****

**Facultad de Ingeniería Informática**

Modelos para el estudio de las variables que más influyen en la estimación de la legitimidad en el modelo probabilístico de autenticación propuesto por Legón.

Informe de Gestión de proyecto

**Autor:** Estrella Cárdenas Blanco ([estrellacarden](hserra@ceis.cujae.edu.cu)asblanco@gmail.com)

**Tutor:** Dr. C. Raisa Socorro Llanes (<raisa@ceis.cujae.edu.cu>)

**Curso:** 2022-2023

**Rol: Especialista de Calidad- Probador**

**Especialista de Calidad**

**Cantidad de defectos por revisión de los diferentes artefactos entregados en la asignatura.**

|  |  |
| --- | --- |
|  | Total de defectos detectados por revisión |
| Plan de proyecto | 4 |
| Análisis de Riesgos | 1 |
| Informe de Rol Analista-Arquitecto-Diseñador | 1 |
| Total | 6 |

Defectos detectados en el plan de proyecto en la revisión 1: Tipo 10 (Documentación)

* No existe la tarea: Estudiar el área de proceso de Gestión de Configuración de CMMi.
* No se encuentran identificados en el plan los elementos que estarán sujetos a la gestión de la configuración.
* No existe la tarea: Elaborar convenio para nombrar los elementos de configuración.
* No existe la tarea: Definir la política utilizada para crear y gestionar ramas.

Defectos detectados en el análisis de riesgos en la revisión 1: Tipo 10 (Documentación)

* Los riesgos de criticidad alta y muy alta deben tener tareas de contingencia

Defectos detectados en el informe de rol Analista-Arquitecto-Diseñador en la revisión 1: Tipo 10 (Documentación)

* No se identifica en los requisitos funcionales, cuál o cuáles serán implementados en la asignatura.

**Diferencias entre Plan y Real, de la variable tiempo para las tareas en el plan del proyecto.**

|  |  |  |
| --- | --- | --- |
| **Métrica** | **Tiempo Estimado (Días)** | **Tiempo Real (Días)** |
| Planificador | 4 días | 2 días |
| Probador | 5 días | 2 días |
| Jefe de Proyecto | 5 días | 2 días |
| Analista - Diseñador | 8 días | 5 días |
| Gestor de la configuración | 11 días | 7 días |
| Programador | 18 días | 12 días |

**Buenas prácticas de metodologías propuesta por PSP aplicadas al proyecto**

Se registró tiempo, defectos y tamaño.

Se elaboró y utilizo un estándar de codificación

Se ejecutaron, diseñaron ye ejecutaron las pruebas

Se planificaron tareas a realizar dentro de un cronograma (Diagrama de Gantt).

**Buenas prácticas de metodologías ágiles aplicadas al proyecto**

**Un diseño simple:** se realizó siempre lo mínimo imprescindible de la forma más sencilla posible.

**Metáforas:** Se buscó nombres de las clases, variables y métodos que definieran cómo funciona el programa.

**Uso de estándares de codificación:** Se usó un estándar de codificación.

**Refactorizar:** se utilizó refactorización en el código algunos de los métodos son Extract Class y Rename Method.

**PROBADOR**

## Pruebas funcionales

Las pruebas de caja negra, también conocidas como pruebas de comportamiento o pruebas funcionales, tienen como objetivo chequear que los sistemas desarrollados cumplan con sus funcionalidades. Se enfocan solamente en la entrada y la salida del sistema sin tener en cuenta la estructura interna del software. Es recomendable ejecutar los escenarios con entradas válidas y fallidas.

Las pruebas de caja blanca, también conocidas como pruebas de cristal o pruebas estructurales, es una técnica que evalúa el código y la estructura interna del programa.

### Casos de prueba

Con el objetivo de comprobar el correcto funcionamiento del sistema de software se proponen casos de pruebas guiados a los casos de uso.

Crear autenticación: Verificar que el usuario existe en la base de datos para entrar al sistema.

Exportar Reportes: Verificar que se muestren correctamente los reportes

Gestionar Imágenes: Verificar que se inserte, modifique y elimine las imágenes.

Gestionar Usuarios: Verificar que se inserte, modifique y elimine los usuarios.

A continuación se muestras las pruebas de caja negras realizadas

Tabla 8 Caso de Prueba checkUser

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Historia de Usuario** | | | checkUser | | | |
| **Desarrollador** | | | Estrella Cárdenas | | | |
| **Probador** | | | Estrella Cárdenas | | | |
| **Fecha** | | | **29/10/2019** | | | |
| **Versión** | | | **1.0** | | | |
| **Objetivo de la Prueba** | | | Comprobar que funcione correctamente | | | |
| **Descripción de la prueba** | | | Permite verificar cuando un usuario se va autenticar si existe o no en la base de datos. Si existe el usuario puede entrar al sistema. | | | |
| **Condiciones** | | | Existir en la base de datos | | | |
| **Combinaciones de valores de entrada** | | | | | **Resultados Esperados** | **Resultados Reales** |
| **CP** | **escenario** | **nombre variable de entrada** | | **valor** |
| 1 checkUser | | username | | star | Devuelve un objeto UserDto ya que el usuario star existe en la base de datos y sus atributos son:  Userid=1  Identification = 00060566498  Gender=femenino  Profid=1  Username= star  Fullname= estrella  Registered=1  Roleid=1  Email= estrellacardenas@gmail.com | Devuelve un objeto UserDto ya que el usuario star existe en la base de datos y sus atributos son:  Userid=1  Identification = 00060566498  Gender=femenino  Profid=1  Username= star  Fullname= estrella  Registered=1  Roleid=1  Email= estrellacardenas@gmail.com |
| 2 | checkUser | username | | juan | Devuelve un objeto UserDto = null ya que el usuario juan no existe en la base de datos | Devuelve un objeto UserDto = null ya que el usuario juan no existe en la base de datos |
|  | | | | | | |

Tabla 9 Caso de Prueba report1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Historia de Usuario** | | | Report1 | | | |
| **Desarrollador** | | | Estrella Cárdenas | | | |
| **Probador** | | | Estrella Cárdenas | | | |
| **Fecha** | | | **29/10/2019** | | | |
| **Versión** | | | **1.0** | | | |
| **Objetivo de la Prueba** | | | Comprobar que funcione correctamente | | | |
| **Descripción de la prueba** | | | Resumen de los intentos de autenticación de todos los usuarios según el tipo en un día dado. | | | |
| **Condiciones** | | | Existir autenticaciones en la base de datos | | | |
| **Combinaciones de valores de entrada** | | | | | **Resultados Esperados** | **Resultados Reales** |
| **CP** | **escenario** | **nombre variable de entrada** | | **valor** |
| 1 report1 | | date | | 2020-05-07 | Devuelve una lista de los intentos de autenticación realizados en la fecha 2020-05-07. | Devuelve una lista de los intentos de autenticación realizados en la fecha 2020-05-07. |
| 2 | report1 | date | | 2020-02-08 | Devuelve null ya que en esta fecha no se realizaron intentos de autenticación | Devuelve null ya que en esta fecha no se realizaron intentos de autenticación |

Tabla 10 Caso de prueba report2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Historia de Usuario** | | | Report2 | | | |
| **Desarrollador** | | | Estrella Cárdenas | | | |
| **Probador** | | | Estrella Cárdenas | | | |
| **Fecha** | | | **29/10/2019** | | | |
| **Versión** | | | **1.0** | | | |
| **Objetivo de la Prueba** | | | Comprobar que funcione correctamente | | | |
| **Descripción de la prueba** | | | Resumen de los intentos de autenticación de un usuario según el tipo por mes. | | | |
| **Condiciones** | | | Existir autenticaciones en la base de datos | | | |
| **Combinaciones de valores de entrada** | | | | | **Resultados Esperados** | **Resultados Reales** |
| **CP** | **escenario** | **nombre variable de entrada** | | **valor** |
| 1 report2 | | username | | star | Devuelve una lista de los intentos de autenticación del usuario star | Devuelve una lista de los intentos de autenticación del usuario star |
| 2 | report2 | username | | alexa | Devuelve null ya que el usuario alexa no está en la base de datos | Devuelve null ya que el usuario alexa no está en la base de datos |

Tabla 11 Caso de prueba report3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Historia de Usuario** | | | Report3 | | | |
| **Desarrollador** | | | Estrella Cárdenas | | | |
| **Probador** | | | Estrella Cárdenas | | | |
| **Fecha** | | | **29/10/2019** | | | |
| **Versión** | | | **1.0** | | | |
| **Objetivo de la Prueba** | | | Comprobar que funcione correctamente | | | |
| **Descripción de la prueba** | | | Resumen de los intentos de autenticación ilegítimas de ese usuario. | | | |
| **Condiciones** | | | Existir autenticaciones en la base de datos | | | |
| **Combinaciones de valores de entrada** | | | | | **Resultados Esperados** | **Resultados Reales** |
| **CP** | **escenario** | **nombre variable de entrada** | | **valor** |
| 1 report3 | | username | | star | Devuelve una lista de intentos de autenticación ilegítimos del usuario star | Devuelve una lista de intentos de autenticación ilegítimos del usuario star |
| 2 | report3 | username | | pepe | Devuelve null ya que el usuario pepe no está en la base de datos | Devuelve null ya que el usuario pepe no está en la base de datos |

Tabla 12 Caso de prueba report4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Historia de Usuario** | | | Report4 | | | |
| **Desarrollador** | | | Estrella Cárdenas | | | |
| **Probador** | | | Estrella Cárdenas | | | |
| **Fecha** | | | **29/10/2019** | | | |
| **Versión** | | | **1.0** | | | |
| **Objetivo de la Prueba** | | | Comprobar que funcione correctamente | | | |
| **Descripción de la prueba** | | | Resumen de las imágenes usadas en los intentos de autenticación clasificadas dudosas legítimas ordenadas por la frecuencia de uso de un usuario dado. | | | |
| **Condiciones** | | | Existir autenticaciones en la base de datos | | | |
| **Combinaciones de valores de entrada** | | | | | **Resultados Esperados** | **Resultados Reales** |
| **CP** | **escenario** | **nombre variable de entrada** | | **valor** |
| 1 report4 | | username | | star | Devuelve un listado de las imágenes usadas en los intentos de autenticación dudosos legitimo del usuario star | Devuelve un listado de las imágenes usadas en los intentos de autenticación dudosos legitimo del usuario star |
| 2 | Report4 | username | | pipo | Devuelve null ya que el usuario pipo no está en la base de datos | Devuelve null ya que el usuario pipo no está en la base de datos |

Tabla 13 Caso de prueba report5

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Historia de Usuario** | | | Report5 | | | |
| **Desarrollador** | | | Estrella Cárdenas | | | |
| **Probador** | | | Estrella Cárdenas | | | |
| **Fecha** | | | **29/10/2019** | | | |
| **Versión** | | | **1.0** | | | |
| **Objetivo de la Prueba** | | | Comprobar que funcione correctamente | | | |
| **Descripción de la prueba** | | | Resumen de los intentos de autenticación en un rango de fecha dado | | | |
| **Condiciones** | | | Existir autenticaciones en la base de datos | | | |
| **Combinaciones de valores de entrada** | | | | | **Resultados Esperados** | **Resultados Reales** |
| **CP** | **escenario** | **nombre variable de entrada** | | **valor** |
| 1 report5 | | dateIni =  dateFin = | | 2020-11-28  2021-10-18 | Devuelve un listado de todos los intentos de autenticación realizados en el rango de fecha 2020-11-28 al 2021-10-18 | Devuelve un listado de todos los intentos de autenticación realizados en el rango de fecha 2020-11-28 al 2021-10-18 |
| 2 | report5 | dateIni =  dateFin = | | 2021-10-20  2021-10-28 | Devuelve null ya entre 2021-10-20 y 2021-10-28 no existen ningún intento de autenticación realizados | Devuelve null ya entre 2021-10-20 y 2021-10-28 no existen ningún intento de autenticación realizados |

A continuación se muestra las pruebas de caja blanca realizadas

Tabla 14 Caso de prueba de caja blanca

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Desarrollador** | | | Estrella Cárdenas | | | |
| **Probador** | | | Estrella Cárdenas | | | |
| **Fecha** | | | 29/10/2019 | | | |
| **Versión** | | | 1.0 | | | |
| **Objetivo de la Prueba** | | | Probar que los métodos funcionen correctamente | | | |
| **Combinaciones de valores de entrada** | | | | | **Resultados Esperados** | **Resultados Reales** |
| **CP** | **escenario** | **nombre variable de entrada** | | **valor** |
| 1 | createPicture | Date  Id  Name  Quality  url | | 2021-10-29  1  o.jpg  alta  images/o-6061015493021598988.png | Devuelve un String= Picture Created | Devuelve un String= Picture Created |
| 2 | deletePicture | Id | | 23 | Devuelve un String= Picture Deleted | Devuelve un String= Picture Deleted |
| 3 | deletePicture | id | | 30 | Devuelve un null ya que no existe el id=30 en la base de datos | Devuelve un null ya que no existe el id=30 en la base de datos |
| 4 | updatePicture | Date  Id  Name  Quality  url | | 2021-10-29  1  a.jpg  alta  images/o-6061015493021598988.png | Devuelve un String= Picture Update | Devuelve un String= Picture Update |
| 5 | createUser | Userid  Identification  Gender  Profid  Username  Fullname  Registered  Roleid  Email | | 34  04041655367  Masculino  12  Alexander  Alexander  0  2  alexandermedero@gmail.com | Devuelve un String= User Created | Devuelve un String= User Created |
| 6 | deleteUser | id | | 3 | Devuelve un String= User Deleted | Devuelve un String= User Deleted |
| 7 | deleteUser | id | | 100 | Devuelve un null ya que no existe el id=100 en la base de datos | Devuelve un null ya que no existe el id=100 en la base de datos |
| 8 | updateUser | Userid  Identification  Gender  Profid  Username  Fullname  Registered  Roleid  Email | | 1  00060566498  femenino  1  star  estrella  1  1  estrellacardenasblanco@gmail.com | Devuelve un String= User Update | Devuelve un String= User Update |