BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN



Proyectos I + D 1

Definición de pruebas

**CABI: Control de acceso de bicicletas**

Docente: Dr. Juan Manuel Gonzáles Calleros

Herbert Joadan Romero Villarreal

Víctor Manuel Hernández Ramírez

Roberto Castillo Medina

Eduardo Tolentino Cortés

Luis Gerardo Alvarado Fuentes

David Avelino Serrano

Ricardo Altamirano Cabrera

José Luis González Victoria

Estefania Cotzomi Navarro

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fecha | Versión | Descripción | Fase | Responsable |
| 01/10/18 | 1.0 | Se definen las pruebas en términos generales | Fase 1 | David Avelino Serrano |
| 02/10/18 | 1.1 | Se definen las pruebas unitarias para backend y frontend y las pruebas de integración como evaluación de heurísticas | Fase 2 y 3 | Herbert Joadan Romero Villarreal |

Historial de cambios

Introducción

Objetivos

Este documento tiene como finalidad entregar los pasos a seguir para la aplicación correcta de las estrategias y pruebas necesarias en el sistema “CABI”. Con el fin de verificar las funciones y procesos de los distintos módulos del software, así como también encontrar los posibles fallos o errores que se presenten durante el periodo de pruebas. Además validar si el sistema cumple con los requerimientos que contemplen el funcionamiento total del mismo.

Estrategia de pruebas

A través de las diferentes historias de usuario que se han realizado, se pretende retomar información directamente relacionada con las pruebas, para asegurar la calidad de estas y del producto.

Se elegirán los requerimientos de acuerdo a su fecha de entrega para poder aplicar las pruebas correspondientes.

Alcance

El equipo de trabajo pretende realizar las pruebas, de manera ascendente, por modulo, para ello se realizaran pruebas unitarias y de integración.

Modulo A

Modulo B

Modulo C

Fases del proyecto

|  |  |
| --- | --- |
| Fases | Descripción |
| 1 | Acceso de administrador al sistema |
| 2 | Registro de usuarios en el sistema |
| 3 | Modificar datos de usuario |
| 4 | Reflejar datos de usuario en aplicación móvil |
| 5 | Validación y registro de entradas y salidas |
| 6 | Registro de visitantes |
| 7 | Reportar discrepancias |

Artefactos de Prueba

Módulos de la aplicación

Módulo

Es un conjunto de elementos, el cual tienen en común la finalidad.

Pruebas

Tipo de pruebas que serán utilizadas dentro del componente.

Descripción

Breve explicación sobre la prueba a realizar

Documentos a entregar

Los documentos a entregar con la aplicación son:

1. Manual de Usuario: Contiene de forma organizada las instrucciones de la aplicación sin problemas.
2. Manual Técnico: Contiene las especificaciones técnicas de la aplicación, así como descripción de como están elaborados los módulos.
3. 3. Manual de Pruebas: Incluye los diferentes tipos de pruebas que se realizan para la validación del sistema.

Nota: Se incluye además link a repositorio de GitHub el cual contiene el desarrollo del sistema.

<https://github.com/MrcRjs/CABI-web> - Versión Angular Web

<https://github.com/RicardoAltamirano/biciBUAP.git> - Versión PHP, Javascript, Ajax Web

<https://github.com/herbertRV/CABI-android> - Versión móvil Android

<https://github.com/victorrmrz/CABI-ios.git> - Versión móvil iOS

El propósito de este artefacto es:

* Proporcionar una manera de capturar entradas para las pruebas, condiciones y resultados esperados para un sistema.
* Identificar sistemáticamente aspectos del software a probar
* Especificar si un resultado esperado ha sido alcanzado basado en la verificación de un requerimiento, conjunto e requerimientos o escenarios.

|  |  |  |
| --- | --- | --- |
| Modulo | Pruebas | Descripción |
| GUI | Facilidad de uso  Look & Feel | La facilidad de uso consiste en que siempre tengan el conocimiento sobre qué pueden o qué deberían hacer los usuarios en cada momento y cómo hacerlo.  Look & feel es la apariencia que se proporciona al usuario |
| Lógica de negocio | Funcionalidad | El sistema debe poder realizar todos los requerimientos establecidos con el cliente, este módulo será guiado por los diferentes 0pos de requerimientos que se han manejado durante el proyecto |
| DAO | Persistencia | El sistema debe ser capaz de guardar datos para ser usados en otro momento, además de tener acceso a ellos sin tener ningún problema de consistencia e integridad. |
| No Funcionales | No Funcionales | El sistema debe cumplir con los requerimientos no funcionales que se han especificado en el SRS teniendo en cuenta el diseño. |

Módulos por probar

Características por probar

|  |  |  |
| --- | --- | --- |
| Característica | Descripción | Módulo |
| Requerimientos funcionales | Se debe tener en cuenta el criterio de aceptación y dependencias, para realizar pruebas en los módulos. Además, se debe utilizar el documento de diagrama de secuencia para tener claro los casos de éxito y fallo, y si la herramienta cumple con ellos. | Los Módulos donde se puede probar esta característica son:  Lógica de negocio  DAO |
| Requerimientos No Funcionales | Se debe tener en cuenta el criterio de aceptación y lo que exige el requerimiento para su cumplimiento, especificado en el documento SRS. | Los Módulos donde se puede probar esta característica son:  No funcionales |

Tipos de Pruebas

Pruebas Unitarias

Pruebas por cada unidad, en este caso una unidad es equivalente a un requerimiento. El requerimiento es probado y aprobado si este cumple con lo que está escrito en la especificación de requerimientos.

|  |  |
| --- | --- |
| NOMBRE Prueba unitaria IDENTIFICADOR CABI\_TipoPrueba\_01 | |
| Identificador caso prueba | P00 |
| Actividad | Análisis de requerimientos del sistema |
| Métodos o Herramientas | Navegador web Google Chrome |
| Entregables | Lista de chequeo sobre el cumplimiento del requerimiento, ¿realiza lo que el requerimiento describe? |

El entregable de este tipo de pruebas es un listado con las diferentes entradas, diferentes salidas, representación gráfica y descripción del resultado.

|  |  |
| --- | --- |
| NOMBRE Resultado prueba unitaria IDENTIFICADOR CABI\_TipoPrueba\_01 | |
| Entradas | Se listan las entradas a las que se somete el sistema |
| Salidas correspondientes | Se listan las salidas para cada entrada |
| Descripción del comportamiento | La reacción del sistema, definiendo como comportamiento |

Pruebas de Frontera

Pruebas frontera, son las que toman en cuenta valores límite, para verificar el comportamiento de la aplicación en esos casos.

|  |  |
| --- | --- |
| NOMBRE Prueba de frontera IDENTIFICADOR CABI\_TipoPrueba\_01 | |
| Actividad | Se realizarán las distintas pruebas con los valores límites y mínimos que debe recibir el programa. |
| Métodos o Herramientas | Navegador web Google Chrome |
| Entregables | Formato de realización de pruebas de frontera |

Formato entregable de las pruebas de frontera:

|  |  |
| --- | --- |
| NOMBRE Resultado prueba de frontera IDENTIFICADOR CABI\_TipoPrueba\_01 | |
| Valores | Máximo: Mínimo: |
| Resultados Esperados |  |
| Resultados Obtenidos |  |
| Estado | Funciona: No Funciona: |
| Comentarios |  |

Pruebas de Integración

Las pruebas de integración, como su nombre lo indica, son pruebas hechas a un conjunto de requerimientos. En este caso lo dividiremos por secciones.

|  |  |
| --- | --- |
| NOMBRE Prueba de integración IDENTIFICADOR CABI\_TipoPrueba\_01 | |
| Actividad | Validación de requerimientos |
| Métodos o Herramientas | Navegador web Google Chrome |
| Entregables | Informe generado en donde se indica si se 0ene un correcto funcionamiento o no. Formato de realización de prueba de integración |

Formato entregable de las pruebas de integración:

|  |  |  |
| --- | --- | --- |
| ID | GRUPO DE REQUERIMIENTOS | RESULTADOS DE LA PRUEBA |
|  | Grupo de requerimientos que están relacionados por cada historia de usuario | Resultado de la prueba realizada |

Proceso de Pruebas

Casos de prueba

A continuación, se listan los casos de prueba para cada requerimiento.

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Registrar usuario en sistema | Pruebas | P1 |
| Propósito | Verificar que el administrador puede registrar usuarios en el sistema, así como su información para futuros accesos | | |
| Prerrequisitos | Contar con un usuario y contraseña de administrador | | |
| Ubicación | * Base de datos Firebase * Formulario de registro web | | |
| Entrada | * Usuario * Contraseña | | |
| Oráculo | Registro exitoso de usuario | | |
| Pasos | 1. Ingresar a la página web de CABI 2. Ingresar usuario 3. Ingresar contraseña 4. Dar click en ingresar | | |
| Módulos Asociados | * Usuarios * Formulario de Login * Formulario de registro | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Generar permiso de acceso a visitantes | Pruebas | P2 |
| Propósito | Generar ID temporal para visitantes ajenos a la comunidad universitaria | | |
| Prerrequisitos | Contar con formulario de datos temporal móvil | | |
| Ubicación | * Formulario de datos temporal móvil * Base de datos Firebase | | |
| Entrada | Datos de usuario | | |
| Oráculo | Acceso concedido | | |
| Pasos | 1. Ingresar a la aplicación móvil de CABI 2. Seleccionar Acceso a visitantes 3. Ingresar datos 4. Generar ID de acceso | | |
| Módulos Asociados | * Usuarios * Formulario temporal | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Modificar datos de usuario | Pruebas | P3 |
| Propósito | Modificar los datos de cualquier usuario para mantenerlos actualizados | | |
| Prerrequisitos | * Contar con usuario y contraseña de administrador * Usuario que desea modificar datos debe estar registrado | | |
| Ubicación | * Formulario de datos web * Base de datos Firebase | | |
| Entrada | * Usuario y contraseña de administrador * Usuario y contraseña de cuenta a modificar | | |
| Oráculo | Modificación de datos exitosa | | |
| Pasos | 1. Ingresar a página web de CABI 2. Ingresar usuario administrativo 3. Ingresar contraseña administrativa 4. Ingresar a modificar datos de un usuario 5. Confirma usuario y contraseña de datos 6. Modificar datos 7. Guardar cambios | | |
| Módulos Asociados | * Formulario Login * Formulario de datos * Usuarios | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Reflejar datos de usuario en aplicación móvil | Pruebas | P4 |
| Propósito | Reflejar los datos de un usuario registrado en la aplicación móvil, así como un código que contenga los mismos | | |
| Prerrequisitos | Registro del usuario por parte del administrador | | |
| Ubicación | * Perfil de usuario en aplicación móvil * Base de datos Firebase | | |
| Entrada | * Usuario de solicitante * Contraseña de solicitante | | |
| Oráculo | Se ingresó de manera exitosa | | |
| Pasos | 1. Ingresar usuario 2. Ingresar Contraseña 3. Click en ingresar | | |
| Módulos Asociados | * Usuarios * Formulario Login | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Validación y registro de entradas y salidas | Pruebas | P5 |
| Propósito | Identificar los datos de un usuario y registrar su entrada a ciudad universitaria | | |
| Prerrequisitos | * Usuario registrado en el sistema * Usuario cuenta con código QR que contiene su información | | |
| Ubicación | Validación y registro de acceso en aplicación móvil | | |
| Entrada | Escaneo de código QR | | |
| Oráculo | Validación exitosa, usuario X a ingresado | | |
| Pasos | 1. Ingresar a aplicación móvil 2. Seleccionar apartado Validar acceso 3. Identificarse como Guardia de acceso 4. Escanear código QR de usuario | | |
| Módulos Asociados | * Usuarios * Validación de acceso | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Reportar discrepancias | Pruebas | P6 |
| Propósito | Indicar discrepancias sobre usuarios que intentan ingresar o salir a Ciudad Universitaria | | |
| Prerrequisitos | * Usuario registrado * Usuario cuenta con código de identificación | | |
| Ubicación | Validación y registro de acceso en aplicación móvil | | |
| Entrada | Botón de reporte sobre un usuario | | |
| Oráculo | Se ha reportado a usuario X | | |
| Pasos | 1. Ingresar a aplicación móvil 2. Seleccionar apartado Validar acceso 3. Identificarse como Guardia de acceso 4. Escanear código QR de usuario 5. Presionar botón de levantar reporte | | |
| Módulos Asociados | * Usuarios * Validación de acceso | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Usuarios administrativos | Pruebas | P7 |
| Propósito | Validar el ingreso al sistema mediante el uso de correo y contraseña. | | |
| Prerrequisitos | Usuarios administrativos registrados en la base de datos | | |
| Ubicación | * Base de datos Firebase * Formulario Login | | |
| Entrada | * Usuario * Contraseña | | |
| Oráculo | Acceso concedido | | |
| Pasos | 1. Ingresar a la página web de CABI 2. Ingresar usuario 3. Ingresar contraseña 4. Dar click en ingresar | | |
| Módulos Asociados | * Usuarios * Formulario Login | | |

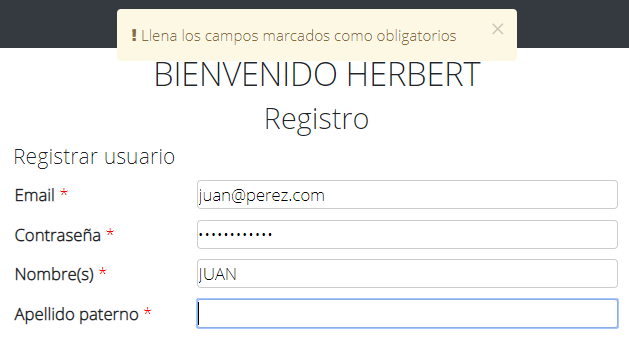
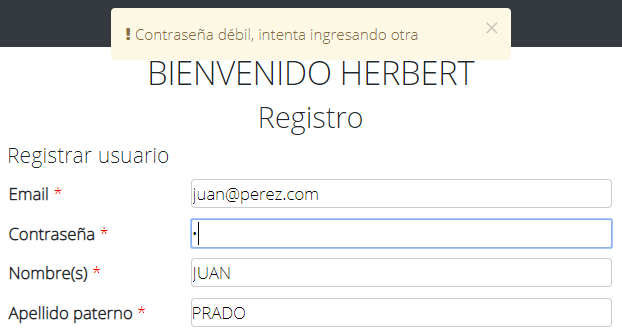
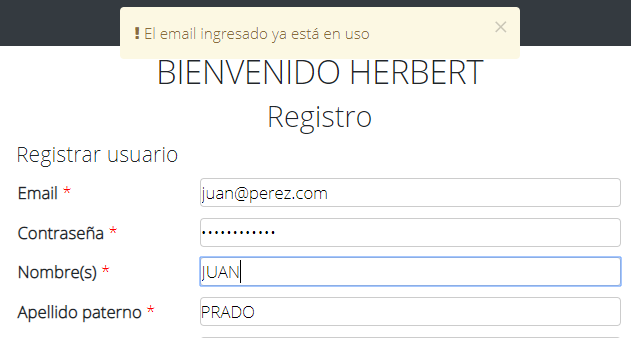
Pruebas unitarias

Pruebas para frontend

|  |  |
| --- | --- |
| NOMBRE Prueba unitaria IDENTIFICADOR CABI\_FE\_01 | |
| Identificador caso prueba | P1 |
| Actividad | Registrar un nuevo usuario en el sistema. |
| Métodos o Herramientas | Validación de datos por scripts. |
| Entregables | Tabla con los resultados de la prueba, correspondiente al identificador. |

|  |  |
| --- | --- |
| NOMBRE Resultado prueba unitaria IDENTIFICADOR CABI\_FE\_01 | |
| Entradas | * Campos incompletos * Correo ya registrado (existente) * Contraseña demasiado débil * Registro todos los campos llenos |
| Salidas correspondientes | * Mensaje de error ‘campos incompletos’ * Mensaje ‘ya existe ese correo’ * Mensaje ‘Contraseña débil’ * Registro correcto |
| Descripción del comportamiento | El sistema maneja correctamente las acciones mencionadas. |

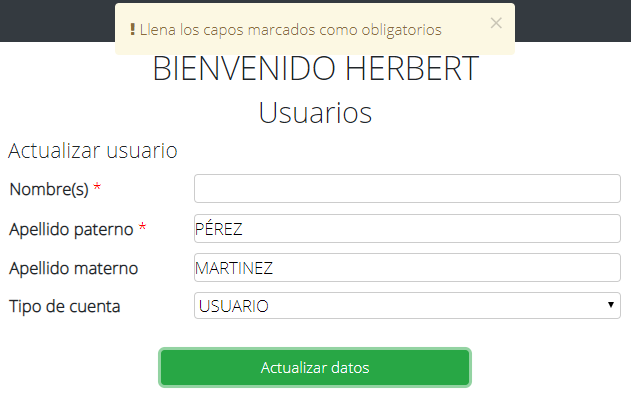
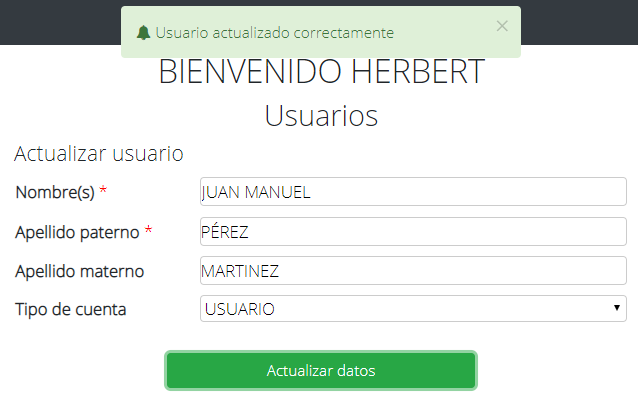
Comportamiento gráfico del sistema:

|  |  |
| --- | --- |
| NOMBRE Prueba unitaria IDENTIFICADOR CABI\_FE\_03 | |
| Identificador caso prueba | P3 |
| Actividad | Modificar datos de un usuario. |
| Métodos o Herramientas | Validación de datos y solicitud de almacenamiento por scripts. |
| Entregables | Tabla con los resultados de la prueba, correspondiente al identificador. |

|  |  |
| --- | --- |
| NOMBRE Resultado prueba unitaria IDENTIFICADOR CABI\_FE\_03 | |
| Entradas | * Campos incompletos * Todos los campos completos |
| Salidas correspondientes | * Mensaje de error ‘campos incompletos’ * Mensaje correcto ‘datos actualizados’ |
| Descripción del comportamiento | El sistema maneja correctamente las acciones mencionadas. |

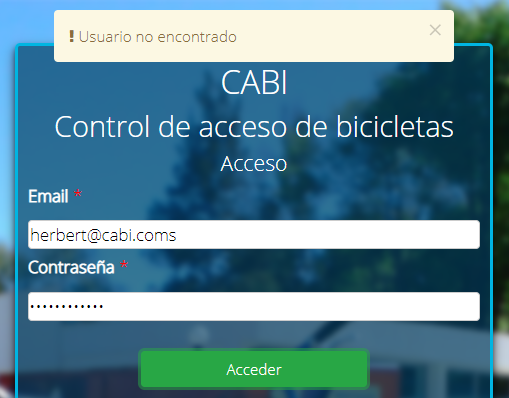
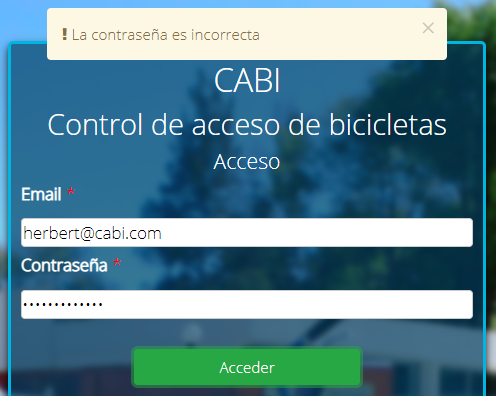
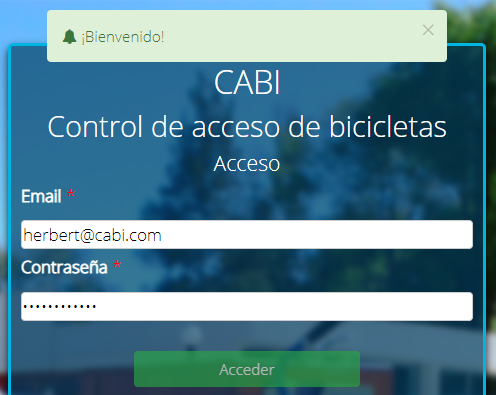
Comportamiento gráfico del sistema:

|  |  |
| --- | --- |
| NOMBRE Prueba unitaria IDENTIFICADOR CABI\_FE\_07 | |
| Identificador caso prueba | P7 |
| Actividad | Validar el acceso de un usuario al sistema. |
| Métodos o Herramientas | Validación de datos por scripts. |
| Entregables | Tabla con los resultados de la prueba, correspondiente al identificador. |

|  |  |
| --- | --- |
| NOMBRE Resultado prueba unitaria IDENTIFICADOR CABI\_FE\_07 | |
| Entradas | * Correo inválido, contraseña correcta * Correo válido, contraseña incorrecta * Correo válido, contraseña correcta |
| Salidas correspondientes | * Mensaje de error ‘correo inválido’ * Mensaje de error ‘contraseña incorrecta * Mensaje de acceso correcto ‘Bienvenido’ |
| Descripción del comportamiento | El sistema maneja correctamente las acciones mencionadas. |

Comportamiento gráfico del sistema:

Pruebas para backend

La simulación de escritura y lectura en base de datos es proporcionada por la herramienta para probar dichas acciones en la consola de Firebase.

|  |  |
| --- | --- |
| NOMBRE Prueba unitaria IDENTIFICADOR CABI\_BE\_01 | |
| Identificador caso prueba | P1 |
| Actividad | Registrar un nuevo usuario en el sistema. |
| Métodos o Herramientas | Validación de datos a través de reglas en formato JSON (Firebase Realtime Database). |
| Entregables | Tabla con los resultados de la prueba, correspondiente al identificador. |

|  |  |
| --- | --- |
| NOMBRE Resultado prueba unitaria IDENTIFICADOR CABI\_FE\_01 | |
| Entradas | 1. Usuario sin autenticarse 2. Un usuario distinto de ADMINISTRADOR intenta el registro 3. Tipo de cuenta inválida 4. Estructura inválida 5. Campos vacíos 6. Las letras no están en mayúsculas 7. Registro completo |
| Salidas correspondientes | 1. Rechazo de escritura (se debe haber autenticado) 2. Rechazo de escritura (no tiene permisos) 3. Rechazo de escritura (no es un usuario válido) 4. Rechazo de escritura (no es la estructura requerida) 5. Rechazo de escritura (hay campos vacíos) 6. Rechazo de escritura (hay letras en minúsculas en los campos que deberían ir mayúsculas) 7. Se permite la escritura, todas las reglas correctas. |
| Descripción del comportamiento | El sistema maneja correctamente las entradas mencionados. |

Comportamiento del sistema de base de datos Firebase Realtime Database:

1)

2)



3)

4)

5)

6)

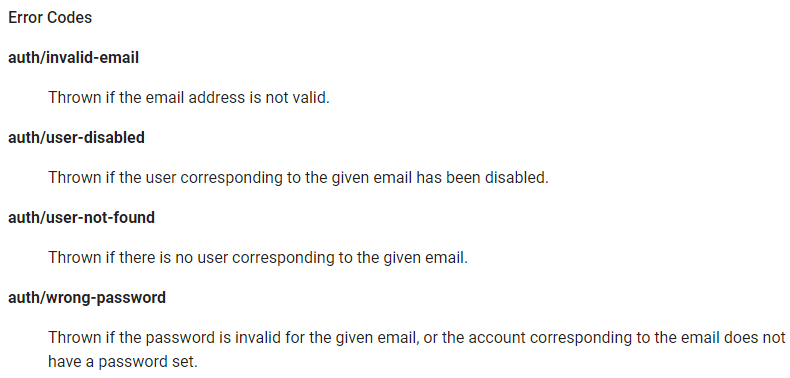
7)

|  |  |
| --- | --- |
| NOMBRE Prueba unitaria IDENTIFICADOR CABI\_BE\_03 | |
| Identificador caso prueba | P3 |
| Actividad | Modificar datos de un usuario. |
| Métodos o Herramientas | Validación de datos a través de reglas en formato JSON (Firebase Realtime Database). |
| Entregables | Tabla con los resultados de la prueba, correspondiente al identificador. |

|  |  |
| --- | --- |
| NOMBRE Resultado prueba unitaria IDENTIFICADOR CABI\_BE\_03 | |
| Entradas | La actualización de datos involucra las mismas acciones de escritura que el registro de un usuario nuevo por lo que las entradas son las mismas que en la prueba unitaria CABI\_BE\_01. |
| Salidas correspondientes | Las mismas que en la prueba unitaria CABI\_BE\_01. |
| Descripción del comportamiento | El sistema maneja correctamente las entradas mencionados . |

|  |  |
| --- | --- |
| NOMBRE Prueba unitaria IDENTIFICADOR CABI\_BE\_07 | |
| Identificador caso prueba | P7 |
| Actividad | Validar el acceso de un usuario al sistema. |
| Métodos o Herramientas | Validación de datos a través de Firebase Authentication. |
| Entregables | Tabla con los resultados de la prueba, correspondiente al identificador. |

|  |  |
| --- | --- |
| NOMBRE Resultado prueba unitaria IDENTIFICADOR CABI\_BE\_07 | |
| Entradas | Correo y contraseña. |
| Salidas correspondientes | La información para las diferentes salidas se muestra en la siguiente ilustración. |
| Descripción del comportamiento | El sistema maneja correctamente las entradas mencionados. |



Pruebas de frontera

Para la aplicación, los únicos valores a los que se les puede aplicar pruebas de frontera son a la longitud en los campos y el tamaño de la rodada de la bicicleta.

|  |  |
| --- | --- |
| NOMBRE Prueba de frontera IDENTIFICADOR CABI\_FR\_01 | |
| Actividad | Probar la escritura para rodada en un valor 14 a 29. |
| Métodos o Herramientas | Validación de datos a través de reglas en formato JSON (Firebase Realtime Database). |
| Entregables | Formato de realización de pruebas de frontera. |

|  |  |
| --- | --- |
| NOMBRE Resultado prueba de frontera IDENTIFICADOR CABI\_FR\_01 | |
| Valores | Máximo: 29 Mínimo: 14 |
| Resultados Esperados | Se admiten valores de 14 a 29. |
| Resultados Obtenidos | Solo se permite la escritura de números en ese rango. |
| Estado | Funciona: Validación backend No Funciona: NA |
| Comentarios | La validación es tanto para frontend y backend, pero al ser la BD el último recurso en comprobar, solo se muestra la validación en backend. |

1) Rodada con valor 13

2) Rodada con valor 20

3) Rodada con valor 30

|  |  |
| --- | --- |
| NOMBRE Prueba de frontera IDENTIFICADOR CABI\_FR\_02 | |
| Actividad | Probar la escritura para cadenas en donde el número de caracteres en el campo es mayor de 100 (La entrada para campos vacíos ya ha sido probada, véase prueba CABI\_BE\_01). |
| Métodos o Herramientas | Validación de datos a través de reglas en formato JSON (Firebase Realtime Database). |
| Entregables | Formato de realización de pruebas de frontera. |

|  |  |
| --- | --- |
| NOMBRE Resultado prueba de frontera IDENTIFICADOR CABI\_FR\_01 | |
| Valores | Máximo: 99 caracteres Mínimo: 1 |
| Resultados Esperados | Se admiten cadenas con 1 a 99 caracteres. |
| Resultados Obtenidos | Solo se permite la escritura cadenas con ese número de caracteres. |
| Estado | Funciona: Validación backend No Funciona: NA |
| Comentarios | La validación es tanto para frontend y backend, pero al ser la BD el último recurso en comprobar, solo se muestra la validación en backend. |

1) Escritura de campo nombre con 100 caracteres

2) Escritura de campo nombre con 54 caracteres