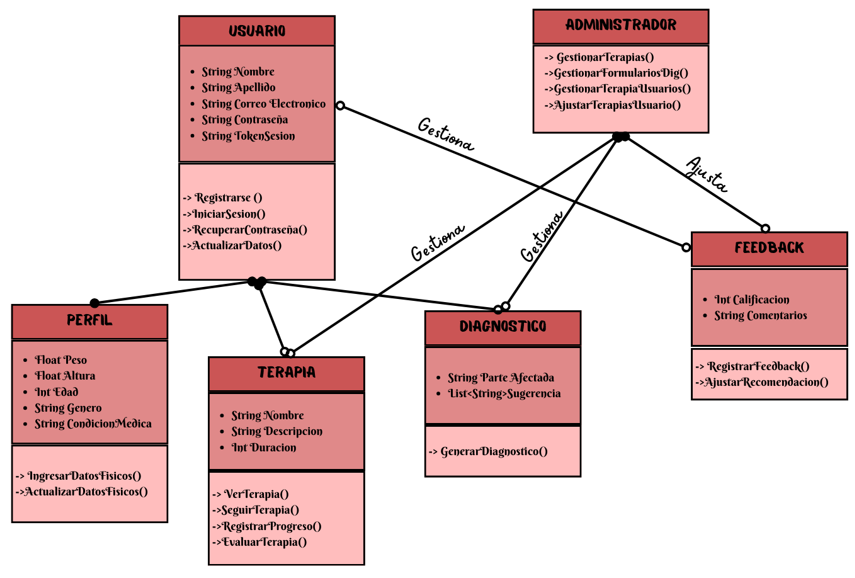
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| GUÍA N°1 TALLER DE PROGRAMACIÓN  Laura Avilés  Ximena Contreras  Nicolás Núñez.  Taller de Programación  Olga Lucia Roa   |  |  |  | | --- | --- | --- | | Identificador: Nombre:  R1 Registro e inicio de sesión en la aplicación. | | | | **Requerimiento que lo Utiliza o Especializa:**  R1.1 | Documentos de visualización Asociados:   Pantalla de registro de usuario.   *Pantalla de inicio de sesión.*  * Pantalla de recuperación de contraseña.* | | | **Entrada:**  ** Correo Electrónico: String (50), formato de email válido.**  ** Contraseña: String (20), debe tener al menos 8 caracteres, incluyendo una letra mayúscula, un número y un carácter especial.**  ** Nombre: String (30), solo letras.**  ** Apellido: String (30), solo letras.** | | **Salida:**  ** Verificación de la creación de la cuenta en la base de datos (usuario registrado exitosamen*te).***  *** Token de sesión generado para la autenticación si el inicio de sesión es exitoso.*** | | **Descripción:**  ** El sistema debe validar que el correo electrónico no esté ya registrado en la base de datos.**  ** El sistema debe validar la fuerza de la contraseña según los criterios mencionados.**  ** Al registrarse, los datos del usuario deben guardarse en la base de datos de manera segura.**  ** En el inicio de sesión, el sistema debe comparar las credenciales proporcionadas con las guardadas en la base de datos para autenticar al usuario.** | | | | **Manejo de Situaciones Anormales**  ** Correo Electrónico Existente: Si el correo ya está registrado, se mostrará un mensaje de error indicando que debe utilizar un correo diferente.**  ** Contraseña Débil: Si la contraseña no cumple con los criterios de seguridad, se mostrará un mensaje de error especificando los requisitos.**  ** Error de Conexión a la Base de Datos: Si no se puede conectar a la base de datos, el sistema deberá mostrar un mensaje de error y sugerir al usuario intentar nuevamente más tarde.** | | | | Criterios de Aceptación   **Registro Exitoso**: El sistema debe gu*ardar l*os datos del usuario en la base de datos de manera correcta y segura.   **Inicio de Sesión Exitoso**: *El sistema debe autenticar al usuario y generar un token de sesión válido.*  ** ***Recuperación de Contraseña:*** *El sistema debe enviar un enlace de recuperación al correo electrónico proporcionado si existe en la base de datos.* | | | |
| |  |  |  | | --- | --- | --- | | Identificador: Nombre:  R2 Mostrar un video de inducción al usuario nuevo. | | | | **Requerimiento que lo Utiliza o Especializa:**  R2.1 | Documentos de visualización Asociados:  Pantalla de video de Inducción | | | **Entrada:**  **Activación automática: El video se muestra automáticamente al usuario nuevo.** | | **Salida:**  ** Video mostrado: El video debe reproducirse correctamente.**  *** Opción de omitir: El usuario puede elegir omitir el video.*** | | **Descripción:**   El video de inducción se muestra al iniciar sesión por primera vez.  * Se debe permitir omitir el video con un botón.* | | | | **Manejo de Situaciones Anormales**  **Error en la reproducción del video: Si el video no carga, mostrar un mensaje de error y la opción de reintentar o saltar el video.** | | | | Criterios de Aceptación  ***Video cargado correctamente:*** *El video debe reproducirse sin problemas.*  ** ***Opción de omitir funcional:*** *La opción de omitir debe ser accesible y funcional.* | | | | |
| |  |  |  | | --- | --- | --- | | Identificador: Nombre:  R3 Ingreso datos físicos usuario. | | | | **Requerimiento que lo Utiliza o Especializa:**  R3.1 | Documentos de visualización Asociados:  Formulario de datos físicos. | | | **Entrada:**  ** Peso: Float, en kilogramos.**  ** Altura: Float, en centímetros.**  ** Edad: In*teger, en años.***  *** Género: String (10).***  *** Condiciones Médicas: String (100).*** | | **Salida:**  **Datos guardados: Los datos ingresados deben almacenarse en la base de datos del usuario.** | | **Descripción:**  ** El usuario introduce sus datos físicos como peso, altura, edad, género, y cualquier condición médica relevante.**  ** Se deben validar los datos ingresados para asegurar que son coherentes.** | | | | **Manejo de Situaciones Anormales**  **Datos incoherentes: Si los datos ingresados no son válidos (por ejemplo, un peso fuera del rango humano posible), se muestra un mensaje de error específico.** | | | | Criterios de Aceptación   **Datos almacenados correctamente:** Los datos deben guardarse en la base de datos sin errores.  ** ***Validación efectiva:*** *Los datos deben ser validados correctamente para evitar incoherencias.* | | | |
| |  |  |  | | --- | --- | --- | | Identificador: Nombre:  R4 Generar diagnóstico inicial. | | | | **Requerimiento que lo Utiliza o Especializa:**  R4.1 | Documentos de visualización Asociados:  Pantalla de diagnóstico inicial. | | | **Entrada:**  ** Parte del cuerpo afectada: String (30), solo se permite seleccionar un grupo mu*scular específico.***  *** Datos físicos del usuario: Datos recopilados previamente (peso, altura, edad, etc.).*** | | **Salida:**  **Diagnóstico generado: Sugerencias de ejercicios y terapias basadas en los datos físicos y parte del cuerpo seleccionada.** | | **Descripción:**   El *sistema debe permitir al usuario seleccionar una parte del cuerpo que le duele (dentro de un grupo muscular específico).*  * Se genera un diagnóstico inicial que sugiere posibles causas y terapias.* | | | | **Manejo de Situaciones Anormales**  **Selección incorrecta: Si el usuario intenta seleccionar más de una parte del cuerpo de diferentes grupos musculares, el sistema debe mostrar un mensaje de error.** | | | | Criterios de Aceptación   *Diagnóstico acertado: El diagnóstico debe ser relevante y basado en los datos físicos del usuario.*  * Validación de selección: El sistema debe validar correctamente que solo se seleccione un grupo muscular.* | | | |
| |  |  |  | | --- | --- | --- | | Identificador: Nombre:  R5 Visualización y seguimiento de progreso en terapias. | | | | **Requerimiento que lo Utiliza o Especializa:**  R5.1 | Documentos de visualización Asociados:  Pantalla de seguimiento de terapias. | | | **Entrada:**  **Progreso diario del usuario: Tiempo dedicado, calificación de bienestar después de la sesión.** | | **Salida:**  ** Progreso registrado: El progreso se almacena en la base de datos.**  *** Tiempo estimado de recuperación: Actualización basada en el progreso registrado.*** | | **Descripción:**   El usuario puede ver las terapias asignadas y registrar su progreso.  * El sistema actualiza el tiempo estimado de recuperación basado en el progreso.* | | | | **Manejo de Situaciones Anormales**  **Error en el registro del progreso: Si el progreso no se puede registrar, se muestra un mensaje de error y se sugiere reintentar.** | | | | Criterios de Aceptación   **Progreso registrado correctamente**: Los datos de progreso deben guardarse de manera confiable.  ** ***Estimación de tiempo precisa:*** *El tiempo estimado de recuperación debe reflejar el progreso real del usuario.* | | | |
| |  |  |  | | --- | --- | --- | | Identificador: Nombre:  R6 Evaluación síntomas después de cada sesión de terapia. | | | | **Requerimiento que lo Utiliza o Especializa:**  R6.1 | Documentos de visualización Asociados:  Pantalla de evaluación de terapia. | | | **Entrada:**  ** Calificación de bienestar: Integer (1-1*0), donde 1 es muy mal y 10 es excelente.***  *** Comentarios adicionales: String (200), opcional.*** | | **Salida:**  **Feedback guardado: La evaluación del usuario se guarda en la base de datos.** | | **Descripción:**  * El usuario califica cómo se siente después de cada sesión de terapia.*  * La calificación se utiliza para ajustar las recomendaciones futuras.* | | | | **Manejo de Situaciones Anormales**  **Calificación fuera de rango: Si la calificación no está dentro del rango permitido, se muestra un mensaje de error.** | | | | Criterios de Aceptación   **Feedback guardado correctamente**: La calificación y comentarios deben guardarse sin problemas.  ** ***Ajustes de recomendaciones:*** *Las recomendaciones deben ajustarse según el feedback del usuario.* | | | |
| |  |  |  | | --- | --- | --- | | Identificador: Nombre:  R7 Gestión perfil personal del usuario. | | | | **Requerimiento que lo Utiliza o Especializa:**  R7.1 | Documentos de visualización Asociados:  Pantalla de perfil de usuario. | | | **Entrada:**   * **Datos personales y físicos: Actualizaciones de nombre, correo, peso, altura, etc.** | | **Salida:**  **Datos actualizados: La información de perfil se guarda en la base de datos.** | | **Descripción:**  El usuario puede ver y editar su información personal y física. | | | | **Manejo de Situaciones Anormales**  **Error al actualizar datos: Si los datos no se pueden actualizar, se muestra un mensaje de error.** | | | | Criterios de Aceptación  **Datos actualizados correctamente**: Las actualizaciones deben reflejarse en la base de datos. | | | |
| Requerimientos Administrador   |  |  |  | | --- | --- | --- | | Identificador: Nombre:  RA1 Permitir al administrador gestionar las terapias disponibles en la base de datos. | | | | **Requerimiento que lo Utiliza o Especializa:**  RA1.1 | Documentos de visualización Asociados:  Pantalla de gestión de terapias. | | | **Entrada:**  **Datos de la terapia: Nombre, descripción, duración estimada, etc.** | | **Salida:**  **Terapia agregada/editada/eliminada: Los cambios se reflejan en la base de datos.** | | **Descripción:**  El administrador puede agregar, editar o eliminar terapias disponibles en la app. | | | | **Manejo de Situaciones Anormales**  **Error en la gestión de la terapia: Si ocurre un error al gestionar una terapia, se muestra un mensaje de error.** | | | | Criterios de Aceptación  **Terapia gestionada correctamente**: Los cambios deben reflejarse inmediatamente en la base de datos. | | | |
| |  |  |  | | --- | --- | --- | | Identificador: Nombre:  RA2 Permitir al administrador gestionar los formularios de diagnóstico. | | | | **Requerimiento que lo Utiliza o Especializa:**  RA2.1 | Documentos de visualización Asociados:  Pantalla de gestión de formularios. | | | **Entrada:**  **Formulario de diagnóstico: Datos y preguntas del formulario.** | | **Salida:**  **Formulario agregado/editado/eliminado: Los cambios se reflejan en la base de datos.** | | **Descripción:**  **El administrador puede ver, editar y eliminar formularios de diagnóstico.** | | | | **Manejo de Situaciones Anormales**  **Error en la gestión del formulario: Si ocurre un error al gestionar un formulario, se muestra un mensaje de error.** | | | | Criterios de Aceptación  **Formulario gestionado correctamente**: Los cambios deben reflejarse en la base de datos sin errores. | | | |
| |  |  |  | | --- | --- | --- | | Identificador: Nombre:  RA3 Permitir al administrador ver y gestionar las terapias de los usuarios. | | | | **Requerimiento que lo Utiliza o Especializa:**  RA3.1 | Documentos de visualización Asociados:  Pantalla de seguimiento de usuarios. | | | **Entrada:**  ** Seleccionar usuario: Identificador del usuario.**  *** Seleccionar terapia: Terapia en curso del usuario.*** | | **Salida:**  **Detalles de la terapia: Información sobre el progreso del usuario.** | | **Descripción:**  El administrador puede ver el progreso de las terapias de los usuarios y realizar ajustes si es necesario. | | | | **Manejo de Situaciones Anormales**  **Error al acceder a los datos: Si los datos no están disponibles, se muestra un mensaje de error.** | | | | Criterios de Aceptación  **Datos accesibles y precisos**: El administrador debe poder acceder a la información de los usuarios sin problemas. | | | |
| |  |  |  | | --- | --- | --- | | Identificador: Nombre:  RA4 Permitir al administrador ajustar las terapias de los usuarios basado en el feedback recibido. | | | | **Requerimiento que lo Utiliza o Especializa:**  RA4.1 | Documentos de visualización Asociados:  Pantalla de ajustes de terapias. | | | **Entrada:**  **Feedback del usuario: Calificación y comentarios del usuario.** | | **Salida:**  **Terapia ajustada: Los ajustes se aplican a la terapia del usuario.** | | **Descripción:**  **El administrador puede ajustar las terapias asignadas a un usuario basándose en su feedback.** | | | | **Manejo de Situaciones Anormales**  **Error al ajustar terapia: Si no se puede realizar el ajuste, se muestra un mensaje de error.** | | | | Criterios de Aceptación  **Ajuste de terapia efectivo**: Los cambios deben reflejarse inmediatamente en la terapia del usuario. | | | |

2. DIAGRAMA DE CLASES



3. PREGUNTAS ORIENTADORAS

¿Cuáles fueron los aprendizajes obtenidos al realizar esta guía?, liste como mínimo 3 aprendizajes y relaciónelos con su futuro que hacer profesional.

* **Importancia de la claridad en el proyecto desde el inicio:** Al definir claramente el objetivo del proyecto desde el principio, se facilita la creación de requerimientos y dependencias, lo que asegura un desarrollo más eficiente. Este aprendizaje es crucial en nuestro futuro profesional, ya que la planificación adecuada es esencial para el éxito en cualquier proyecto de desarrollo de software.
* **Identificación y organización de dependencias:** Aprender a identificar y organizar las dependencias entre los requerimientos me ha permitido entender mejor cómo se interconectan las diferentes partes de un proyecto. En nuestra carrera, esto será vital para asegurar que los desarrollos se realicen en el orden correcto y evitar retrasos o errores.
* **Desglose detallado de requerimientos:** Descomponer los requerimientos en partes más manejables nos enseñó la importancia de abordar problemas complejos de manera estructurada. Esta habilidad será útil en nuestro futuro profesional al diseñar y desarrollar soluciones que satisfagan los criterios de aceptación de manera efectiva.

¿Dónde presentó mayor dificultad resolviendo la guía? ¿cómo lo resolvieron? ¿cuáles fueron las estrategias de solución?

* La mayor dificultad se presentó al encontrar y organizar las dependencias entre los requerimientos. Para resolverlo, identificamos que tener una visión clara del proyecto desde el principio era fundamental. La estrategia de solución fue asegurarnos de comprender completamente el objetivo del proyecto antes de continuar con la creación de los requerimientos y sus dependencias. Esto nos permitió organizar mejor la información y asegurar un desarrollo más ordenado y eficiente.

4. ACTIVIDAD DE TRABAJO AUTÓNOMO

|  |  |  |  |
| --- | --- | --- | --- |
| PARADIGMA DE PROGRAMACIÓN | VENTAJAS | DEVENTAJAS | LENGUAJE DE PROGRAMACIÓN |
| Programación Imperativa | Control detallado del flujo del programa, alta eficiencia. | Puede ser más propenso a errores y difícil de mantener a medida que el programa crece. | C, Pascal |
| Programación Declarativa | Abstracción del "cómo" se realiza la tarea, más fácil de razonar y probar. | Menos control sobre la ejecución detallada, lo que puede conducir a problemas de rendimiento en algunos casos. | Haskell, Prolog |
| Programación Orientada a Objetos | Modularidad, reutilización de código, fácil mantenimiento. | Puede agregar complejidad innecesaria, especialmente en programas pequeños o simples. | Java, Python, C++ |
| Programación Orientada a Aspectos | Permite una mejor modularización de aspectos transversales como seguridad o logging. | Puede complicar la estructura del código, haciendo que sea más difícil de seguir y depurar. | AspectJ |
| Programación Funcional | Evita efectos secundarios, facilita la concurrencia, código más predecible y fácil de probar. | Puede ser menos intuitivo para los programadores acostumbrados a otros paradigmas, a veces menos eficiente en términos de rendimiento. | Lisp, Haskell, Erlang |
| Programación Lógica | Adecuado para problemas de lógica y razonamiento automático, fácil de expresar reglas complejas. | Puede ser ineficiente para tareas fuera del ámbito de la lógica pura, difícil de integrar con otros paradigmas. | Prolog |
| Programación Reactiva | Muy útil para sistemas interactivos y que requieren alta capacidad de respuesta, fácil manejo de eventos asíncronos. | Puede ser difícil de depurar y entender, requiere un enfoque diferente en el diseño del software. | RxJava, ReactiveX |
| Programación Basada en Eventos | Ideal para aplicaciones interactivas y en tiempo real, buen manejo de la interacción con el usuario. | Puede ser difícil de escalar, el código puede volverse difícil de seguir si hay demasiados eventos. | JavaScript (en desarrollo web) |

ENGLISH VIDEO LINK: <https://www.canva.com/design/DAGNgyQaBeg/dGGZnlt4uA-WQ6UPjQSUUw/watch?utm_content=DAGNgyQaBeg&utm_campaign=designshare&utm_medium=link&utm_source=editor>

Integrantes del equipo: Laura Aviles, Ximena Contreras, Nicolás Núñez.

LINK REPOSITORIO GITHUB (accesible para miembros del equipo): <https://github.com/nicoprog4/GuiaN.1.git>