**Plataforma para la comunicación y colaboración en clases asíncronas “MEAN”**

### **Planificación del desarrollo**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| N° | Tarea | Responsable | Iniciar | Terminar | Duración |
| 1 | P1 | Pedro  José  Gustavo | 07 de Mayo | 04 de Junio | 28 días |
| 2 | P2 | Pedro | 07 de Mayo | 04 de Junio | 28 días |
| 3 | P3 | Gustavo | 07 de Mayo | 04 de Junio | 28 días |
| 4 | P4 | José | 07 de Mayo | 04 de Junio | 1. días |
| 5 | P5 | Gustavo | 07 de Mayo | 04 de Junio | 29 días |

### **SPRINT 1: GESTIÓN DE GRUPOS DE ESTUDIO**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N° SPRINT :** | 1 | **MÓDULO:** |  | GESTIÓN DE GRUPOS DE ESTUDIO |
| **TAREAS** | **DOCUMENTO** | **FECHA INICIO** | **FECHA FINALIZA** | **OBSERVACIÓN** |
| **ANÁLISIS** | Diagramas de casos de uso expandidos de gestión de grupos de estudio | 07/05 | 10/05 |  |
| **Diseño de base de datos** | Diagrama XML |  |  |  |
| **Diseño de interfaces** | Mockups:   * Administración de información e integrantes de grupo * Ventana de listado de grupos | 10/05 |  |  |
| **Diseño de proceso** | Diagrama de actividades |  |  |  |
| **Diseño navegacional** | Diagrama de flujo de interfaces |  | 14/05 |  |
| **Desarrollo del código** | Base de datos.  Interfaces (Pantallas).  Scripts. | 14/05 | 29/05 |  |
| **Pruebas** | Pruebas unitarias | 29/05 | 4/06 |  |

### **Análisis de “Gestión de grupos de estudio”**

* **Diagrama de casos de uso expandido**

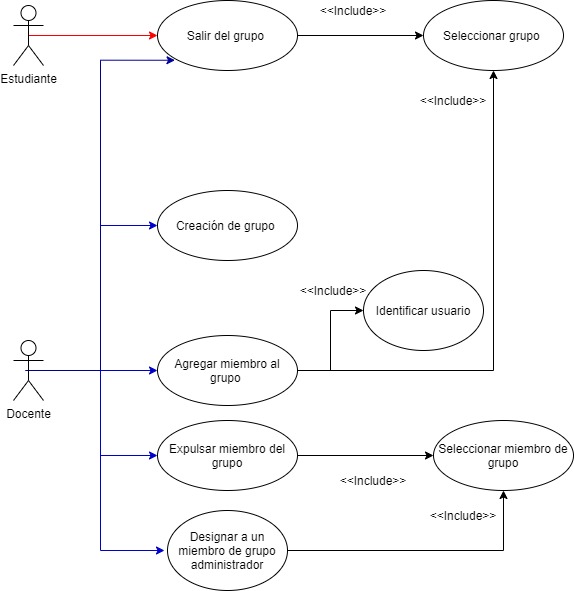


Figura 1, caso de uso expandido para: Gestión de grupos de estudio.

* **Tablas descriptivas**

|  |  |
| --- | --- |
| **Caso de uso** | Gestión de grupos de estudio |
| **Actores** | Estudiante y Docente |
| **Propósito** | Permitir a los diferentes usuarios participar de grupos de estudio clasificándose como: miembro de grupo o administrador de grupo. |
| **Tipo** | Primario |
| **Referencia cruzada** | Inicio de sesión// REQUISITOS QUE SE ESTÁN CUMPLIENDO |
| **CURSO NORMAL DE ACCIONES** | |
| **ACCIÓN DEL ACTOR** | **RESPUESTA DEL SISTEMA** |
| 1. El usuario selecciona un grupo de estudio | 1. El sistema apunta al registro de un grupo almacenado en la base de datos. |
| 1. El usuario selecciona Salir de un grupo de estudio | 1. El sistema verifica su identificador y elimina su registro de la base de datos. |
| 1. El usuario selecciona el botón de “Creación de grupo” 2. El usuario llenara el formulario presentado por el sistema y pulsara un botón para enviar el formulario | 1. El sistema le muestra al usuario un formulario de datos que se registraran con el grupo 2. EL sistema verificara el formulario y creara la instancia del grupo. |
| 1. El usuario seleccionara la opción de eliminar miembro de grupo 2. El usuario seleccionara un miembro de grupo 3. El usuario pulsara un botón para confirmar la eliminación | 1. El sistema le pedirá a un usuario seleccionar al usuario 2. El sistema apuntara al registro de un miembro del grupo 3. El sistema borrara el registro del miembro en el grupo seleccionado almacenado dentro de la base de datos |
| 1. El usuario seleccionara la opción de agregar miembro de grupo 2. El usuario seleccionara un miembro de grupo 3. El usuario pulsara un botón para confirmar la agregación | 1. El sistema le pedirá a un usuario seleccionar al usuario 2. El sistema apuntara al registro de un miembro del grupo 3. El sistema agregara el registro del miembro en el grupo seleccionado almacenado dentro de la base de datos |
| 1. El usuario seleccionara la opción de designar administrador de grupo 2. El usuario seleccionara un miembro de grupo 3. El usuario pulsara un botón para confirmar la designación de administrador | 1. El sistema le pedirá a un usuario seleccionar al usuario 2. El sistema apuntara al registro de un miembro del grupo 3. El sistema cambiara el registro del miembro en el grupo seleccionado almacenado dentro de la base de datos |
|  |  |

* 1. **DISEÑO DE BASE DE DATOS**

A continuación, se muestra el diagrama XML de nuestra base de datos, se hace uso de este diagrama dado que para el presente proyecto se utiliza una base de datos no relacional (NoSQL) como lo es mongoDB, por lo tanto, tampoco se muestra un diagrama de clases.

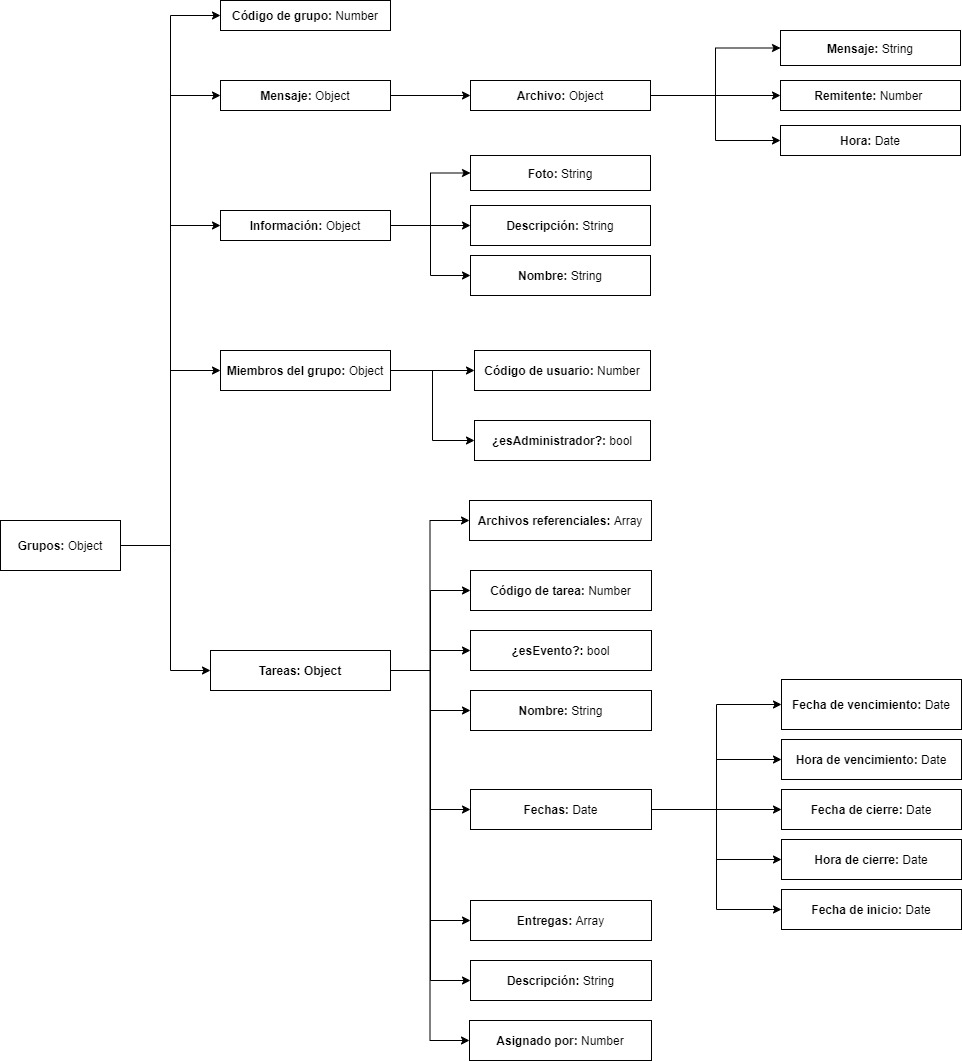


Figura 2, diagrama XML para el esquema de datos.

### **Diseño de interfaces**

* **Interfaz de creación de grupo de estudio, agregar miembro a un grupo de estudio, eliminar miembro.**

**Salir de un grupo de estudio**

En la siguiente pantalla se observa un menú que se desplegará al hacer clic sobre el nombre del grupo. Dentro de este menú se encuentran varías opciones de entre las cuales la que nos interesa es la de “Salir del grupo” *señalada con un rectángulo rojo,* al seleccionar esta opción se despliega una ventana emergente con dos opciones para confirmar la acción.

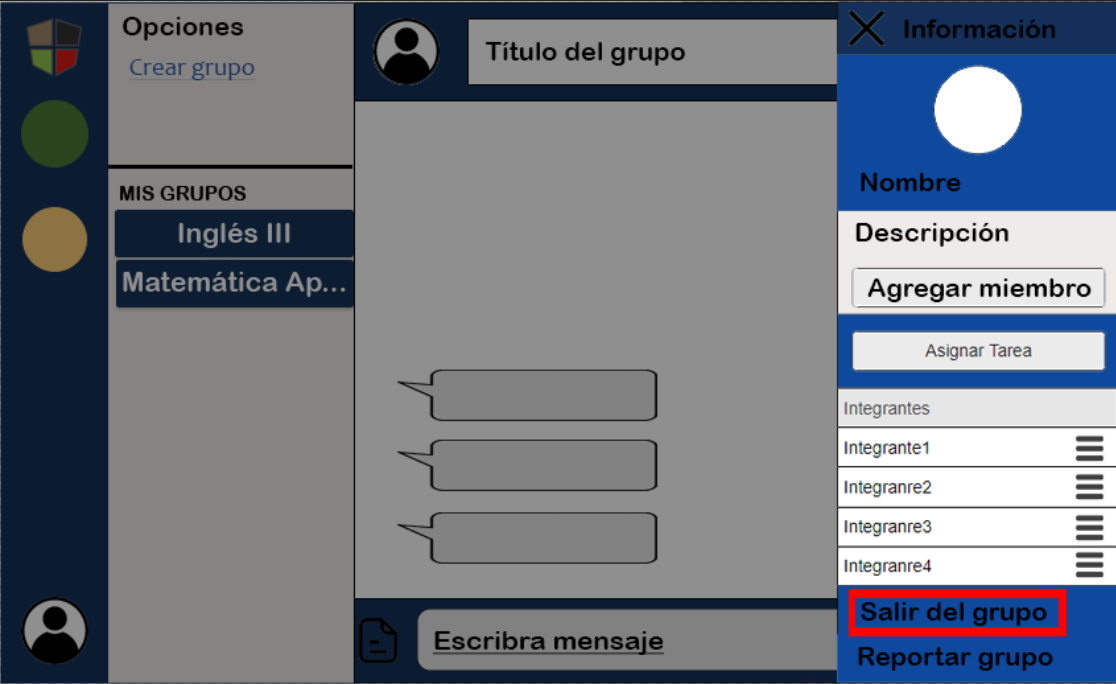


Figura 3, mockup de menú de opciones del grupo versión web.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 4, mockup de menú de opciones del grupo versión móvil.

**Interfaz de creación de grupo de estudio**

En la siguiente imagen se muestra el *Mockup* de la pantalla de creación del grupo *[Figura 5].* La misma se ubicará en el apartado para los chats al seleccionar el botón de “crear grupo” ubicado en la pantalla principal de la aplicación.

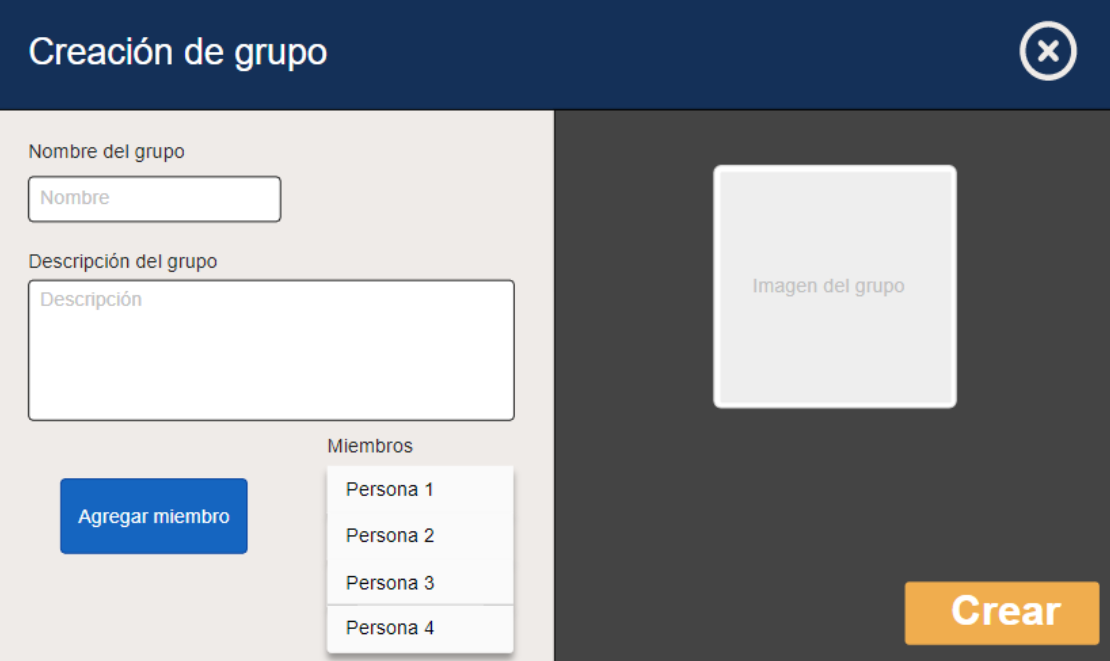


Figura 5, mockup de creación de grupo versión web.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 6, mockup de creación de grupo versión móvil.

**Agregar miembro a un grupo**

El Mockup que se muestra a continuación corresponde a la pantalla de búsqueda de personas, la ventana mostrada saldrá de manera emergente al seleccionar la opción de: *“Agregar miembro” [Figura 5]* o al seleccionar el botón homónimo en el menú de opciones del grupo *[Figura 3]*.

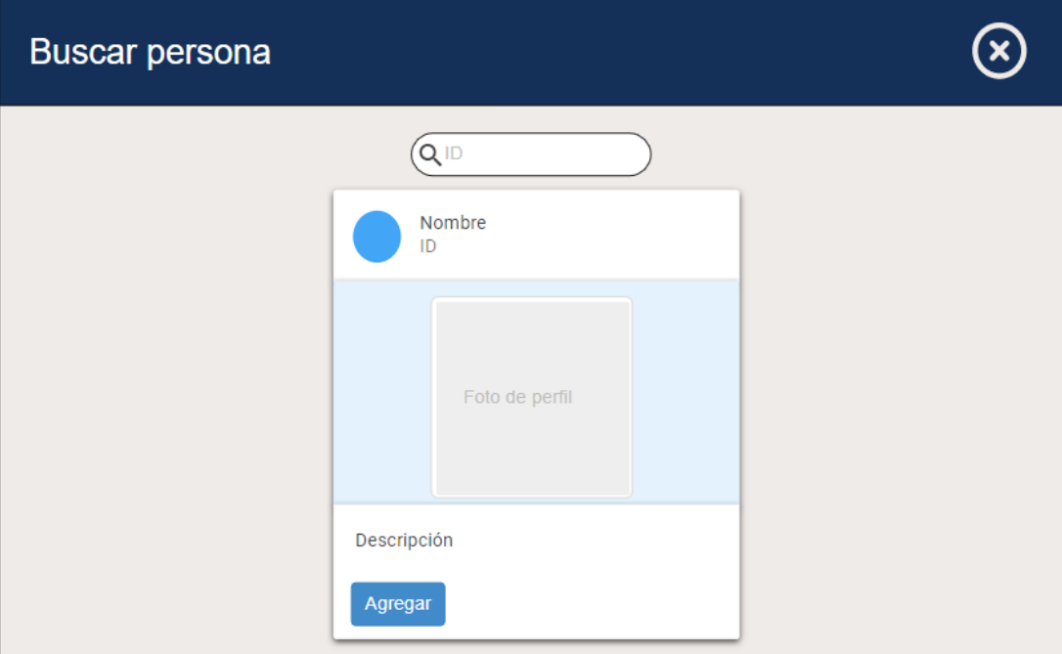


Figura 7, mockup de buscar persona versión web.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 8, mockup de buscar persona versión móvil.

**Eliminar un miembro**

Para poder eliminar un miembro de un grupo de estudio es necesario: primero, acceder al menú de opciones de grupo *[Figura 3]*, dentro de la misma figura se mostrará una lista con todos los integrantes del grupo, cada elemento de la lista tendrá un botón con la imagen de tres barras verticales en el lado derecho del nombre, al seleccionar dicho botón se desplegará un pequeño con menú con dos opciones, al seleccionar la opción de eliminar miembro saldrá una ventana emergente con dos botones para confirmar o cancelar la acción.



Figura 9, mockup de eliminar miembro versión web.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 10, mockup de eliminar miembro versión móvil.

**Designar administrador**

Los pasos que seguir para designar un administrador son iguales a los pasos indicados en la pantalla anterior *[Figura 9],* con la excepción de que aquí en lugar de seleccionar “*Eliminar miembro*” se seleccionará la opción de “*Hacer* administrador”



Figura 11, mockup de designar administrador versión web.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 12, mockup de designar administrador versión móvil.

* 1. **Diagrama de proceso**

A) Diagrama de procesos

**Salir de un grupo**

Diagrama

Descripción generada automáticamente

Figura 13, diagrama de actividades para el proceso: salir del grupo.

**Crear un grupo**

Diagrama

Descripción generada automáticamente

Figura 14, diagrama de actividades para el proceso: crear un grupo.

**Agregar un miembro**

Diagrama

Descripción generada automáticamente

Figura 15, diagrama de actividades para el proceso: agregar un miembro.

**Eliminar un miembro**

Diagrama

Descripción generada automáticamente

Figura 16, diagrama de actividades para el proceso: Eliminar un miembro

### **Diseño navegacional**

* **Diagrama de componentes**

Diagrama

Descripción generada automáticamente

Figura 17, Diagrama de componentes de interfaces del proceso: gestión de grupos.

* **Diagrama de navegación**

Diagrama

Descripción generada automáticamente

Figura 18, Diagrama de navegación del proceso: gestión de grupos.

**PRUEBAS**

|  |  |  |  |
| --- | --- | --- | --- |
| PRUEBA UNITARIA | | | |
| SUB PROCESO/MODULO | GESTION DE GRUPOS | N° | 1 |
| OBJETIVO: | CREAR UN GRUPO DE ESTUDIO | | |
| PRE CONDICIÓN/ RESTRICCIONES | HABERSE REGISTRADO, HABER INICIADO SESION | | |
| PROCESO | 1.Entrar a Mis Grupos, 2. Seleccionar la opcion "Crear Grupo", 3. Rellenar los campos vacios con los datos pedidos, 4. Agregar miembros, 5. Añadir una imagen de grupo, 6. Seleccionar la opcion "Crear". | | |
| RESULTADOS ESPERADOS | Un grupo creado exitosamente. | | |
| EQUIPOS UTILIZADOS | EQUIPOS (HARDWARE, SOFTWARE, RED) | | |
| RESPONSABLE | Pedro E. Chacolla Espinoza, José F. Quezada Quiroga, Gustavo F. Valda Dorado | | |
| FECHA | 10/6/2021 | | |
| USUARIOS FINALES | Los estudiantes y docentes. | | |
| RESULTADOS OBTENIDOS | Se ha creado un grupo exitosamente. | | |
| CAMBIOS PROPUESTOS | Actualización de registros de la base de datos y interfaz de usuario actualizada. | | |
| FECHA DE LA SIGUIENTE PRUEBA | 10-jun | N° 2 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| RESPONSABLE 1 | RESPONSABLE 2 |  |  |
|  |  |  |  |
|  |  |  |  |
| USUARIO FINAL |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| PRUEBA UNITARIA | | | |
| SUB PROCESO/MODULO | GESTION DE GRUPOS | N° | 2 |
| OBJETIVO: | BUSCAR UN USUARIO | | |
| PRE CONDICIÓN/ RESTRICCIONES | HABERSE REGISTRADO, HABER INICIADO SESION, HABER ENTRADO A CREACIÓN GRUPO | | |
| PROCESO | 1.Escribo la ID del usuario que quiero buscar en el campo de busqueda, 2. Presiono el boton de buscar con icono de lupa, 3. Si el usuario existe se desplegara su informacion, caso contrario no se mostrara nada, 4. Selecciono la opción "Añadir miembro". | | |
| RESULTADOS ESPERADOS | El usuario se habra añadido a la creación de grupo exitosamente | | |
| EQUIPOS UTILIZADOS | EQUIPOS (HARDWARE, SOFTWARE, RED) | | |
| RESPONSABLE | Pedro E. Chacolla Espinoza, José F. Quezada Quiroga, Gustavo F. Valda Dorado | | |
| FECHA | 10/6/2021 | | |
| USUARIOS FINALES | Todos los usuarios. | | |
| RESULTADOS OBTENIDOS | Se ha agregado un miembro para la creación exitosamente. | | |
| CAMBIOS PROPUESTOS | Actualización en la interfaz de usuario. | | |
| FECHA DE LA SIGUIENTE PRUEBA | 10-jun | N° 3 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| RESPONSABLE 1 | RESPONSABLE 2 |  |  |
|  |  |  |  |
|  |  |  |  |
| USUARIO FINAL |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| PRUEBA UNITARIA | | | |
| SUB PROCESO/MODULO | GESTION DE GRUPOS | N° | 3 |
| OBJETIVO: | ELIMINAR UN USUARIO DEL GRUPO | | |
| PRE CONDICIÓN/ RESTRICCIONES | HABERSE REGISTRADO, HABER INICIADO SESION, HABER ENTRADO AL MENU DE GRUPO, SER ADMINISTRADOR | | |
| PROCESO | 1. Selecciono un usuario de la lista de integrantes del menu de grupo, 2. Selecciono la opción "Eliminar del Grupo". | | |
| RESULTADOS ESPERADOS | El usuario seleccionado sera eliminado del grupo | | |
| EQUIPOS UTILIZADOS | EQUIPOS (HARDWARE, SOFTWARE, RED) | | |
| RESPONSABLE | Pedro E. Chacolla Espinoza, José F. Quezada Quiroga, Gustavo F. Valda Dorado | | |
| FECHA | 10/6/2021 | | |
| USUARIOS FINALES | Docente | | |
| RESULTADOS OBTENIDOS | Se ha eliminado al usuario exitosamente | | |
| CAMBIOS PROPUESTOS | Actualización en la base de datos y en la interfaz de usuario. | | |
| FECHA DE LA SIGUIENTE PRUEBA | 10-jun | N° 4 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| RESPONSABLE 1 | RESPONSABLE 2 |  |  |
|  |  |  |  |
|  |  |  |  |
| USUARIO FINAL |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| PRUEBA UNITARIA | | | |
| SUB PROCESO/MODULO | GESTION DE GRUPOS | N° | 4 |
| OBJETIVO: | HACER ADMINISTRADOR UN USUARIO DEL GRUPO | | |
| PRE CONDICIÓN/ RESTRICCIONES | HABERSE REGISTRADO, HABER INICIADO SESION, HABER ENTRADO AL MENU DE GRUPO, SER ADMINISTRADOR | | |
| PROCESO | 1. Selecciono un usuario de la lista de integrantes del menu de grupo, 2. Selecciono la opción "Hacer administrador". | | |
| RESULTADOS ESPERADOS | El usuario seleccionado se volvera administrador | | |
| EQUIPOS UTILIZADOS | EQUIPOS (HARDWARE, SOFTWARE, RED) | | |
| RESPONSABLE | Pedro E. Chacolla Espinoza, José F. Quezada Quiroga, Gustavo F. Valda Dorado | | |
| FECHA | 10/6/2021 | | |
| USUARIOS FINALES | Docente | | |
| RESULTADOS OBTENIDOS | Se ha convertido al usuario en administrador exitosamente | | |
| CAMBIOS PROPUESTOS | Actualización en la base de datos y en la interfaz de usuario. | | |
| FECHA DE LA SIGUIENTE PRUEBA | 10-jun | N° 5 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| RESPONSABLE 1 | RESPONSABLE 2 |  |  |
|  |  |  |  |
|  |  |  |  |
| USUARIO FINAL |  |  |  |

**CAPTURAS**

Se muestran las capturas del módulo Creación de grupo. No se específica como se acceden a ellas dado que eso se aclara en la parte de los Mockups.

* **Crear Grupo**

**Versión Web**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Versión Móvil**

**Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente**

* **Buscar Usuario**

**Versión Web**

**Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente**

**Versión Móvil**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

* **Eliminar Usuario de Grupo**

**Tabla

Descripción generada automáticamente**

* **Hacer administrador a un usuario**

**Tabla

Descripción generada automáticamente**

**CODIGO**

Se agrega el código fuente del respectivo proceso, el código mostrado solo es la parte más relevante de todo el código para el módulo (No sé incluye la declaración de variables globales).

**Crear Grupo**

async createGroup(){

    let groupName =this.componentRef.instance.getNombreDeGrupo();

    let groupDescription =this.componentRef.instance.getDescripcionDeGrupo();

    if(this.miembros && Object.keys(this.miembros).length===0 && this.miembros.constructor===Object){

      alert('No se han seleccionado miembros')

      return;

    }

    let integrantesGrupo=[];

    for (let miembro of this.miembros) {

          integrantesGrupo.push(miembro.id)

    }

    integrantesGrupo.push(this.currentUserId);

    let informacionDelGrupo1={

      nombre: groupName.value,

      descripcion:groupDescription.value,

      foto:""

    }

    let mensaje:{

      archivo:{

          mensaje:"",

          remitente:"",

          hora:Date

      }

    }

    this.NewGroupAdmins.push(this.currentUserId);

    let miembrosDelGrupo={

        integrantes:integrantesGrupo,

        admin:this.NewGroupAdmins

    };

    let idGroup:number = -1;

    await this.http.post('http://localhost:3000/getGroupCount',{}).toPromise().then((data:any) => idGroup = data.conteo);

    idGroup++;

    if(idGroup === -1){

      alert('Erro con el id de grupo');

      return;

    }

    let grupo\_n = {id:idGroup, miembrosDelGrupo:miembrosDelGrupo,informacion:informacionDelGrupo1,mensaje:mensaje}

    let data = {

      ids: integrantesGrupo,

      infoGrupo:grupo\_n

    }

    this.socket.socket.emit('nuevoGrupo',data,cb=>{

      if(cb.err){

        console.log(cb.error);

      }

    });

    this.createComponent(1);

  }

**Buscar Usuario**

searchUser(id:any){

      this.userData.id = id;

      this.http.post(`http://localhost:3000/usuarios/${id}`,{}).subscribe(data =>{

      let userData = JSON.stringify(data);

      let userData\_ = JSON.parse(userData);

      this.userData.name = userData\_.nombre;

      this.userData.description = userData\_.descripcion;

      this.userData.fotoPerfil = userData\_.fotoPerfil;

      this.userData.email = userData\_.email;

      this.createComponent(3);

    });

  }

**Eliminar usuario de un grupo**

deleteMember(memberInformation:any){

    let userId = memberInformation.value.id;

    let groupId = this.currentGroupId.toString();

    for (let index = 0; index < this.currentMembers.length; index++) {

      const member = this.currentMembers[index];

      if(member.id === userId){

        this.currentMembers.splice(index,1);

      }

    }

    userId = userId.toString();

    this.socket.emit('salir-grupo',{userId,groupId,expulsado:true});

  }

**Hacer administrador a un usuario**

makeAdmin(memberInformation:any){

    let userId = memberInformation.value.id;

    let groupId = this.currentGroupId;

    this.socket.emit('admin-nuevo',{userId,groupId});

  }

**SPRINT 3: GESTION DE CUENTAS DE USUARIO**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N° SPRINT :** | 3 | **MÓDULO:** |  | GESTION DE CUENTAS DE USUARIO |
| **TAREAS** | **DOCUMENTO** | **FECHA INICIO** | **FECHA FINALIZA** | **OBSERVACIÓN** |
| **ANÁLISIS** | Diagramas de casos de uso expandidos de gestión de cuentas de usuario | 07/05 | 10/05 |  |
| **Diseño de base de datos** | Diagrama XML |  |  |  |
| **Diseño de interfaces** | ADV'S/Mockups:  Interfaz de gestión de cuentas de usuario y perfil de usuario | 10/05 |  |  |
| **Diseño de proceso** | Diagrama de actividades |  |  |  |
| **Diseño navegacional** | Diagrama de flujo de interfaces |  | 14/05 |  |
| **Desarrollo del código** | Interfaces (Pantallas) | 14/05 | 29/05 |  |
| **Pruebas** | Pruebas unitarias | 29/05 | 4/06 |  |

### **Análisis de “Gestión de cuentas de usuario”**

* **Diagrama de casos de uso expandido**

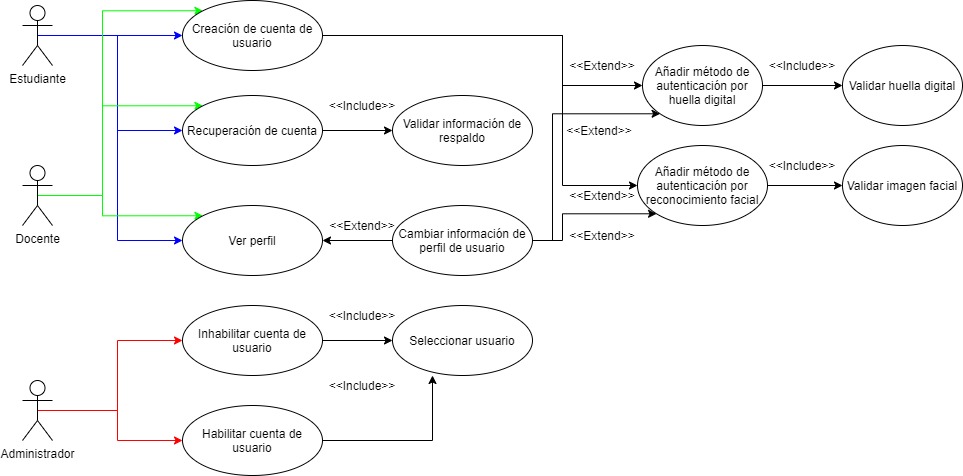


Figura 19, diagramas de casos de uso expandido para el módulo: gestión de cuentas de usuario.

* **Tabla descriptiva**

|  |  |
| --- | --- |
| **Caso de uso** | Gestión de cuentas de usuarios |
| **Actores** | Estudiante , Docente y Administrador del sistema |
| **Propósito** | Permitir a los usuarios docente y estudiante registrar y gestionar una cuenta de usuario en el sistema. Y posibilitar al administrador del sistema gestionar las cuentas de usuario registradas en el sistema. |
| **Tipo** | Primario |
| **Referencia cruzada** | Inicio de sesión |
| **CURSO NORMAL DE ACCIONES** | |
| **ACCIÓN DEL ACTOR** | **RESPUESTA DEL SISTEMA** |
| 1. El usuario selecciona un crear cuenta de usuario 2. EL usuario llena el formulario 3. El usuario pulsa un botón para crear cuenta de usuario | 1. El sistema presenta un formulario para crear una cuenta de usuario. 2. El sistema valida el formulario 3. El sistema añade el registro de un usuario en la base de datos |
| 1. El usuario selecciona la opción de recuperar cuenta | 1. El sistema presenta al usuario un formulario que debe llenar un usuario para crear su cuenta |
| 1. EL usuario selecciona la opción ver perfil | 1. El sistema presenta al usuario una ventana donde el usuario puede visualizar toda su información personal |
| 1. El usuario administrador selecciona la opción de inhabilitar cuenta de usuario 2. El usuario administrador busca al usuario y pulsa un botón para identificarlo 3. El usuario confirma inhabilitar la cuenta de usuario | 1. El sistema presenta al usuario un buscador para identificar el registro de un usuario. 2. El sistema apunta el registro de un usuario 3. El sistema inhabilita la cuenta apuntada cambiando su registro en la base de datos |
| 1. El usuario administrador selecciona la opción de habilitar cuenta de usuario 2. El usuario administrador busca al usuario y pulsa un botón para identificarlo 3. El usuario confirma habilitar la cuenta de usuario | 1. El sistema presenta al usuario un buscador para identificar el registro de un usuario inhabilitado. 2. El sistema apunta el registro de un usuario 3. El sistema habilita la cuenta apuntada cambiando su registro en la base de datos |

* 1. **Diseño de base de datos**

A continuación, se muestra el diagrama XML de nuestra base de datos, se hace uso de este diagrama dado que para el presente proyecto se utiliza una base de datos no relacional (NoSQL) como lo es mongoDB, por lo tanto, tampoco se muestra un diagrama de clases..

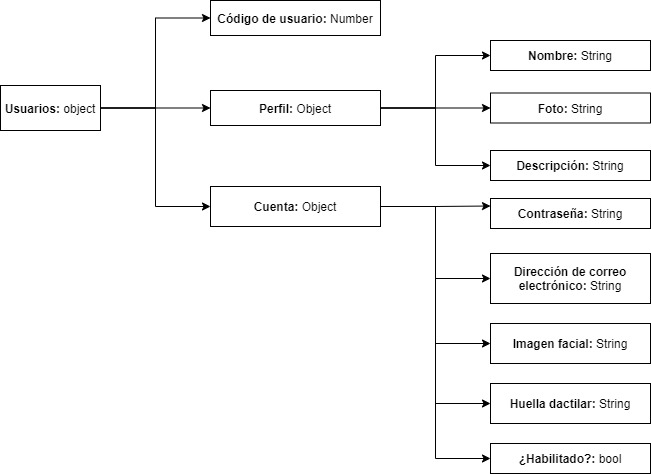


Figura 20, diagrama XML para la colección: Usuarios.

### **Diseño de interfaces**

* **Interfaz de creación de cuenta de usuario y perfil de usuario**

**Creación de cuenta de usuario**

En el siguiente Mockup se muestra la pantalla de registro de cuentas. A la misma se accede luego de seleccionar “Registrarse” *[Figura 35]*. La pantalla se despliega en toda la pantalla de forma no emergente.



Figura 21, mockup de registro de cuentas de usuario versión web.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Figura 22, mockup de registro de cuentas de usuario versión móvil.

**Perfil de usuario**

En el siguiente Mockup se muestra un menú desplegable que emergerá de la izquierda y se posicionará como en la imagen al seleccionar el botón de perfil de usuario. *El mismo puede apreciarse con el ícono de perfil de usuario ubicado en la esquina inferior izquierda [Figura 3].*

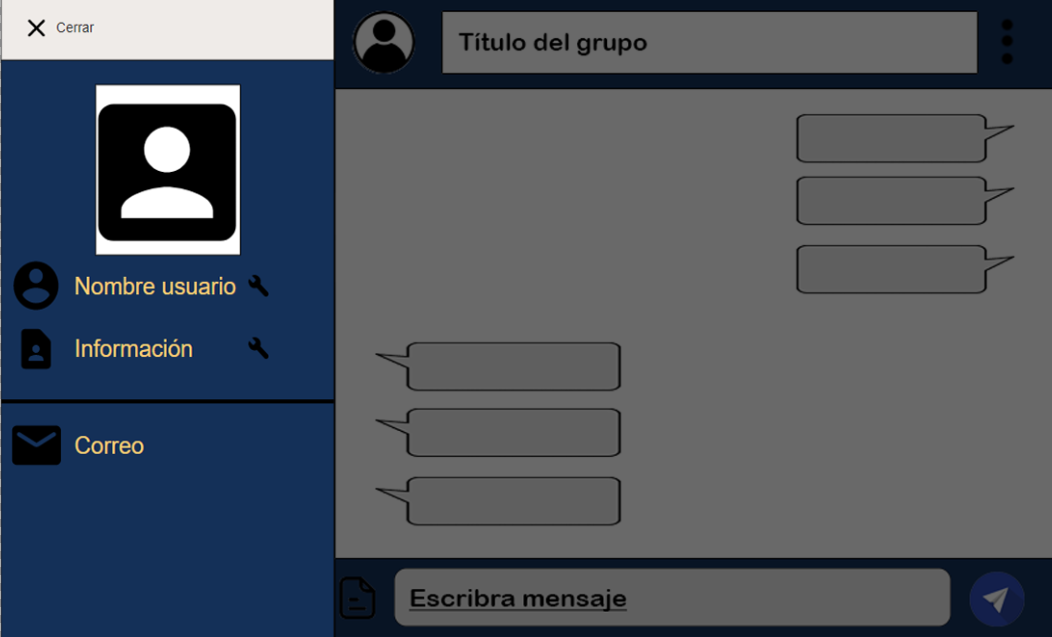


Figura 23, mockup de perfil de usuario versión web.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

Figura 24, mockup de perfil de usuario versión móvil.

**Recuperación de Cuenta**

En la pantalla *[Figura 25]* se muestra la pantalla de recuperación de cuentas a la cual se accede desde el enlace "Olvidaste tu contraseña" *[Figura 35]*. La misma se despliega en toda la pantalla (se usará el mismo mockup para la versión móvil).

Interfaz de usuario gráfica

Descripción generada automáticamente

Figura 25, mockup de perfil de usuario versión móvil.

* 1. **Diagrama de proceso**

A) Diagrama de procesos

**Creación de cuenta**

Diagrama

Descripción generada automáticamente

Figura 26, diagrama de actividades para el proceso: creación de cuenta.

**Recuperación de cuenta**

Diagrama

Descripción generada automáticamente

Figura 27, diagrama de actividades para el proceso: recuperación de cuenta.

**Editar perfil de usuario**

Diagrama

Descripción generada automáticamente

Figura 28, diagrama de actividades para el proceso: editar perfil de usuario.

**Inhabilitar cuenta de usuario**

Diagrama

Descripción generada automáticamente

Figura 29, diagrama de actividades para el proceso: inhabilitar cuentas de usuario.

**Habilitar cuenta de usuario**

Diagrama

Descripción generada automáticamente

Figura 30, diagrama de actividades para el proceso: habilitar cuentas de usuario.

**Ver perfil**

Diagrama

Descripción generada automáticamente

Figura 31, diagrama de actividades para el proceso: ver perfil.

### **Diseño navegacional**

* **Diagrama de componentes**

Diagrama, Esquemático

Descripción generada automáticamente

Figura 32, Diagrama de flujo de interfaces del proceso: gestión de cuentas de usuario.

* **Diagrama de navegación**

Diagrama

Descripción generada automáticamente

Figura 33, Diagrama de flujo de interfaces del proceso: gestión de cuentas de usuario

**PRUEBAS**

Se incluyen las pruebas unitarias de los subprocesos del respectivo modulo.

|  |  |  |  |
| --- | --- | --- | --- |
| PRUEBA UNITARIA | | | |
| SUB PROCESO/MODULO | GESTION DE CUENTAS DE USUARIO | N° | 6 |
| OBJETIVO: | REGISTRARSE EN EL SISTEMA | | |
| PRE CONDICIÓN/ RESTRICCIONES | HABER SELECCIONADO LA OPCIÓN “REGISTRARSE” | | |
| PROCESO | 1. Rellenar los campos vacíos, 2. Seleccionar la opción “Registrase” para completar el registro. | | |
| RESULTADOS ESPERADOS | Se habrá registrado un nuevo usuario en el sistema | | |
| EQUIPOS UTILIZADOS | EQUIPOS (HARDWARE, SOFTWARE, RED) | | |
| RESPONSABLE | Gustavo F. Valda Dorado | | |
| FECHA | 10/6/2021 | | |
| USUARIOS FINALES | Todos los usarios | | |
| RESULTADOS OBTENIDOS | El nuevo usuario se ha registrado en el servidor exitosamente | | |
| CAMBIOS PROPUESTOS | Actualización en la base de datos | | |
| FECHA DE LA SIGUIENTE PRUEBA | 10-jun | N° 7 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| RESPONSABLE 1 | RESPONSABLE 2 |  |  |
|  |  |  |  |
|  |  |  |  |
| USUARIO FINAL |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| PRUEBA UNITARIA | | | |
| SUB PROCESO/MODULO | GESTION DE CUENTAS DE USUARIO | N° | 7 |
| OBJETIVO: | RECUPERAR CONTRASEÑA | | |
| PRE CONDICIÓN/ RESTRICCIONES | HABERSE REGISTRADO | | |
| PROCESO | 1. Seleccionar la opción "¿Olvido su contraseña?" | | |
| RESULTADOS ESPERADOS | Si el usuario ah respuesto correctamente la pregunta de seguridad, se le enviara a su correo con el que creo la cuenta su contraseña, caso contrario no se enviara nada | | |
| EQUIPOS UTILIZADOS | EQUIPOS (HARDWARE, SOFTWARE, RED) | | |
| RESPONSABLE | Gustavo F. Valda Dorado | | |
| FECHA | 10/6/2021 | | |
| USUARIOS FINALES | Todos los usuarios | | |
| RESULTADOS OBTENIDOS | Se ha enviado exitosamente la contraseña al correo | | |
| CAMBIOS PROPUESTOS |  | | |
| FECHA DE LA SIGUIENTE PRUEBA | 10-jun | N° 8 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| RESPONSABLE 1 | RESPONSABLE 2 |  |  |
|  |  |  |  |
|  |  |  |  |
| USUARIO FINAL |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| PRUEBA UNITARIA | | | |
| SUB PROCESO/MODULO | GESTION DE CUENTAS DE USUARIO | N° | 8 |
| OBJETIVO: | REVISAR PERFIL | | |
| PRE CONDICIÓN/ RESTRICCIONES | HABERSE REGISTRADO, HABER INICIADO SESION | | |
| PROCESO | 1. Seleccionar la opción "Perfil" | | |
| RESULTADOS ESPERADOS | El usuario visualizara sus datos de perfil | | |
| EQUIPOS UTILIZADOS | EQUIPOS (HARDWARE, SOFTWARE, RED) | | |
| RESPONSABLE | Gustavo F. Valda Dorado | | |
| FECHA | 10/6/2021 | | |
| USUARIOS FINALES | Todos los usuarios | | |
| RESULTADOS OBTENIDOS | Se ha visualizado la información del usuario exitosamente | | |
| CAMBIOS PROPUESTOS |  | | |
| FECHA DE LA SIGUIENTE PRUEBA | 10-jun | N° 9 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| RESPONSABLE 1 |  |  |  |
|  |  |  |  |
|  |  |  |  |
| USUARIO FINAL |  |  |  |

**CAPTURAS**

Se muestran las capturas del módulo Gestión de cuentas de usuario. No se específica como se acceden a ellas dado que eso se aclara en la parte de los Mockups.

* + **Registro**

**Versión Web**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Versión Móvil**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

* + **Recuperar Contraseña**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

* + **Perfil**

**Versión Web**

**Interfaz de usuario gráfica

Descripción generada automáticamente**

**Versión Móvil**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**CODIGO**

Se agrega el código fuente del respectivo proceso, el código mostrado solo es la parte más relevante de todo el código para el módulo (No sé incluye la declaración de variables globales).

**Registrarse**

async register(){

    let nameField : any = document.getElementById('txtNombre');

    let emailField : any = document.getElementById('email');

    let securityField : any = document.getElementById('cbSecurityQ');

    let securityAns : any = document.getElementById('txtSecurityAns');

    let passwordField : any = document.getElementById('psw');

    let passwordConfirm : any = document.getElementById('psw-repeat');

    this.registerFields = {

      name : nameField.value,

      email : emailField.value,

      securityQuestion : this.questions.filter(x => x.value=== securityField.value)[0].viewValue,

      securityAns : securityAns.value,

      password : passwordField.value

    }

    if(this.registerFields.password !== passwordConfirm.value){

      alert('Las contraseñas no coinciden');

      return;

    }

    for(let field in this.registerFields){

      if(this.registerFields[field]===""){

        alert('Tiene que completar los campos');

        return;

      }

    }

    let newUser;

    await this.http.post('http://localhost:3000/crearUser',this.registerFields).toPromise().then((res:any)=>{

      if(res.error){

        alert(res.mensaje);

        return;

      }else{

        alert('Registrado!');

        newUser = res.user;

      }

    });

    this.socket.emit('login-nuevo',{id: newUser.id,created:true});

    sessionStorage.setItem('currentUser',newUser.id);

    let userDataSave = JSON.stringify(newUser);

    sessionStorage.setItem('currentUserData',userDataSave);

    this.router.navigate(['/','home']);

  }

**Recuperar Contraseña**

async confirmAnswer(answer : String){

    let securityField : any = document.getElementById('securityField');

    let currentQuestion  = this.questions.filter(x => x.value=== securityField.value)[0].viewValue;

    if(currentQuestion === this.recuperationFields.pregunta && answer === this.recuperationFields.respuesta){

      this.http.post(`http://localhost:3000/sendPassword/${this.currentEmail}`,{}).subscribe((res:any)=>{

        let message:HTMLParagraphElement = document.createElement('p');

        message.innerText = `Se ha enviado su contraseña a: ${this.currentEmail}`;

        document.getElementById('question-container').parentElement.append(message);

        //Revisar si funciona

        setTimeout(()=>{

          this.router.navigate(['/','inicio']);

        },3000);

      });

    }else{

      alert('Uno de los campos es incorrecto');

    }

  }

**Perfil de usuario**

async generateUserData() {

    await this.http.post(`http://localhost:3000/usuarios/${this.currentUserId}`,{}).toPromise().then(data =>{

        let aux = JSON.stringify(data);

        this.currentUser = JSON.parse(aux);

    });

    this.currentUser.fotoPerfil = `http://localhost:3000/${this.currentUser.fotoPerfil}`;

    this.http.post('http://localhost:3000/gruposId/',this.currentUser.grupos).subscribe(data =>{

          let userData = JSON.stringify(data);

          this.grupos = JSON.parse(userData);

    });

  }

### **SPRINT 5: INICIO DE SESIÓN EN EL SISTEMA**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N° SPRINT :** | 5 | **MÓDULO:** |  | INICIO DE SESIÓN EN EL SISTEMA |
| **TAREAS** | **DOCUMENTO** | **FECHA INICIO** | **FECHA FINALIZA** | **OBSERVACIÓN** |
| **ANÁLISIS** | Diagramas de casos de uso expandidos del inicio de sesión en el sistema | 07/05 | 10/05 |  |
| 1. **Diseño de interfaces** | ADV'S/Mockups:   * Interfaz de inicio de sesión en el sistema | 10/05 |  |  |
| 1. **Diseño de proceso** | Diagrama de actividades |  |  |  |
| 1. **Diseño navegacional** | Diagrama de flujo de interfaces |  | 14/05 |  |
| **Desarrollo del código** | Interfaces (Pantallas) | 14/05 | 29/05 |  |
| **Pruebas** | Pruebas unitarias | 29/05 | 4/06 |  |

**Análisis de “Inicio de sesión en el sistema”**

* **Diagrama de casos de uso expandido**

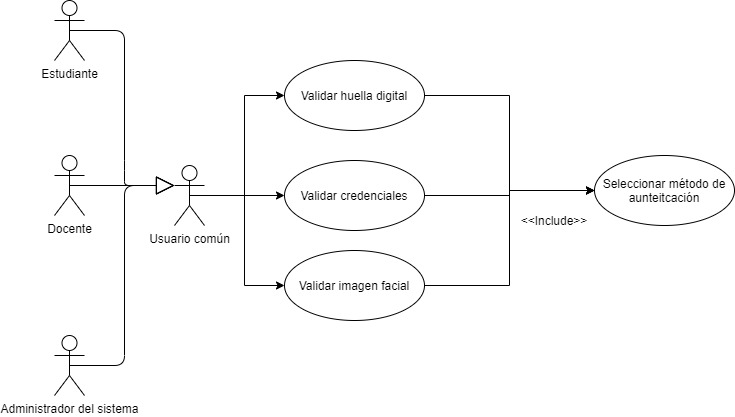


Figura 34, diagrama de casos de uso expandido para el proceso: Inicio de sesión.

* **Tabla descriptiva**

|  |  |
| --- | --- |
| **Caso de uso** | Inicio de sesión en el sistema |
| **Actores** | Estudiante , Docente y Administrador del sistema |
| **Propósito** | Permitir a los diferentes usuarios autenticarse e iniciar sesión en el sistema |
| **Tipo** | Primario |
| **Referencia cruzada** |  |
| **CURSO NORMAL DE ACCIONES** | |
| **ACCIÓN DEL ACTOR** | **RESPUESTA DEL SISTEMA** |
| 1. El usuario ingresa al sistema | 1. El sistema le brinda una interfaz de inicio donde debe de iniciar sesión, registrarse o recuperar su cuenta. |
| 1. El usuario selecciona la opción iniciar sesión. 2. El usuario seleccionar la opción autenticarse por medio de credenciales. 3. El usuario llena el formulario y selecciona un botón para autenticarse. | 1. El usuario le pregunta por tres métodos de autenticación. 2. El sistema le brinda al usuario un formulario donde el usuario debe de validar sus credenciales. 3. EL sistema valida sus credenciales e inicia la sesión del usuario en el sistema. |
| 1. El usuario selecciona la opción iniciar sesión. 2. El usuario seleccionar la opción autenticarse por medio de huella digital. 3. El usuario brinda una imagen de su huella digital y selecciona un botón para autenticarse. | 1. El usuario le pregunta por tres métodos de autenticación. 2. El sistema pide al usuario su huella digital. 3. EL sistema valida su huella digital e inicia la sesión del usuario en el sistema |
| 1. El usuario selecciona la opción iniciar sesión. 2. El usuario seleccionar la opción autenticarse por medio de reconocimiento facial. 3. El usuario brinda una imagen facial y selecciona un botón para autenticarse. | 1. El usuario le pregunta por tres métodos de autenticación. 2. El sistema pide al usuario una imagen facial. 3. EL sistema valida su imagen facial e inicia la sesión del usuario en el sistema. |

### **Diseño de interfaces**

* **Interfaz de trabajos y eventos**

**Interfaz de Logeo**

Se muestra el mockup de la interfaz de login, el mismo será la primera pestaña que se mostrará al acceder a la aplicación.



Figura 35, mockup de login en la aplicación versión web.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 36, mockup de login en la aplicación versión móvil.

### **Diagrama de proceso**

A) Diagrama de procesos

**Login en la aplicación**

Diagrama

Descripción generada automáticamente

Figura 37, diagrama de actividades para el proceso: Login en la aplicación.

* 1. **Diseño navegacional**
* **Diagrama de componentes**

Diagrama

Descripción generada automáticamente

Figura 38, Diagrama de componentes de interfaces del proceso: login en la aplicación versión web.

* **Diagrama de navegación**

Diagrama

Descripción generada automáticamente

Figura 39, Diagrama de navegación de interfaces del proceso: login en la aplicación.

**PRUEBAS**

Se incluyen las pruebas unitarias de los subprocesos del respectivo modulo.

|  |  |  |  |
| --- | --- | --- | --- |
| PRUEBA UNITARIA | | | |
| SUB PROCESO/MODULO | GESTION DE CUENTAS DE USUARIO | N° | 5 |
| OBJETIVO: | INICIAR SESIÓN EN EL SISTEMA | | |
| PRE CONDICIÓN/ RESTRICCIONES | HABERSE REGISTRADO | | |
| PROCESO | 1. Iniciar la aplicación, 2. Ingresar las credenciales requeridas por el sistema (correo y contraseña), 3. Pulsar el boton de iniciar sesión. | | |
| RESULTADOS ESPERADOS | Sesión de usuario iniciada de manera exitosa | | |
| EQUIPOS UTILIZADOS | EQUIPOS (HARDWARE, SOFTWARE, RED) | | |
| RESPONSABLE | Gustavo F. Valda Dorado | | |
| FECHA | 10/6/2021 | | |
| USUARIOS FINALES | Los estudiantes, docentes y administrador. | | |
| RESULTADOS OBTENIDOS | Sesión de usuario iniciada de manera exitosa | | |
| CAMBIOS PROPUESTOS | Interfaz de usuario actualizada. | | |
| FECHA DE LA SIGUIENTE PRUEBA | 10-jun | N° 6 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| RESPONSABLE 1 | RESPONSABLE 2 |  |  |
|  |  |  |  |
|  |  |  |  |
| USUARIO FINAL |  |  |  |

**CAPTURAS**

Se muestran las capturas del módulo Login en la aplicación. No se específica como se acceden a ellas dado que eso se aclara en la parte de los Mockups.

* + **Inicio de Sesión**

**Versión Web**

**Interfaz de usuario gráfica

Descripción generada automáticamente**

**Versión Móvil**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**CODIGO**

Se agrega el código fuente del respectivo proceso, el código mostrado solo es la parte más relevante de todo el código para el módulo (No sé incluye la declaración de variables globales).

**Iniciar sesión**

async login(id:string, password:string):Promise<any>{

    let callbackData;

    if(["1","2","3"].includes(id)){

      this.socket.emit('login-nuevo',{id,test:true});

      sessionStorage.setItem('currentUser',id);

      this.router.navigate(['/','home']);

      return;

    }

    await this.getLoginData(id,password).then(data=>{

      callbackData = data;

    });

    if(!callbackData.error){

      sessionStorage.setItem('currentUser',callbackData.user.id);

      let userDataSave = JSON.stringify(callbackData.user);

      sessionStorage.setItem('currentUserData',userDataSave);

      console.log(callbackData.user);

      this.router.navigate(['/','home']);

    }else{

      alert(callbackData.mensaje);

    }

  }

  async getLoginData(id:string, password:string){

    return new Promise(resolve =>{

      this.socket.socket.emit('login-nuevo',{id,password,test:false},callbackData=>{

        resolve(callbackData);

    });

  });

  }