

SERVICE 1 - SAVE IMAGE CONTROLLER

```
//TODO: Save image on local folder
```

The image will have a name like this `"id_number"`

The field `id` will be unique for each report

the file number will be a counter

The image controller tell us how many images will have the report

The report could have blank images |

[illegible][illegible]

SERVICE 2 - Envio Informe

Currently this service save a report in a sharepoint and send an email

JTODD: 1- recovery images

JTODD: 2- Build PDF

JTODD: 3- Get PDF in B64

After that the service have to do the same that already does

Share file in Microsoft Sharepoint and Send the mail

Before the report in B64 was send by the app

Now the report will be generate by the Service

[illegible]

```

using UnityEngine.Networking;
using UnityEngine.Networking.UnityWebRequest;

using UnityEngine.Networking.UnityWebRequest;

using UnityEngine;

namespace UnityEngine.Networking.UnityWebRequest
{
    public static class UnityWebRequest
    {
        public static UnityWebRequest Get(string url)
        {
            return new UnityWebRequest(url, "GET");
        }

        public static UnityWebRequest Post(string url, string data)
        {
            return new UnityWebRequest(url, "POST");
        }

        public static UnityWebRequest Put(string url, string data)
        {
            return new UnityWebRequest(url, "PUT");
        }

        public static UnityWebRequest Delete(string url)
        {
            return new UnityWebRequest(url, "DELETE");
        }

        public static UnityWebRequest Head(string url)
        {
            return new UnityWebRequest(url, "HEAD");
        }

        public static UnityWebRequest Options(string url)
        {
            return new UnityWebRequest(url, "OPTIONS");
        }

        public static UnityWebRequest Get(string url, string data)
        {
            return new UnityWebRequest(url, "GET");
        }

        public static UnityWebRequest Post(string url, string data)
        {
            return new UnityWebRequest(url, "POST");
        }

        public static UnityWebRequest Put(string url, string data)
        {
            return new UnityWebRequest(url, "PUT");
       }

        public static UnityWebRequest Delete(string url)
        {
            return new UnityWebRequest(url, "DELETE");
        }

        public static UnityWebRequest Head(string url)
        {
            return new UnityWebRequest(url, "HEAD");
        }

        public static UnityWebRequest Options(string url)
        {
            return new UnityWebRequest(url, "OPTIONS");
        }
    }
}

```

No Image Found

No Image Found