TALLER DE PROYECTOS II

“TrainUP! Desarrollo de una aplicación de entrenamiento personal.”

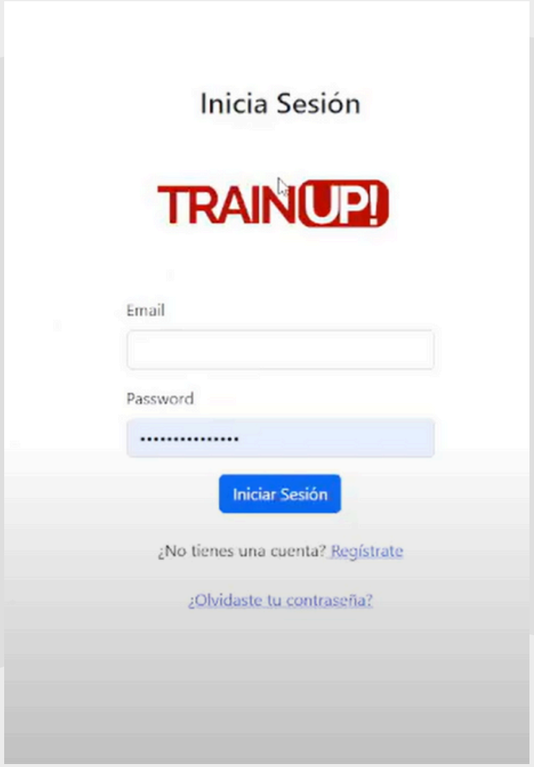
Presentado por:

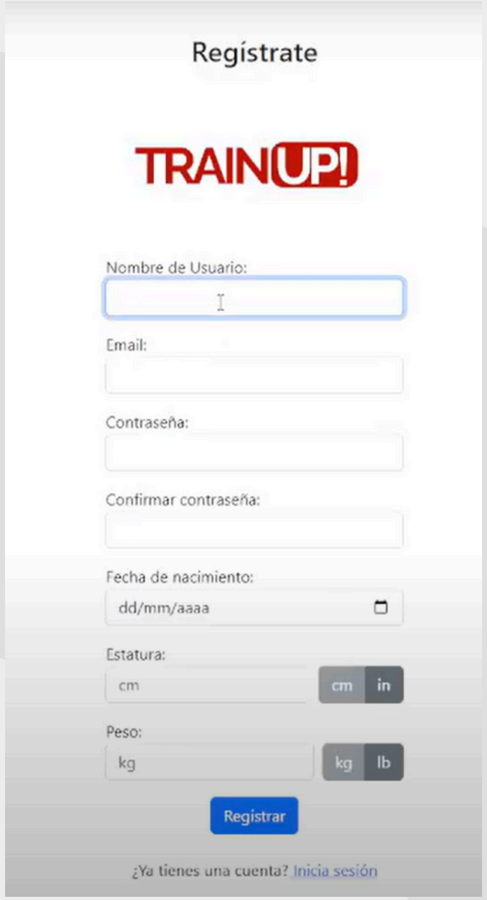
Becerra Atencio Franz
Chahuin Miraval Xiomara Solange
Coronado Huamaní Edilson
Güere Casas Jose Manuel
Hilario Caso Hieyson Steven
Huertos Roque Jean Esteven
Mauricio Orihuela Isaac
Quispe Quispe Florentino Eduardo
Solis Torres Jose Eduardo
Torres de la cruz Jhon Henry
Torres Torres Angela Paola

Docente: Daniel Gamarra Moreno

TRAINUP! SPRINT 1

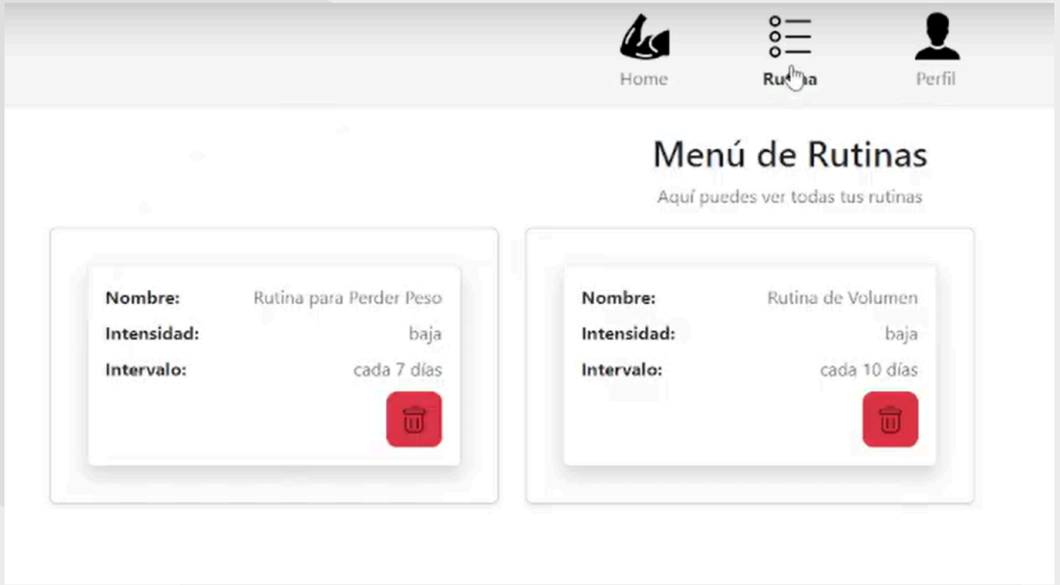
HU01	**Registro de Usuario**
HU02	**Inicio de Sesión**
HU03	**Selección de Nivel de Dificultad**
HU04	**Generación de Rutinas Básicas**

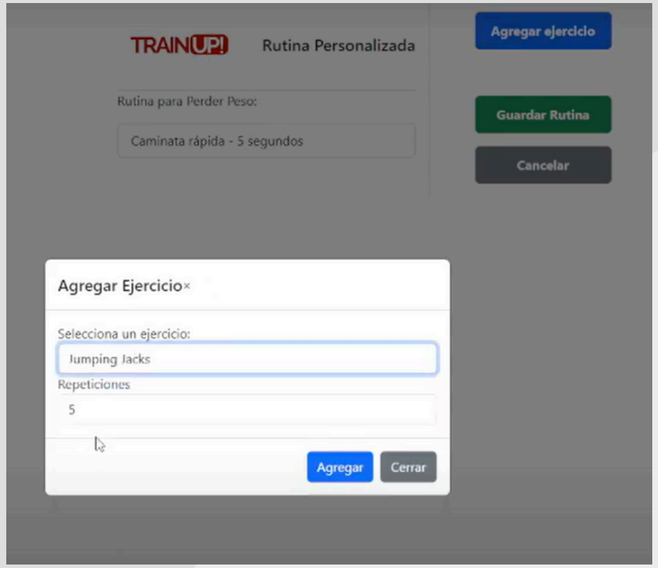




TRAINUP! SPRINT 2

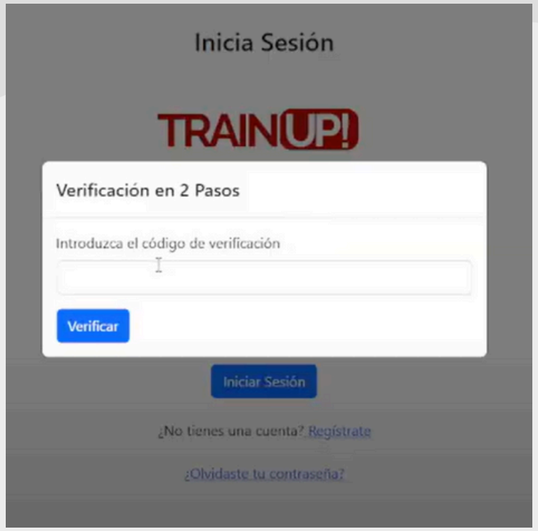
HU05	**Menú de Rutinas**
HU06	**Personalización de Rutinas**
HU07	**Gestión de Preferencias y Programación**
HU08	**Menú de Rutinas Personalizadas**

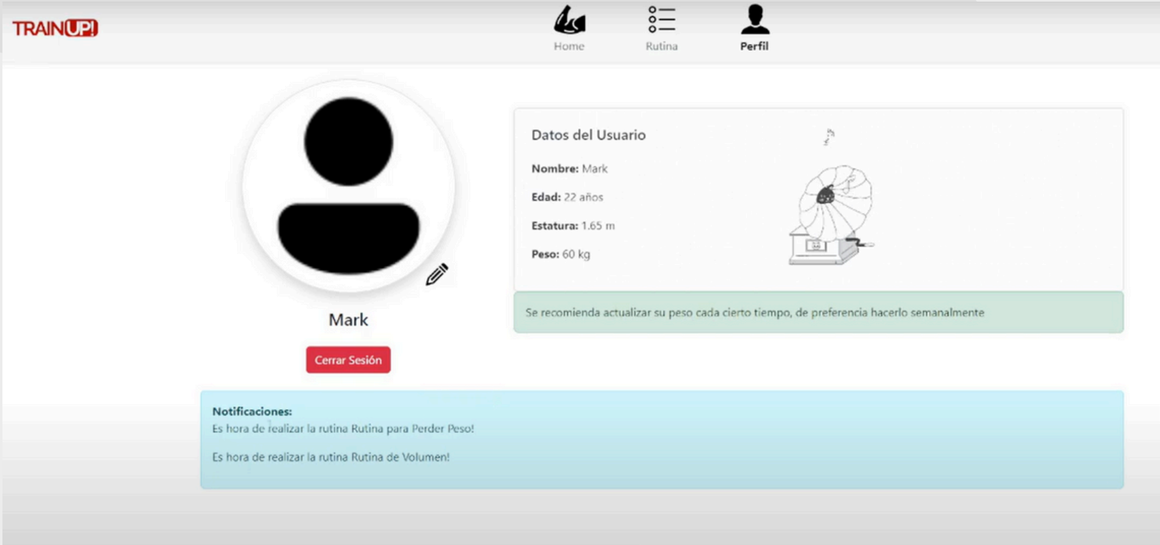




TRAINUP! SPRINT 3

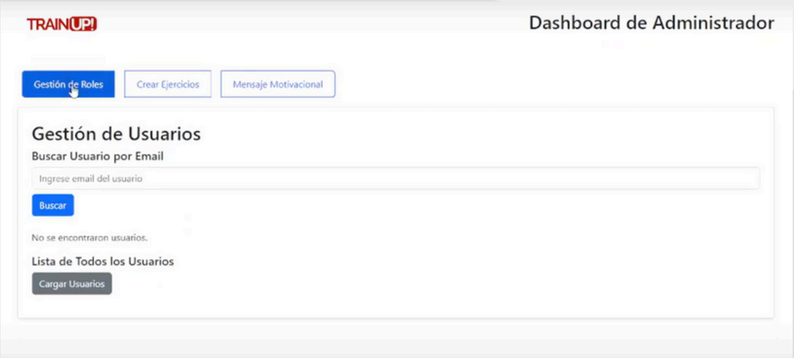
HU09	**Notificaciones**
HU10	**Seguimiento de Progreso**
HU11	**Implementar la previsualización de rutinas básicas según el nivel de dificultad seleccionado**
HU12	**Recuperación de Contraseña**
HU13	**Verificación dos Pasos**

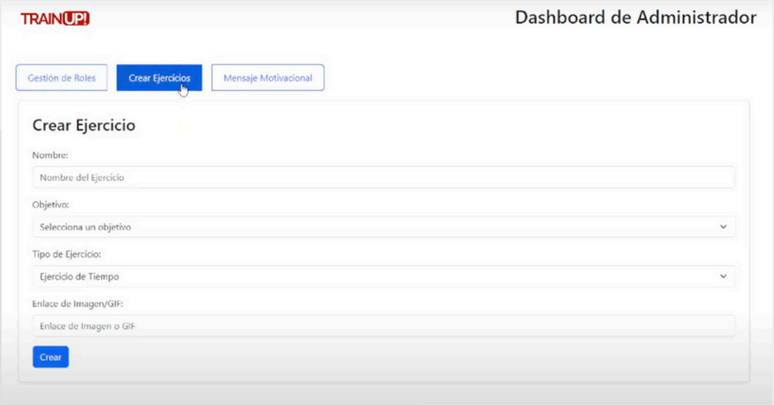


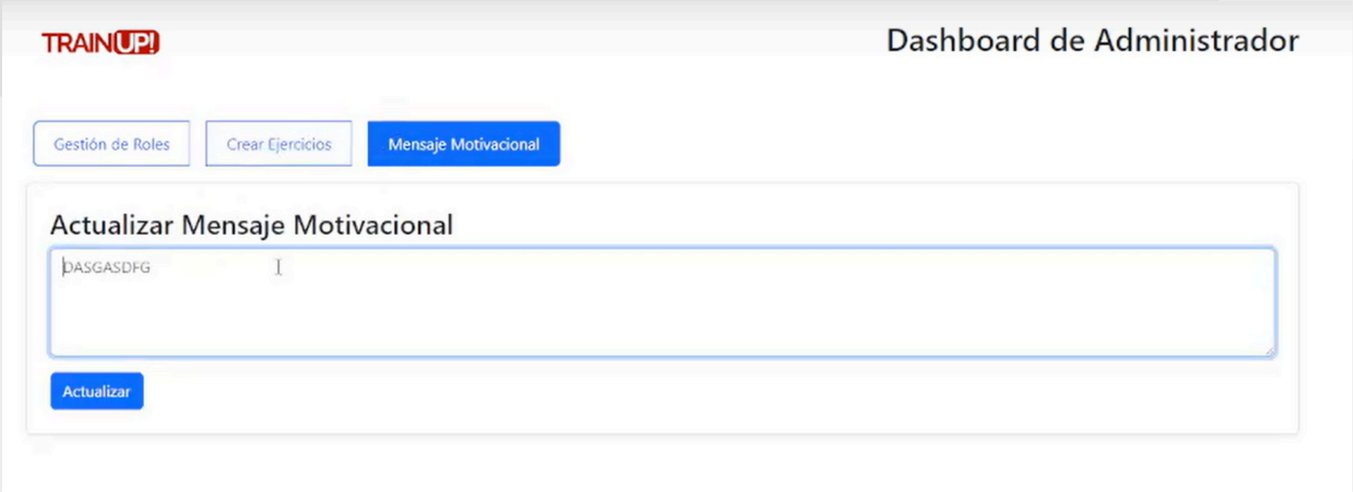


TRAINUP! SPRINT 4

HU14	**Administrar los retos semanales de manera eficiente y permitir a los usuarios seguir su progreso.**
HU15	**Visualizar el panel de retos donde los usuarios puedan acceder a la información de sus desafíos.**
HU16	**Mostrar mensajes motivacionales para apoyar a los usuarios a seguir adelante con sus rutinas.**
HU17	**Gestionar los datos de los participantes, asegurando que toda la información esté organizada y accesible.**
HU18	**Definir los roles y permisos adecuados para gestionar la seguridad y las funciones dentro de la aplicación.**







TALLER DE PROYECTOS II

Integración continua

INTEGRACIÓN CONTINUA

HERRAMIENTAS DE DESARROLLO:

1. Visual Studio Code:
   - Descripción: Un entorno de desarrollo integrado (IDE) ligero y potente.
   - Uso: Utilizado para escribir y depurar el código del proyecto.
2. Git Flow:
   - Descripción: Una extensión de Git que proporciona un modelo de ramificación para el desarrollo.
   - Uso: Para el control de versiones y la gestión de ramas, asegurando que el código esté siempre en un estado desplegable.
3. Docker:
   - Descripción: Una plataforma para desarrollar, enviar y ejecutar aplicaciones dentro de contenedores.
   - Uso: Para la contenedorización, facilitando la replicación del entorno de desarrollo y producción.

LENGUAJES Y FRAMEWORKS

1. Node.js:
- Descripción: Un entorno de ejecución para JavaScript en el servidor.
- Uso: Para el desarrollo del backend de la aplicación.

2. React.js:
- Descripción: Una biblioteca de JavaScript para construir interfaces de usuario.
- Uso: Para el desarrollo del frontend de la aplicación.

3. Express.js:
- Descripción: Un framework web para Node.js.
- Uso: Para construir el backend y manejar las solicitudes HTTP.

4. MongoDB:
- Descripción: Una base de datos NoSQL orientada a documentos.
- Uso: Para almacenar y gestionar los datos de la aplicación.

INTEGRACIÓN CONTINUA

HERRAMIENTAS DE COLABORACIÓN Y GESTIÓN

1. GitHub:

- Descripción: Una plataforma de alojamiento de código fuente y control de versiones.
- Uso: Para alojar el repositorio del proyecto y facilitar la colaboración entre los desarrolladores.

.

HERRAMIENTAS DE DISEÑO Y PROTOTIPADO

- Canva:
- Balsamiq Mockups
- Pencil Project

.

HERRAMIENTAS DE PRUEBAS

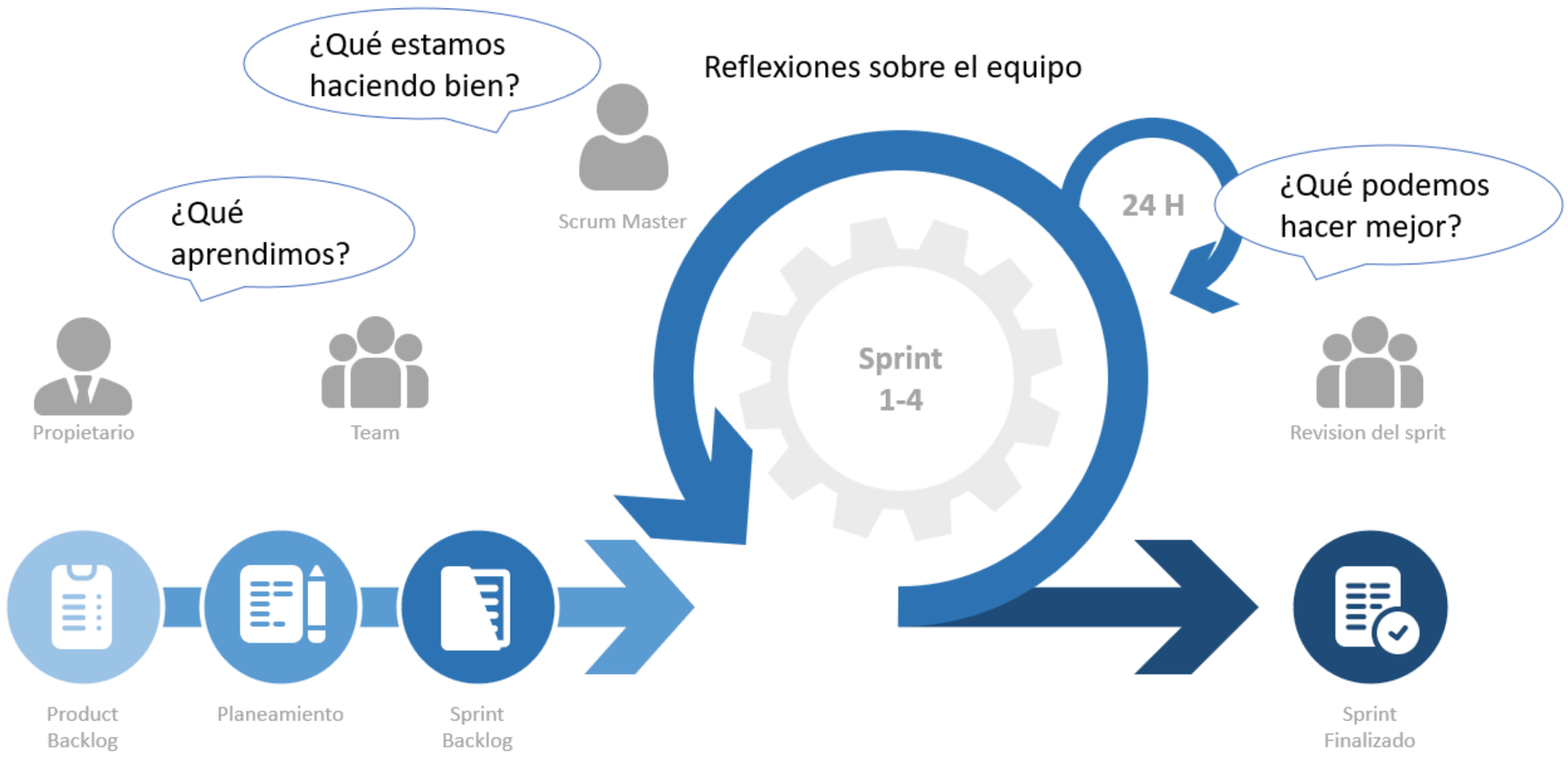
1. Cypress:

- Descripción: Una herramienta de pruebas automatizadas para aplicaciones web.
- Uso: Para realizar pruebas de extremo a extremo (E2E) y asegurar la calidad del software.

2. Postman:

- Descripción: Una herramienta para probar APIs.
- Uso: Para realizar pruebas automáticas de las APIs desarrolladas en el proyecto.

INTEGRACION CONTINUA



Validacion de registro y autenticacion en dos pasos

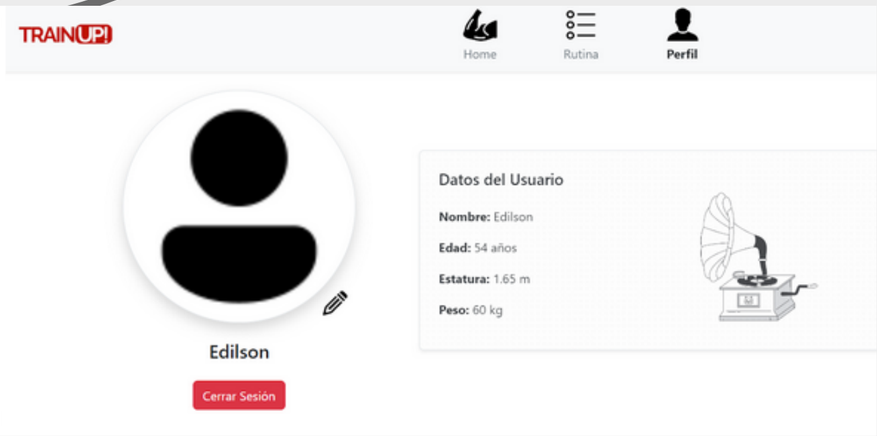
TALLER DE PROYECTOS II

Demostración del proyecto

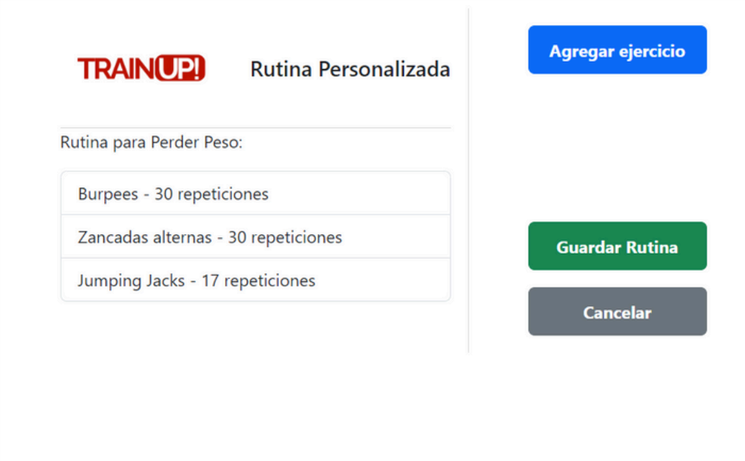
DEMOSTRACIÓN DEL PROYECTO

PRESENTACIÓN DE FUNCIONALIDADES PRINCIPALES:

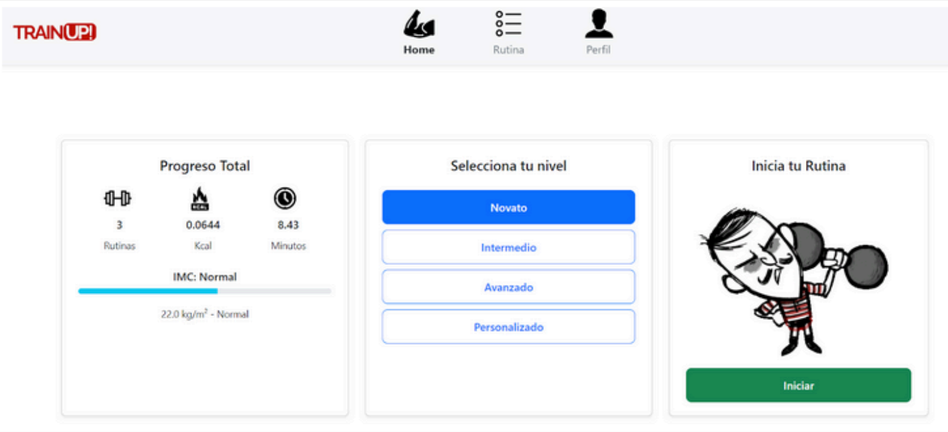
Registro y Seguimiento de Rutinas:



Creación de rutinas Personalizadas:



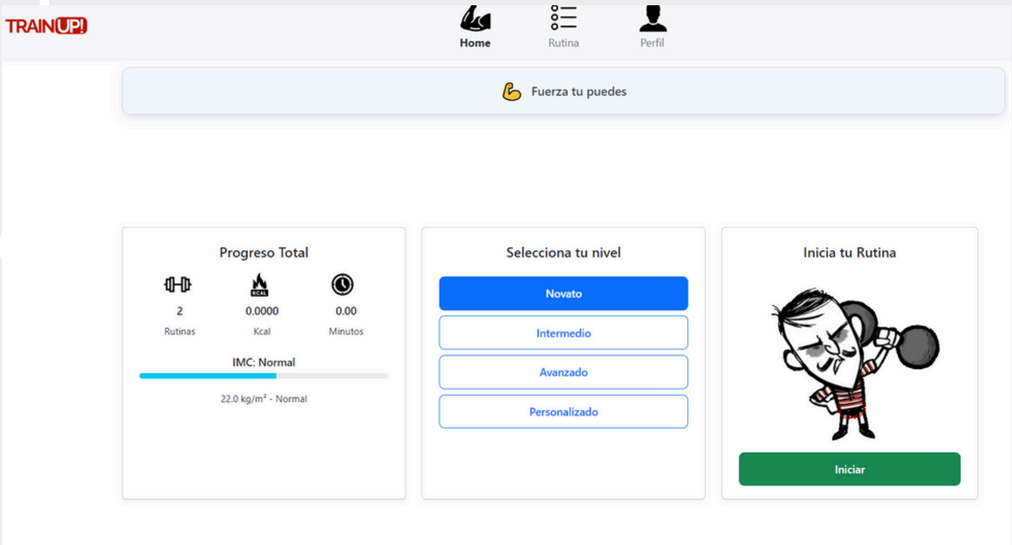
Estadísticas de Progreso:



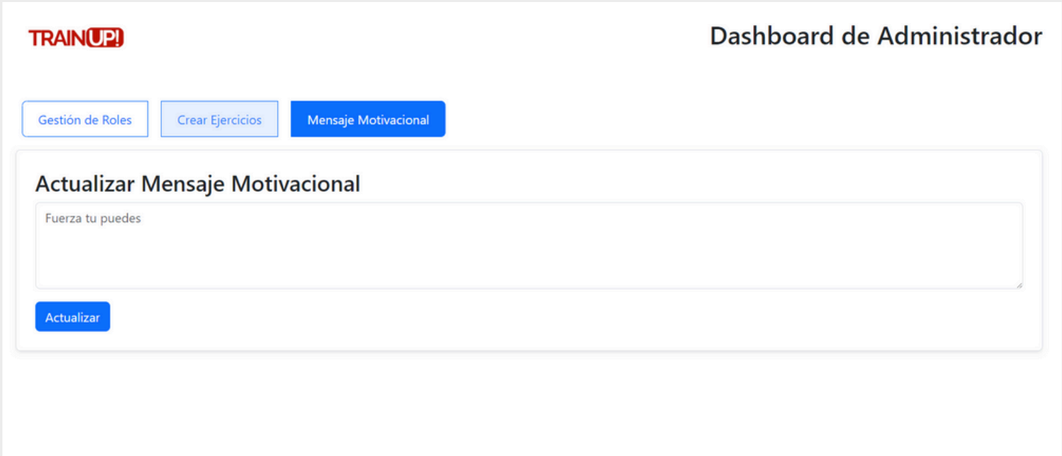
DEMOSTRACIÓN DEL PROYECTO

PRESENTACIÓN DE FUNCIONALIDADES ADICIONALES:

Mensaje motivacional



Dashboard admins



Video de la demostración del proyecto

RECEPCIÓN DE FEEDBACK

Encuestas de satisfacción:

1. ¿Cómo calificarías tu experiencia general con la aplicación TrainUp?
   a. Excelente
   b. Buena
   c. Regular
   d. Mala
2. ¿Qué tan fácil te resultó registrarte e iniciar sesión en la aplicación?
   a. Muy fácil
   b. Fácil
   c. Difícil
   d. Muy difícil
3. ¿La interfaz de usuario es intuitiva y fácil de navegar?
   a. Totalmente de acuerdo
   b. De acuerdo
   c. En desacuerdo
   d. Totalmente en desacuerdo
4. ¿Las rutinas de entrenamiento personalizadas cumplen con tus expectativas y necesidades?
   a. Totalmente de acuerdo
   b. De acuerdo
   c. En desacuerdo
   d. Totalmente en desacuerdo
5. ¿Qué tan útil encuentras las notificaciones y recordatorios de la aplicación?
   a. Muy útiles
   b. Útiles
   c. Poco útiles
   d. Inútiles

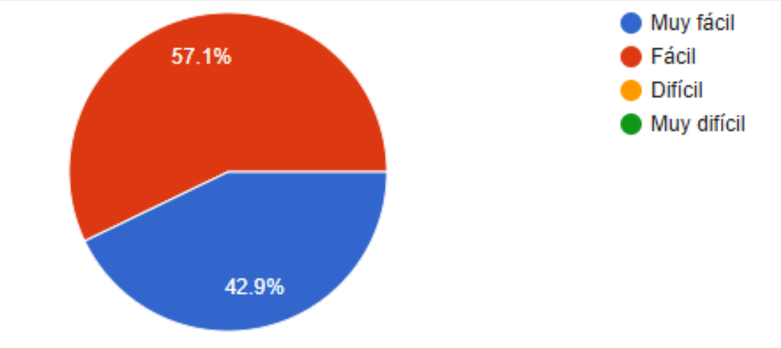
6. ¿Cómo calificarías la funcionalidad de seguimiento de progreso en la aplicación?
   a. Excelente
   b. Buena
   c. Regular
   d. Mala
7. ¿Has experimentado algún problema técnico o error al usar la aplicación?
   a. No, nunca
   b. Raramente
   c. A menudo
   d. Sí, siempre
8. ¿Qué tan satisfecho estás con la calidad de las recomendaciones de ejercicio proporcionadas por la aplicación?
   a. Muy satisfecho
   b. Satisfecho
   c. Insatisfecho
   d. Muy insatisfecho

9 ¿Recomendarías la aplicación TrainUp a otros usuarios?
   a. Definitivamente sí
   b. Probablemente sí
   c. Probablemente no
   d. Definitivamente no

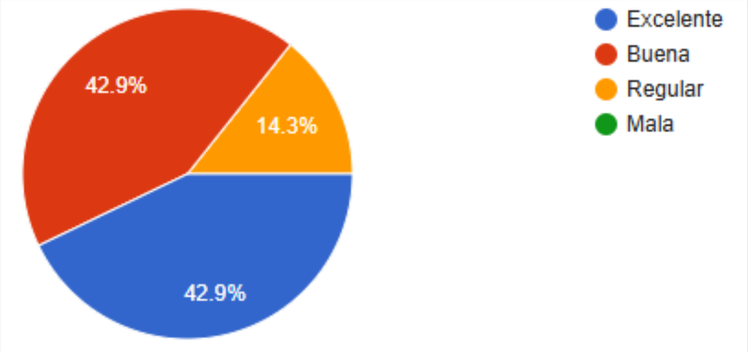
10. ¿Qué mejoras o nuevas funcionalidades te gustaría ver en futuras versiones de la aplicación?
   [Espacio para respuesta abierta]

RESULTADOS

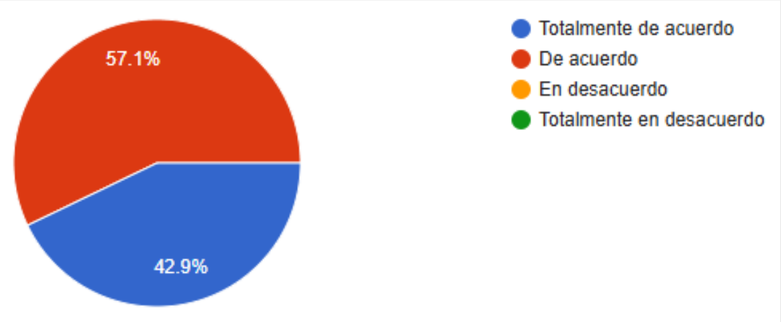
Registro e iniciar sesión



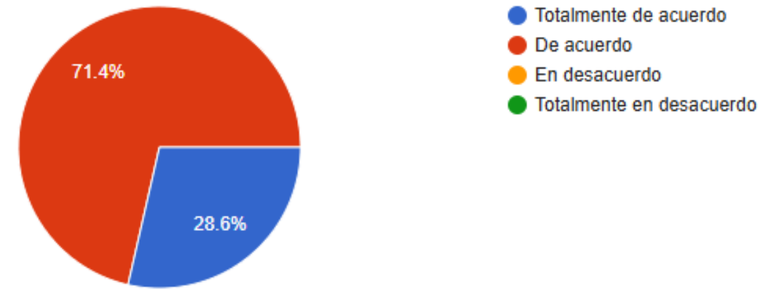
Notificaciones y recordatorios



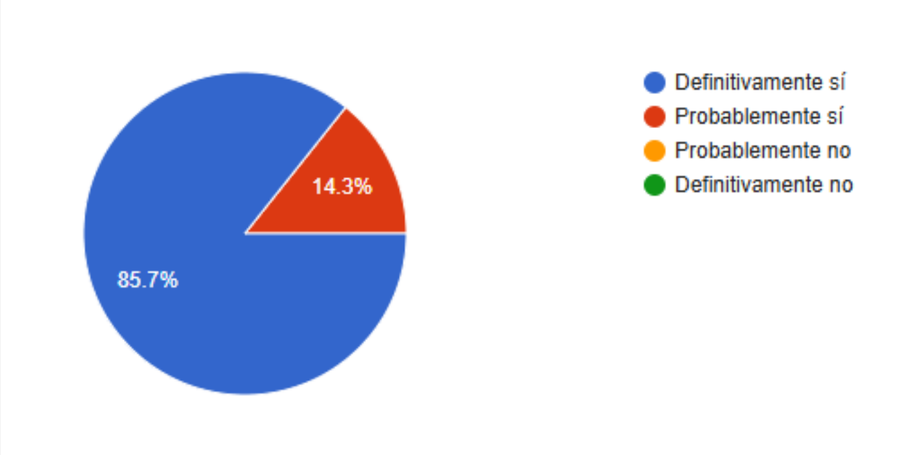
Intuitividad



Rutinas de entrenamiento



¿Recomendarías la aplicación TrainUp a otros usuarios?

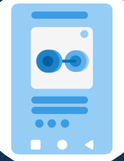


TALLER DE PROYECTOS II

Pruebas y Validación

PRUEBAS

01. La aplicación debe ser compatible con los diferentes navegadores web como chrome, edge, etc.



02. La IU debe ser intuitiva y fácil de usar para los usuarios finales

03. Asegurar que los datos de los usuarios estén protegidos mediante encriptación y autenticación segura.

04. Garantizar tiempos de respuesta rápidos y una experiencia de usuario fluida

PRUEBAS DE CAJA NEGRA

Caso de prueba de caja negra: La aplicación debe ser compatible con los diferentes navegadores web como chrome, edge, etc. para garantizar el uso de este mismo.	
Descripción:	Verificación de que la aplicación sea compatible con los navegadores más usados como chrome, edge, entre otros.
Precondiciones:	• La aplicación debe estar lista para correr en los distintos navegadores web.
Pasos:	• Iniciar la aplicación web en los distintos navegadores web. • Verificar que la aplicación funcione de manera fluida. • Registrar usuarios en la aplicación web • Iniciar sesión con los usuarios registrados. • Iniciar rutinas de entrenamientos. • Editar perfil de usuario. • Cierre de sesión en la aplicación.
Resultado esperado:	• La aplicación funciona correctamente en los distintos navegadores web.
Criterio de validación:	• Se valida que la aplicación funcione de manera fluida en los distintos navegadores web • Se valida que las funciones de la aplicación funcionen correctamente y no haya errores en los procesos.
Resultados:	• El caso de prueba se considera exitoso, ya que la aplicación funciona de manera normal y fluida en los diferentes navegadores web, por la parte de procesos como registro, inicio de sesión, entre otros funcionan como se esperaba.

Caso de prueba de caja negra: La IU debe ser intuitiva y fácil de usar para los usuarios finales	
Descripción:	Verificación de que la interfaz de la aplicación sea amigable, intuitiva y fácil de usar para los usuarios
Precondiciones:	• La aplicación debe estar ejecutándose para su prueba o uso.
Pasos:	• Iniciar la aplicación web en un navegador web. • iniciar sesión. • Interactuar con el software ya sea editando datos, creando rutinas de entrenamiento, comenzando rutinas de entrenamiento etc. • Repetir el 3er paso con diferentes usuarios.
Resultado esperado:	• La UI le parece amigable, fácil de usar e intuitiva a los diferentes usuarios
Criterio de validación:	• Se valida que la interfaz de la aplicación sea amigable, fácil de usar e intuitiva a los diferentes usuarios.
Resultados:	• El caso de prueba se considera exitoso, ya que la interfaz de la aplicación les parece amigable, fácil de usar e intuitiva a los diferentes usuarios.

PRUEBAS DE CAJA NEGRA

Caso de prueba de caja negra: Garantizar que las historias de usuario definidas para el sprint 1 funcionen correctamente	
Descripción:	Asegurar que la aplicación registre usuarios, inicie sesión, seleccione el nivel de dificultad de la rutina a crear, y estas mismas se generen.
Precondiciones:	• La aplicación debe estar ejecutándose para su prueba o uso.
Pasos:	• Iniciar la aplicación web en un navegador web. • Registro de usuario • Inicio de sesión • Seleccionar nivel de rutina • Generar rutina
Resultado esperado:	• Que la aplicación responda de manera exitosa registrando usuarios, también que los usuarios registrados puedan iniciar sesión, que los usuarios seleccionen un nivel de rutina y puedan generar la rutina.
Criterio de validación:	• Se valida que la respuesta de la aplicación con las distintas funciones sean exitosas y rapidas
Resultados:	• El caso de prueba se considera exitoso, ya que la aplicación responde de manera efectiva con las funcionalidades de las historias de usuario que se realizó.

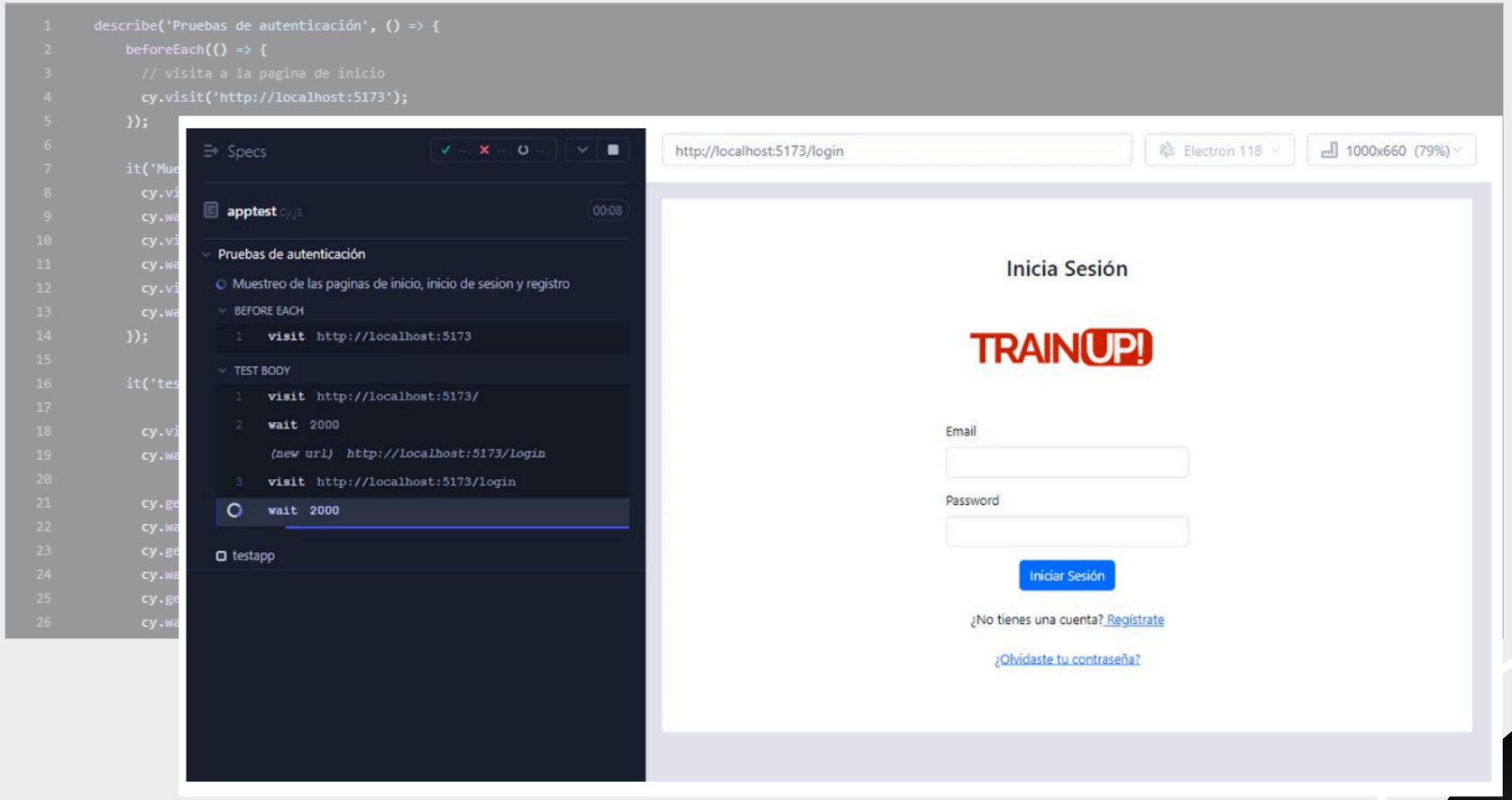
Caso de prueba de caja negra: Garantizar que las historias de usuario definidas para el sprint 2 funcionen correctamente	
Descripción:	Asegurar que la aplicación muestre el menu de rutinas, permita la personalización de rutinas, gestionar preferencias y programarlas para que muestre un menú de rutinas personalizadas por el usuario
Precondiciones:	• La aplicación debe estar ejecutándose para su prueba o uso.
Pasos:	• Visualizar Menú de rutinas • Personalización de rutinas • Gestión de Preferencias y programación • Menú de Rutinas Personalizadas
Resultado esperado:	• Que la aplicación responda de manera exitosa mostrando un menú de rutinas, también que el usuario pueda personalizar las rutinas según su preferencia y programación, así también como la visualización del menú de rutinas personalizadas por el usuario.
Criterio de validación:	• Se valida que la respuesta de la aplicación con las distintas funciones sean exitosas y rapidas
Resultados:	• El caso de prueba se considera exitoso, ya que la aplicación responde de manera efectiva con las funcionalidades de las historias de usuario que se realizó en este sprint.

PRUEBAS DE CAJA NEGRA

Caso de prueba de caja negra: Garantizar que las historias de usuario definidas para el sprint 3 funcionen correctamente	
Descripción:	Asegurar que la aplicación muestre notificaciones, seguimiento de progreso, también previsualizar las rutinas básicas según el nivel de dificultad seleccionado, también el cambio de contraseña y la verificación en dos pasos.
Precondiciones:	• La aplicación debe estar ejecutándose para su prueba o uso.
Pasos:	• Visualizar Notificaciones • Seguimiento de Progreso • Previsualización de rutinas básicas según el nivel de dificultad seleccionado.
Resultado esperado:	• Que la aplicación responda de manera exitosa visualizando notificaciones, seguimiento de progreso, rutinas básicas según el nivel de dificultad seleccionado, permite el cambio de contraseña y la verificación en dos paso.
Criterio de validación:	• Se valida que la respuesta de la aplicación con las distintas funciones sean exitosas y rapidas
Resultados:	• El caso de prueba se considera exitoso, ya que la aplicación responde de manera efectiva con las funcionalidades de las historias de usuario definidas para este sprint que se realizó.

Caso de prueba de caja negra: Garantizar que las historias de usuario definidas para este sprint 4 funcionen correctamente	
Descripción:	Asegurar que la aplicación administre retos semanales de manera eficiente y permita a los usuarios seguir su progreso, visualizar el panel de retos donde los usuarios puedan acceder a la información de sus desafíos, mostrar mensajes motivacionales, gestionar los datos de los participantes, también definir los roles y permisos adecuados para la seguridad de información
Precondiciones:	• La aplicación debe estar ejecutándose para su prueba o uso.
Pasos:	• Administrar los retos semanales de manera eficiente y permitir a los usuarios seguir su progreso. • Visualizar el panel de retos donde los usuarios puedan acceder a la información de sus desafíos. • Mostrar mensajes motivacionales para apoyar a los usuarios a seguir adelante con sus rutinas. • Gestionar los datos de los participantes, asegurando que toda la información esté organizada y accesible. • Definir los roles y permisos adecuados para gestionar la seguridad y las funciones dentro de la aplicación.
Resultado esperado:	• Que la aplicación responda de manera exitosa administrar los retos semanales, visualizar el panel de retos, visualizar los mensajes motivacionales, gestión de datos de los participantes y asignar roles y permisos para la seguridad de la información de los usuarios.
Criterio de validación:	• Se valida que la respuesta de la aplicación con las distintas funciones sean exitosas y rapidas
Resultados:	• El caso de prueba se considera exitoso, ya que la aplicación responde de manera efectiva con las funcionalidades de las historias de usuario que se realizó.

PRUEBAS E2E CON CRYPRESS



Cypress es una herramienta poderosa para realizar pruebas automáticas de aplicaciones web de extremo a extremo

PRUEBAS E2E CON CYPRESS

```
// Registro de datos para la aplicacion
cy.get('#root > div > form > div:nth-child(3) > input').type('testeando ando'); // nombre de usuario
```

```
// CREACION DE RUTINAS
cy.get('#root > div > div.flex-grow-1.p-3 > div > div > div > div:nth-child(2) > div > div > button:nth-child(2)').click({ force: true }); // Selección d
```

```
// CREAR RUTINA PERSONALIZADA
cy.get('#root > div > div.flex-grow-1.p-3 > div > div > div > div:nth-child(2) > div > div > button.btn.btn-primary').scrollIntoView().should('be.visible')
cy.wait(2000):
```

```
// VER RUTINAS CREADAS
cy.get('#root > div > div.navbar.navbar-expand-lg.navbar-light.bg-light.shadow-sm.fixed-top > div > div.d-flex.justify-content-center.align-it
cy.wait(2000);
```

```
// ELEGIR UNA RUTINA Y ENTRENAR
cy.get('#root > div > div.flex-grow-1.p-3 > div > div.row.g-4 > div:nth-child(1) > div').click();
cy.wait(2000);
```

```
// ELIMINAR RUTINAS CREADAS
cy.get('#root > div > div.navbar.navbar-expand-lg.navbar-light.bg-light.shadow-sm.fixed-top > div > div.d-flex.justify-content-center.align-items-cent
cy.wait(2000):
```

```
// EDITAR PERFIL DEL USUARIO
cy.get('#root > div > div.navbar.navbar-expand-lg.navbar-light.bg-light.shadow-sm.fixed-top > div > div.d-flex.justify-content-center.align-items-center.w-
cy.wait(2000);
```

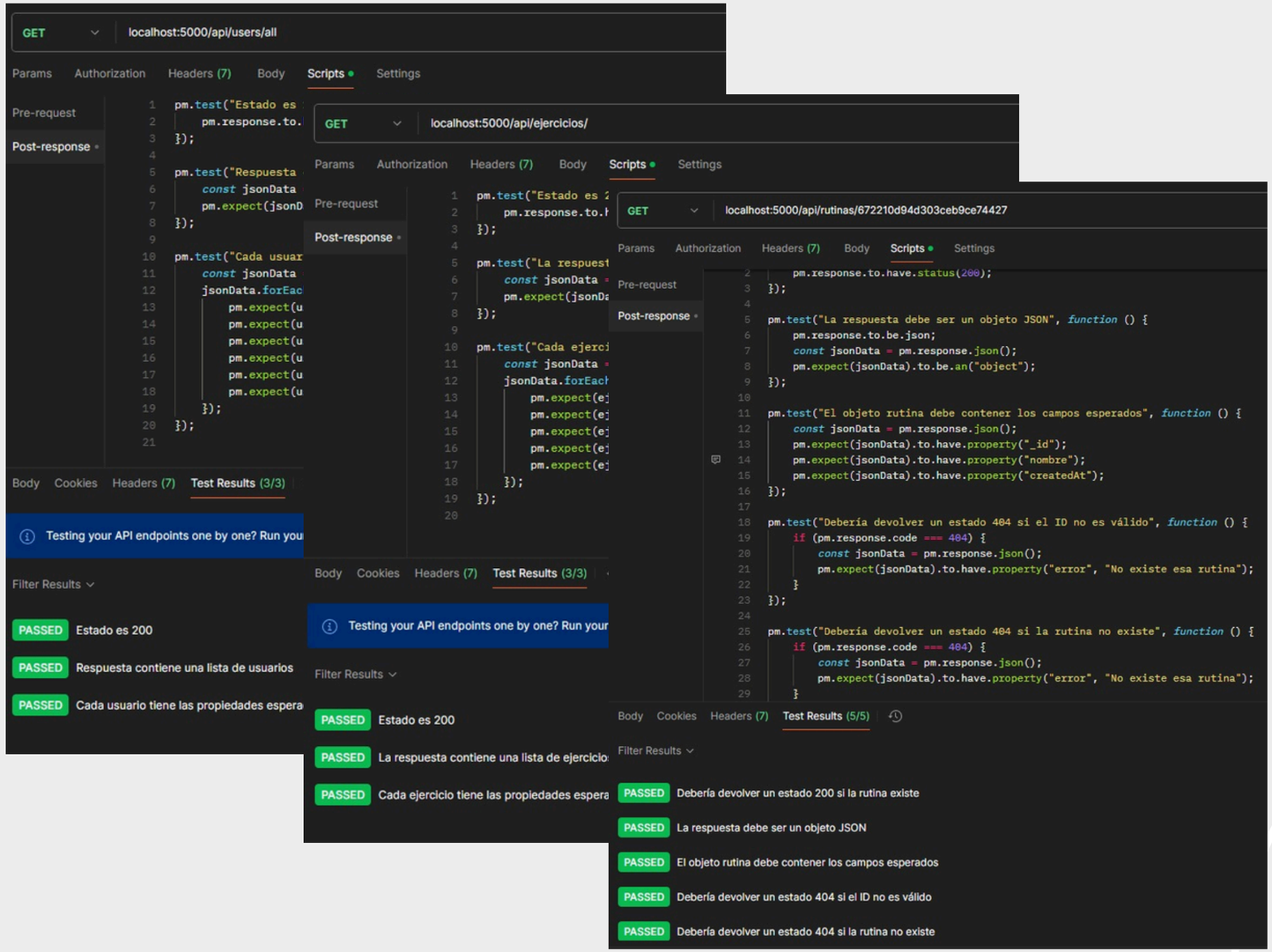
```
// LOGUOT DE LA APLICACION
//Hacer clic en el boton de perfil
cy.get('#root > div > div.navbar.navbar-expand-lg.navbar-light.bg-light.shadow-sm.fixed-top > div > div.d-flex.justify-content-center.align-items-center
cy.wait(2000);
cy.get('#root > div > div.navbar.navbar-expand-lg.navbar-light.bg-light.shadow-sm.fixed-top > div > div.d-flex.justify-content-center.align-items-center

// Hacer clic en el botón de salida
cy.get('#root > div > div.flex-grow-1.p-3 > div > div > div.col-md-4.text-center > button').click();
cy.wait(1000);

cy.url().should('eq', 'http://localhost:5173/login'); // Verifica que el usuario haya sido redirigido a la página de inicio de sesión

cy.get('.user-info').should('not.exist'); // Verifica que el contexto de usuario esté null
);
```

PRUEBAS E2E CON CYPRESS



Postman es una herramienta poderosa para realizar pruebas automáticas de APIS.

RECOMENDACIÓN

- **Capacitación continua:** Mantener al equipo actualizado en tecnologías emergentes como Docker, React Native, y otras relacionadas al stack MERN.
- **Implementación de un sistema de monitoreo:** Integrar herramientas avanzadas para la detección proactiva de fallos y áreas de mejora.
- **Retroalimentación periódica:** Realizar encuestas y análisis de métricas para ajustar las funcionalidades a las necesidades del mercado.
- **Fortalecimiento de la seguridad:** Ejecutar auditorías regulares y aplicar estándares como OWASP para proteger la información de los usuarios.
- **Ampliación de compatibilidad:** Extender la aplicación a más plataformas, como tablets y sistemas operativos menos convencionales.
- **Análisis de uso:** Implementar mapas de calor y análisis detallados para rastrear patrones de interacción con la aplicación.
- **Optimización del diseño de interfaz:** Mejorar la experiencia del usuario con un diseño más intuitivo y accesible.

CONCLUSION

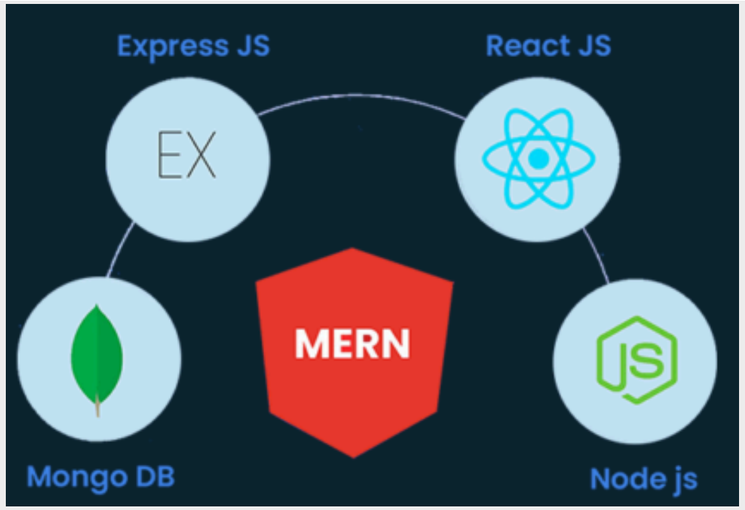
- TrainUP! ha optimizado la gestión de rutinas de entrenamiento, mejorando la interacción entre entrenadores y clientes. Esto se ha logrado mediante la implementación de un sistema digital que facilita el seguimiento del progreso y la personalización de las rutinas, cumpliendo con estándares internacionales como ISO/IEC 25010.

- Experiencia del usuario mejorada: La implementación de tecnologías como MERN y Docker ha mejorado la experiencia del usuario en un 30%, garantizando flexibilidad y eficiencia.

- Reducción de errores y mayor satisfacción: La digitalización del sistema ha reducido el margen de error en un 20%, lo que ha contribuido a una mayor satisfacción de los usuarios, incrementando su satisfacción en un 25%.

- Cumplimiento de requerimientos funcionales y no funcionales: TrainUP! ha cumplido con los requerimientos funcionales (como la personalización de rutinas y notificaciones) y no funcionales (seguridad, usabilidad), lo que ha contribuido a una experiencia de usuario más completa y satisfactoria.

MANUAL TECNICO APLICACIÓN TRAINUP



ESTRUCTURA DEL PROYECTO

La estructura del proyecto sigue el patrón de arquitectura de tres capas. A continuación se describe la organización de los directorios principales:

1. Backend (Servidor)**:
   - `index.js`: Archivo principal donde se configura el servidor Express y se conecta a MongoDB.
   - `models/`: Contiene los modelos de Mongoose, incluyendo `User.js`, `Rutina.js` y otros.
   - `routes/`: Contiene las rutas de la API para gestionar las solicitudes HTTP,
   - `controllers/`: Contiene la lógica para manejar las rutas, como la creación o actualización de un usuario.

2. Frontend (Cliente):
   - `src/`: Carpeta principal con todos los componentes de React.
   - `components/`: Componentes reutilizables como `FragmentPerfil.js`, `FrameRunEjercicio.js` y otros.
   - `App.js`: Componente principal donde se maneja la navegación con React Router.

3. Base de Datos:
   - `MongoDB`: Utiliza MongoDB como base de datos NoSQL para almacenar información sobre rutinas y usuarios.

INSTALACIÓN DE Mongo

```
npm install mongoose
```

INSTALACIÓN DE Express

```
npm install express
```

INSTALACIÓN DE React

```
npx create-react-app nombre-de-tu-aplicacion
```

INSTALACIÓN DE React router

```
npm install react-router-dom
```

LEVANTAMIENTO DE BASE DE DATOS

```
docker compose up -d
```

INICIALIZACIÓN DEL PROYECTO

```
npm run dev
```

ACTUALIZACIONES Y MANTENIMIENTO

Procedimientos para realizar actualizaciones.

- **Verificación de la Versión Actual:**
  - Identificar la versión actual instalada.
  - Comprobar la disponibilidad de nuevas actualizaciones.
- **Copia de Seguridad:**
  - Realizar una copia de seguridad completa de la base de datos y archivos relevantes antes de comenzar la actualización.
- **Acceso a Recursos de Actualización:**
  - Consultar los contactos oficiales de "Trainup" o el portal de soporte para obtener acceso a las últimas actualizaciones.
- **Desactivación Temporal del Sistema:**
  - Notificar a los usuarios sobre la inminente actualización.
  - Desactivar temporalmente el acceso al sistema para evitar cambios concurrentes durante la actualización.

MANUAL TECNICO APLICACIÓN TRAINUP

LINK

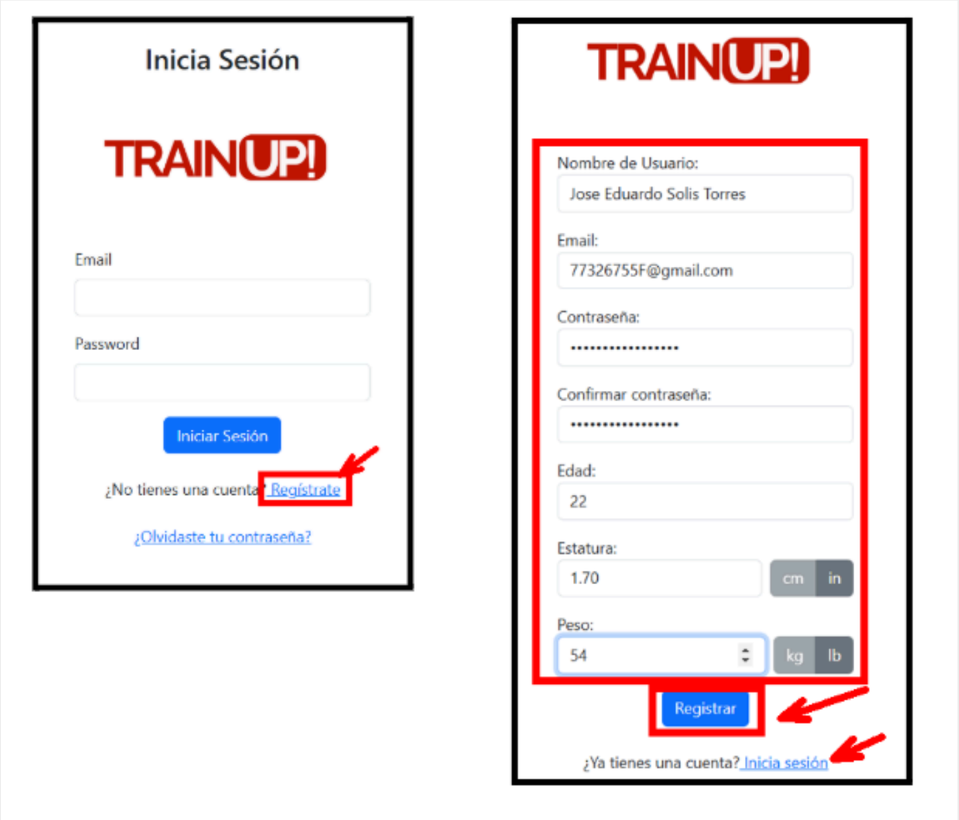
Manual Técnico

Versión 1.0.0.0.0.0
Fecha de Publicación: 27/11/2024
Huancayo, Junín, Perú

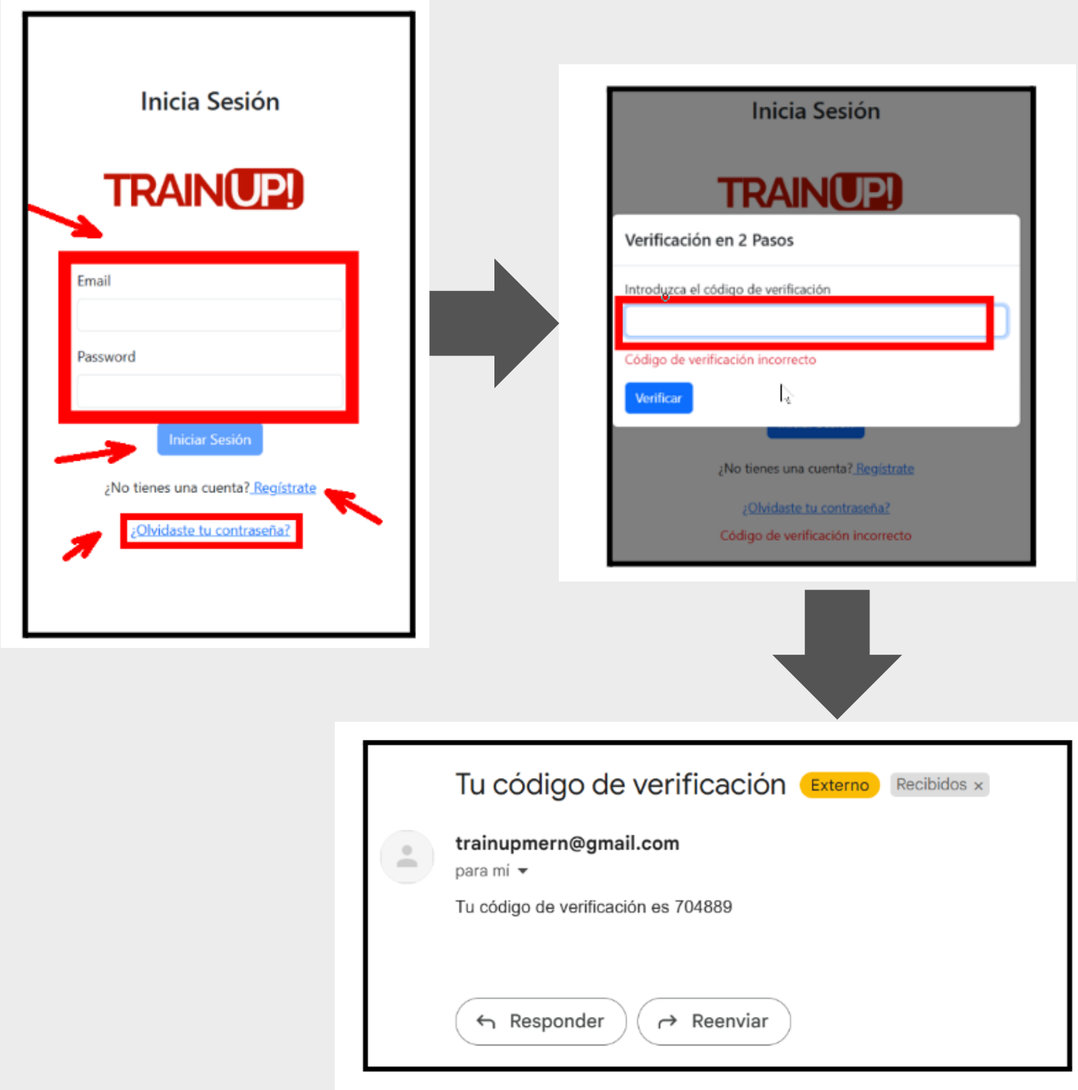
MANUAL DE TÉCNICO
APLICACIÓN TRAINUP

MANUAL DE USUARIO APLICACIÓN TRAINUP

Registro de Usuario: Configuración Inicial de Perfil

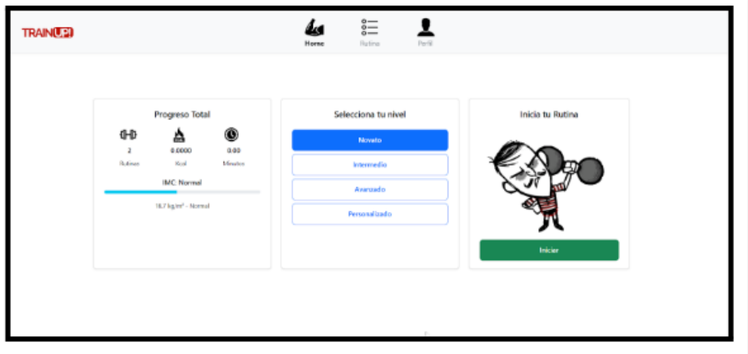


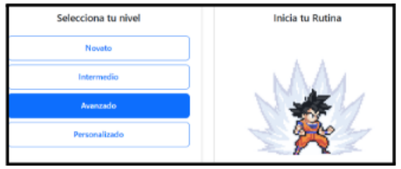
Inicio de Sesión: Acceso a tu Cuenta Trainup

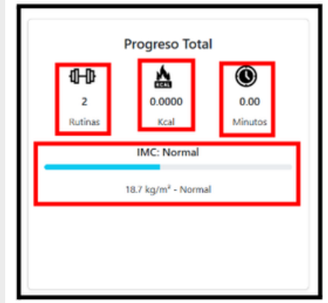


MANUAL DE USUARIO APLICACIÓN TRAINUP

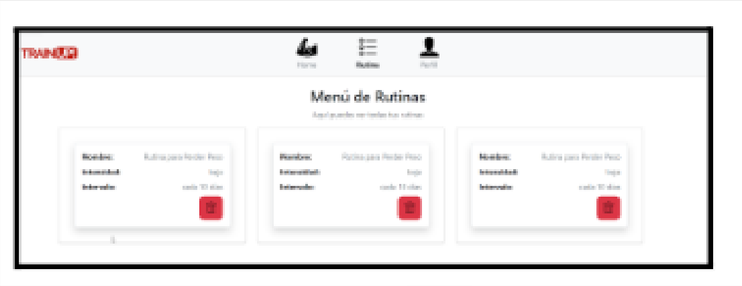
Menú Principal: Tu Panel de Control Personalizado y inicio de Rutina Rapida

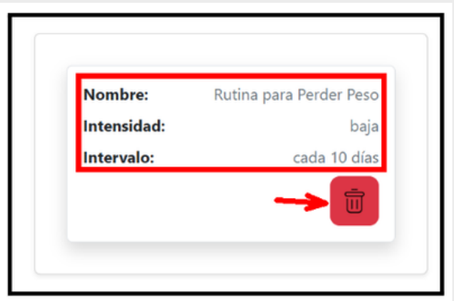






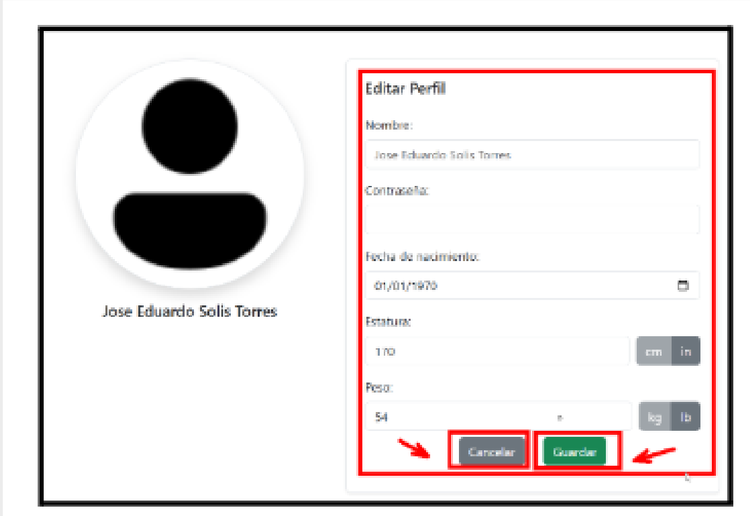
Creación de Rutinas: Personaliza tu Plan de Ejercicios





MANUAL DE USUARIO APLICACIÓN TRAINUP

Perfil de Usuario: Personaliza y Administra tu Información, Cierre de sesion



MANUAL DE USUARIO APLICACIÓN TRAINUP

LINK

Manual Usuario

Versión 1.0.0.0.0.0
Fecha de Publicación: 27/11/2024
Huancayo, Junín, Perú

MANUAL DE USUARIO
APLICACIÓN TRAINUP

GRACIAS