want to follow their children’s situation when they are not standing by them. Getting sudden notifications when the children are under the risk of any kind of danger is one of parent’s needs. Nonetheless, getting statistical information of children’s mental stress will also help parent’s to learn presence of a problem that bothers children.

2. Evaluation of Muhammed Said Yeter' First Mockup

Heuristic	Severity	Comment
Visibility of system status	0	System status are clear. Instillation, settings and statistics parts are always followed with system status on the top of the screen.
Match between system and real world	1	In case of not knowing how to perceive graphs user might have difficulty to understand what visual graphics mean. Except this problem program is purged from technical terms so easy to understand.
User control and freedom	2	Instillation and settings parts have undo, redo and verifying options but in the gear user's home screen in case of tapping emergency there is nothing to do expect canceling phone call. And remote user don't have option for tuning stress level threshold when to get notification.
Consistency and standards	0	Icons are fit to world wide icon usage easy to understand.
Error prevention	1	Emergency button may cause some problem if the user tap the screen accidentally. Except this error application is not tend to make some critical mistakes.
Recognition better than recall	0	Icons which direct user are commonly used all over the application.
Flexibility and efficiency of use	1	In the gear user's settings and remote user's statistics parts, you have to click button back two times to turn back to the home screen. There might be a shortcut to turn back to the home screen.
Aesthetic and minimalist design	0	Since interface of the application is simple, there is not too much multifaceted aspects.
Help users recognize, diagnose, and recover from errors	2	There is no precaution for probable errors.
Help and documentation	2	Documentation is partially provided in the instillation and settings parts but there is not any tips.

Summary

Strength: Good to retrieving detailed information and analyzing this information.

Weakness: For less educated user it might be difficult to use.

3. Evaluation of Muhammed Said Yeter’ Second Mockup

Heuristic	Severity	Comment
Visibility of system status	0	System status are clear. Instillation and settings parts are always followed with system status on the top of the screen.
Match between system and real world	0	Program is purged from technical terms so easy to understand.
User control and freedom	3	Instillation and settings parts have undo, redo and verifying options but in the gear user’s home screen in case of tapping emergency there is nothing to do expect canceling phone call. And remote user don’t have option for tuning stress level threshold when to get notification. In addition, user wouldn’t be able to retrieve real time situation of the gear user.
Consistency and standards	0	Icons are fit to world wide icon usage easy to understand.
Error prevention	1	Emergency button may cause some problem if the user tap the screen accidentally. Except this error application is not tend to make some critical mistakes.
Recognition better than recall	0	Icons which direct user are commonly used all over the application.
Flexibility and efficiency of use	1	In the gear user’s settings, you have to click button back two times to turn back to the home screen. There might be a shortcut to turn back to the home screen.
Aesthetic and minimalist design	0	Since interface of the application is simple, there is not too much multifaceted aspects.
Help users recognize, diagnose, and recover from errors	2	There is no precaution for probable errors.
Help and documentation	2	Documentation is partially provided in the instillation and settings parts but there is not any tips.

Summary

Strength: Since having less option of use every type of the user can use it easily.

Weakness: Keeping track of real time information