**Proyecto YET**

**Integrantes:**

**Cristian Camilo Arcila Delgado 1151152**

**María Alejandro Bohórquez 1151109**

**Cesar Alberto Guerrero 1151095**

**Universidad Francisco De Paula Santander**

**Ing. Sistemas**

**2017**

**Arquitectura Del Software**

**Profesor:**

**Ing. Milton Vera**

**Universidad Francisco De Paula Santander**

**Ing. Sistemas**

**2017**

**Introducción**

Este documento se encuentra toda la información con respecto al modelo de diseño de la aplicación (simulador de la aplicación yet de Gmail), diagramas, especificación de requerimientos, diseño de interfaces, arquitectura entre otros.

1. **Identificación de Roles y Responsabilidades**

|  |  |  |
| --- | --- | --- |
| **Rol** | **Responsabilidad** | **Responsable** |
| Arquitecto | Modelo De Diseño | Camilo Arcila |
| Programador | Desarrollar el Core y genera archivos JAR con toda la lógica requerida para el proyecto. | Cesar Guerrero |
| Programador | Desarrolla la GUI Web que usa los JAR anteriores | Alejandra Bohórquez |
| Pruebas | Aplicar las pruebas | Camilo Arcila |

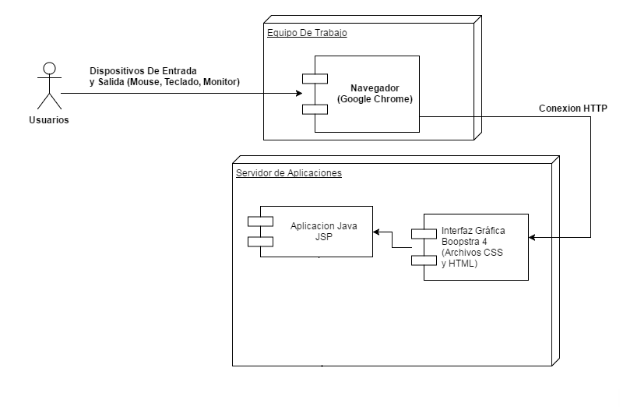
**2. Arquitectura:**

Este proyecto se realizó en base a una arquitectura MVC (Modelo Vista Controlador), El patrón de arquitectura MVC (Modelo Vista Controlador) es un patrón que define la organización independiente del Modelo (Objetos de Negocio), la Vista (interfaz con el usuario u otro sistema) y el Controlador (controlador del workflow de la aplicación).

El patrón de arquitectura "modelo vista controlador", es una filosofía de diseño de aplicaciones, y las razones por la cuales nos decidimos por usar esta arquitectura para el desarrollo de este proyecto es porque nos brinda:

Un modelo el cual contiene el núcleo de la funcionalidad (dominio) de la aplicación. Encapsula el estado de la aplicación. No sabe nada / independiente del Controlador y la Vista. Las Vistas son la presentación del Modelo. Puede acceder al Modelo pero nunca cambiar su estado. Puede ser notificada cuando hay un cambio de estado en el Modelo. Y por último está el controladorreacciona a la petición del Cliente, ejecutando la acción adecuada y creando el modelo pertinente. La principal ventaja de una aplicación distribuida bien diseñada es su buen escalado, es decir, que puede manejar muchas peticiones con el mismo rendimiento simplemente añadiendo más hardware. El crecimiento es casi lineal y no es necesario añadir más código para conseguir esta escalabilidad.

**3. Vista Lógica:**



**4. Identificación De Requerimientos Funcionales Y No Funcionales:**

**4.1 Requerimientos Funcionales:**

**RF1: Ingresar**

El sistema deberá solicitar al usuario un correo electrónico y una contraseña válida como requisito para empezar a utilizar la aplicación.

**RF2: Crear borrador**

El sistema debe permitir al usuario crear el borrador del mensaje a enviar usando las variables y siguiendo la sintaxis indicada por el sistema.

**RF3: Cargar Archivo Excel**

El sistema debe permitir analizar y validar que el archivo Excel contenga las variables referenciadas en el borrador creado con anterioridad.

**RF4: Enviar Correos**

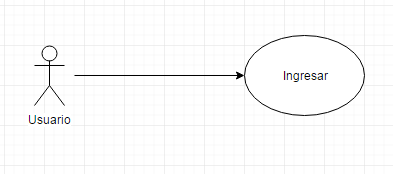
Después de tener ya el correo de borrador creado y el archivo de Excel haya sido analizado y cumpla con los requisitos, el sistema deberá permitir el envío masivo de correos electrónicos.

**4.2 Requerimiento No Funcionales:**

* El sistema debe brindar tiempos de respuesta óptimos.
* El sistema debe estar disponible en todo momento.
* El sistema debe poder soportar cambios, en pro de mejorar.
* El sistema debe tener una interfaz amigable con el usuario.

**5. Diagramas De Casos De Usos:**

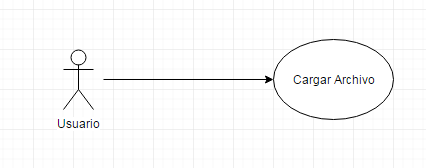
**5.1 Diagrama Caso De Uso Ingresar:**



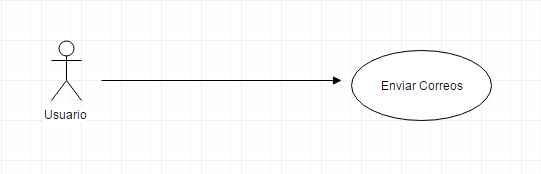
**5.2 Diagrama Caso De Uso Crear Borrador:**



**5.3 Diagrama Caso De Uso Cargar Archivo:**



**5.4 Diagrama Caso De Uso Enviar Correo:**



**6. Especificación de Casos De Uso:**

**6.1 Caso De Uso Ingresar:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Ingresar** | | | |
| **Descripción** | | **El sistema debe permitir al usuario ingresar un correo y una contraseña, necesarias para el envío de correos** | |
| **Actores** | | **Usuario** | |
| **Precondición** | | **Correo y Contraseña Válidas** | |
| **Entradas** | | * **Correo** * **Contraseña** | |
| **No** | **Actor** | **● Actividades** | **Escenario alterno** |
| **1** | **Usuario** | **El usuario ingresa su correo** |  |
| **2** | **Usuario** | **El usuario ingresa su contraseña** |  |
| **3** | **Usuario** | **El usuario da clic en ingresar** |  |
| **4** | **Sistema** | **El sistema verifica que sean un correo y contraseña válidos y procede a almacenar estos datos para su uso en el proceso de envío de correos** |  |
| **6** | **Sistema** | **Muestra la vista para la creación del mensaje borrador** |  |

**6.2 Caso De Uso Crear Borrador:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Crear Borrador** | | | |
| **Descripción** | | **El sistema debe permitir al usuario crear un borrador del mensaje indicando las variables y siguiendo la sintaxis indicada.** | |
| **Actores** | | **Usuario** | |
| **Precondición** | | **Mensaje sigla la sintaxis indicada.** | |
| **Entradas** | | * **Mensaje** | |
| **No** | **Actor** | **● Actividades** | **Escenario alterno** |
| **1** | **Usuario** | **El usuario ingresa su mensaje** |  |
| **2** | **Usuario** | **El usuario da clic en guardar mensaje** |  |
| **4** | **Sistema** | **El sistema verifica que el mensaje sea válido y procede a almacenar estos datos para su uso en el proceso de envío de correos** |  |
| **6** | **Sistema** | **Muestra la vista para subir el archivo** |  |

**6.3 Caso De Uso Subir Archivo:**

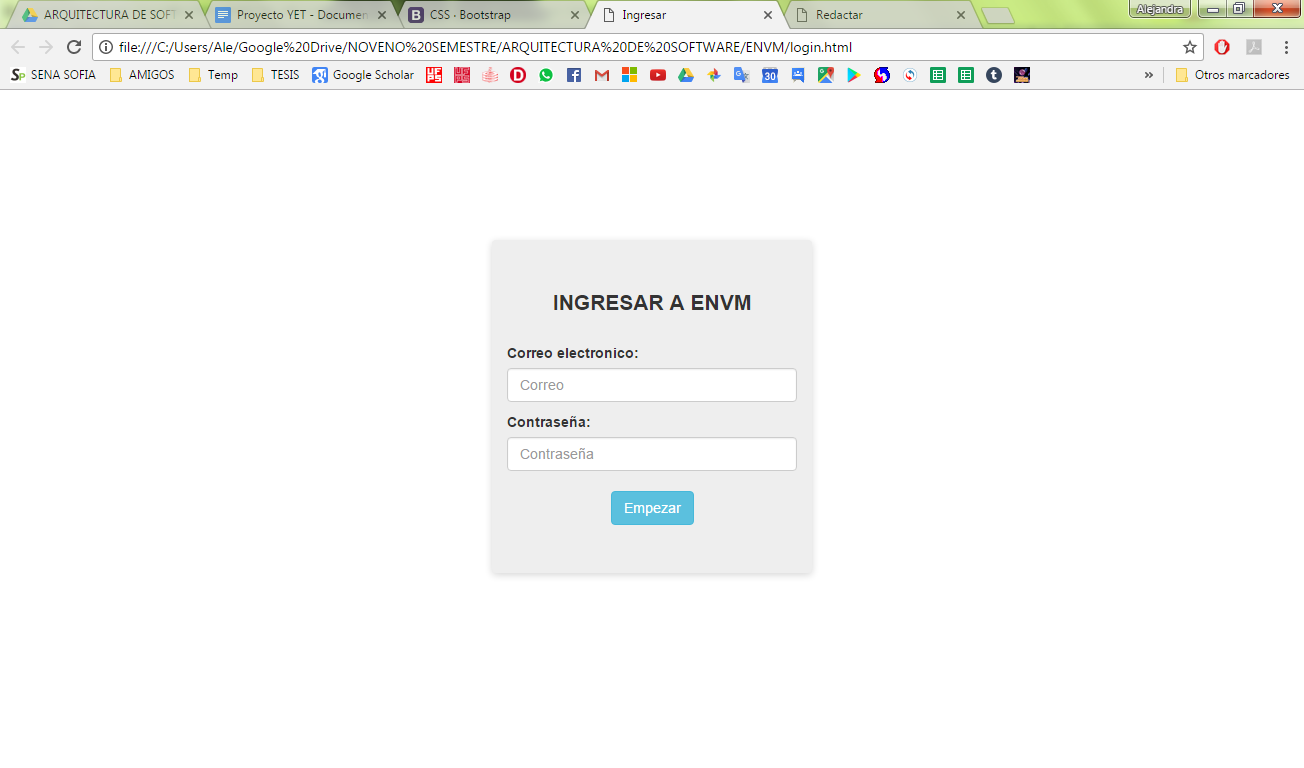
|  |  |  |  |
| --- | --- | --- | --- |
| **Cargar Archivo** | | | |
| **Descripción** | | **El sistema debe permitir al usuario subir un archivo de Excel y se debe validar que esté tenga las variables indicadas en el mensaje de borrador.** | |
| **Actores** | | **Usuario** | |
| **Precondición** | | **Archivo contenga las variables indicadas en el mensaje de borrador** | |
| **Entradas** | | * **Archivo** | |
| **No** | **Actor** | **● Actividades** | **Escenario alterno** |
| **1** | **Usuario** | **El usuario selecciona el archivo que desea cargar** |  |
| **2** | **Usuario** | **El usuario da clic subir archivos** |  |
| **4** | **Sistema** | **El sistema verifica que el archivo cumpla con los requisitos y si tenga con las variables necesarias para seguir el proceso.** |  |
| **6** | **Sistema** | **Muestra un mensaje de confirmación al subirse el archivo** |  |

**6.4 Caso De Uso Enviar Mensajes:**

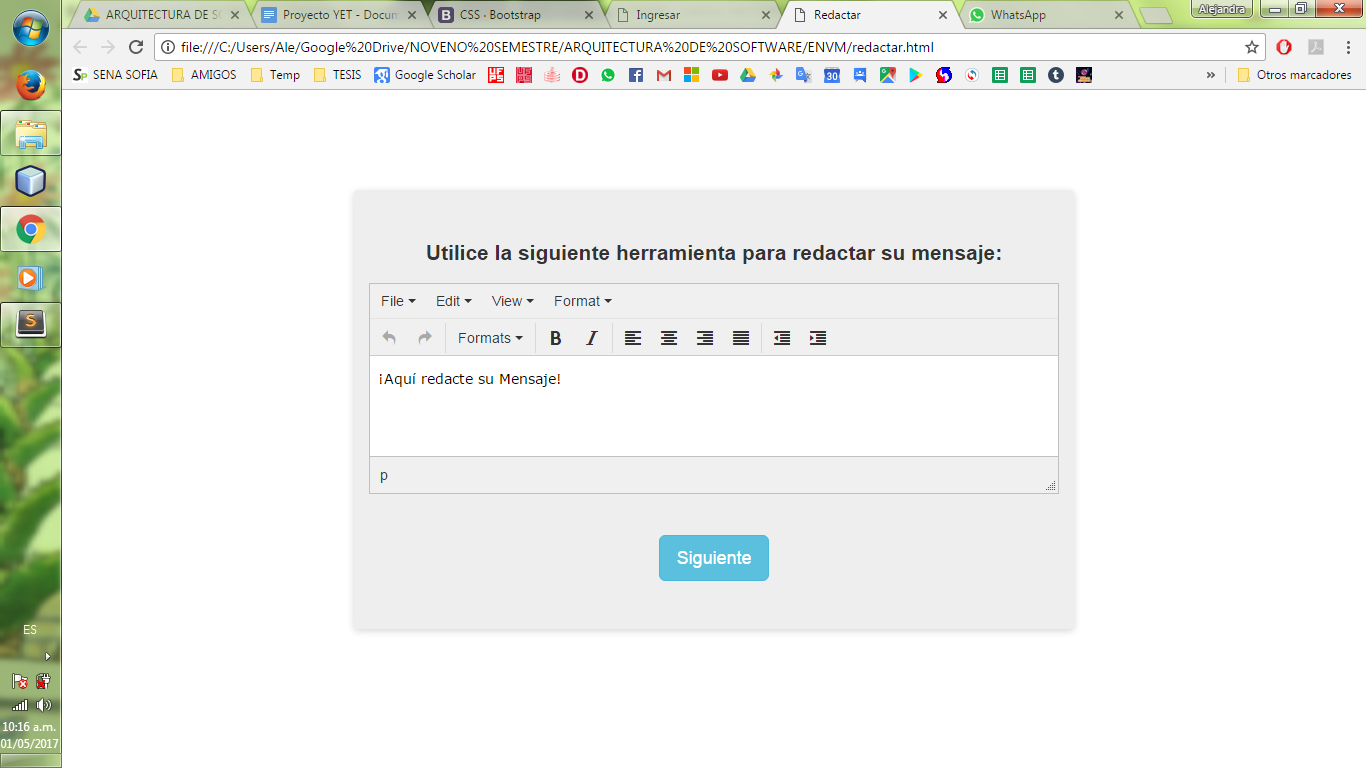
|  |  |  |  |
| --- | --- | --- | --- |
| **Enviar Mensajes** | | | |
| **Descripción** | | **El sistema debe permitir al usuario el envío masivo de correos** | |
| **Actores** | | **Usuario** | |
| **Precondición** | | **Ninguna** | |
| **Entradas** | | **Ninguna** | |
| **No** | **Actor** | **● Actividades** | **Escenario alterno** |
| **1** | **Usuario** | **El usuario Da clic en “Enviar Correos”** |  |
| **4** | **Sistema** | **El sistema realiza el envío de correos masivos a los diferentes correos capturados del archivo Excel** |  |
| **6** | **Sistema** | **Muestra mensaje de confirmación de proceso.** |  |

**7. Vistas:**

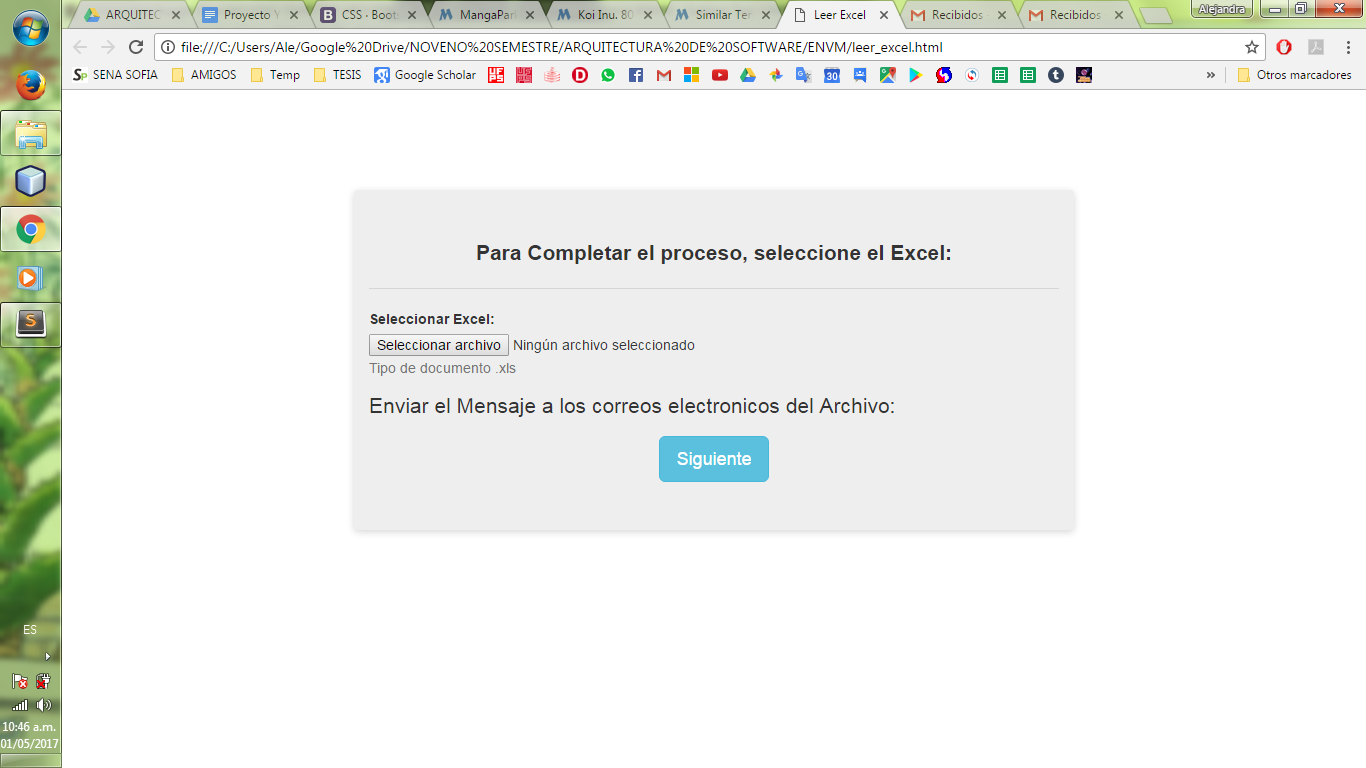
**Ingresar:** El usuario ingresa un correo y una contraseña válidos para iniciar a utilizar la aplicación.



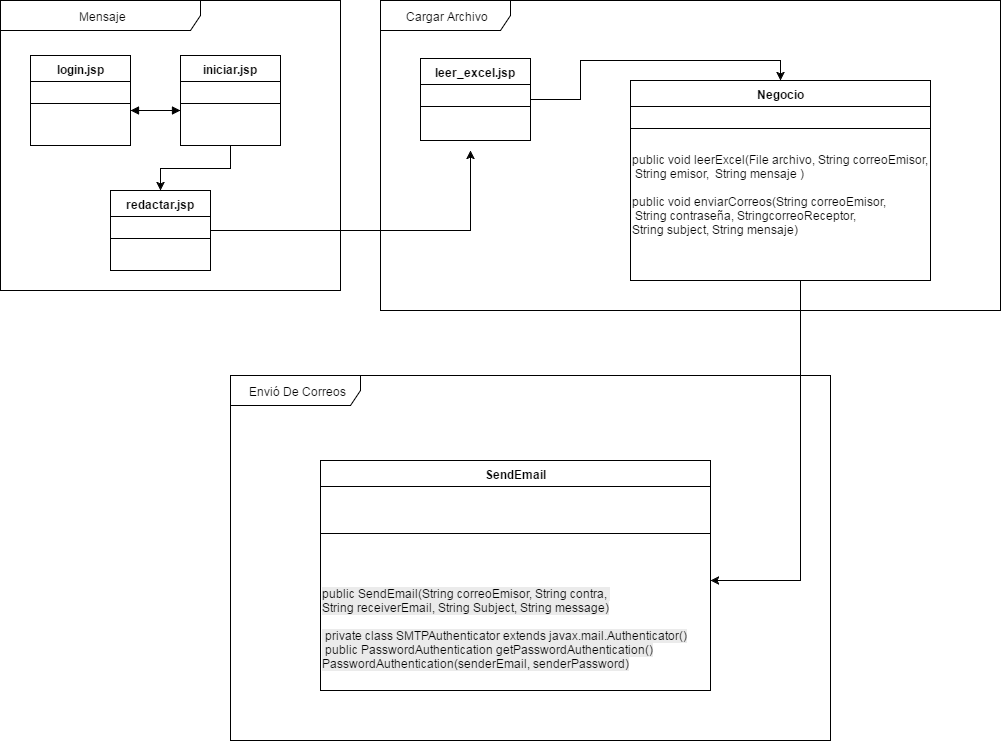
**Crear Borrador:** Después del ingreso el usuario debe crear el mensaje de borrador.



**Cargar Excel Y Enviar Correos:** El usuario selecciona el archivo Excel y realiza el envío masivo de correos



**8. Diagrama De Componentes:**



**9. Pruebas:**

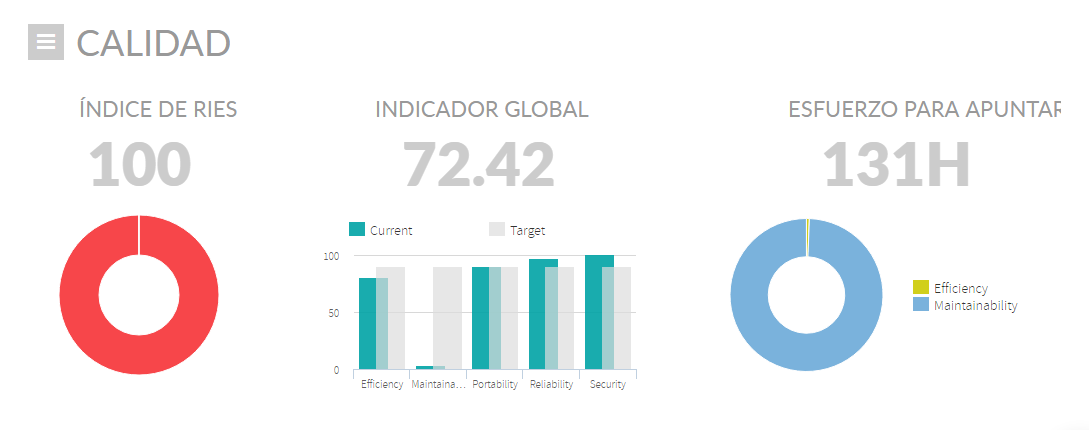
Las pruebas se realizaron utilizando la aplicación kiuwan ya utilizada y analizada en este curso:

A primera instancia nos arroja una evaluación de seguridad media, con una sola vulnerabilidad normal, 1829 líneas de código y 17 archivos analizados.

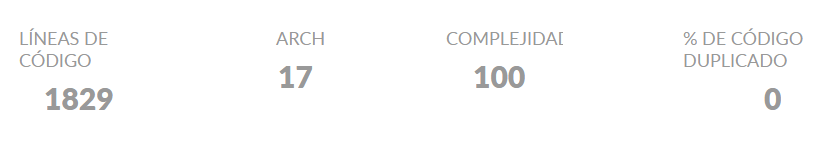




En calidad los índices no son tan alentadores, empezando porque el índice de riesgo es de un cien por ciento



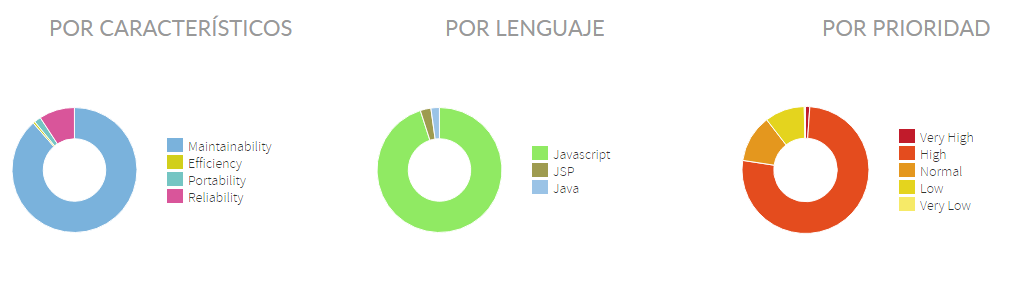
Un cien por ciento de complejidad, es importante analizar que no hay nada de código duplicado siendo esté un factor muy recurrente.



 En el análisis de archivos principales encontramos:



Y sus principales defectos son:





Link Sandbox:

[http://sandbox1.ufps.edu.co:8080/ufps\_35-ENVM/](https://l.facebook.com/l.php?u=http%3A%2F%2Fsandbox1.ufps.edu.co%3A8080%2Fufps_35-ENVM%2F&h=ATMU2V29bTgwoVmij-rFvI-p7MMTOadPxchf8oTWKWqPNGi-hZqJWQ6m8YWmkIAq7RHCvMkOldowzaqz4-fO7PTwH71DJUGZVQHzsVfS2kAbw-5JqpS5SDhpIDnfhoNgBVumrAU)

Link Git:

[https://github.com/Kingmaq12/ENVM](https://l.facebook.com/l.php?u=https%3A%2F%2Fgithub.com%2FKingmaq12%2FENVM&h=ATMU2V29bTgwoVmij-rFvI-p7MMTOadPxchf8oTWKWqPNGi-hZqJWQ6m8YWmkIAq7RHCvMkOldowzaqz4-fO7PTwH71DJUGZVQHzsVfS2kAbw-5JqpS5SDhpIDnfhoNgBVumrAU)