

## README.md

# FUNDAMENTOS DE INGENIERIA DE SOFTWARE - OBLIGATORIO 1

---

## Estudiantes:

- Laura Collazo - 252803
- Mario Ricca - 212393
- Gaston Salgado - 212387

## REPOSITORIOS GIT

---

### Creación de repositorios locales y remotos

Se creo un repositorio en GitHub y luego cada integrante del equipo lo clono, utilizando el comando `git clone` y la url. Asi cada uno trabajar sobre el repositorio local, y luego se suben los cambios al repositorio remoto con el comando `git push`. Siempre antes de comentzar a trabajar en el repositorio local, se debe hacer un `git pull` para traer los cambios que se hayan hecho en el repositorio remoto.

### Comandos Git ejecutados desde el terminal y desde el IDE

- `git clone [URL del repo de GitHub]`

Inicializa un directorio existente como un repositorio Git local.

- `git status`

Muestra el estado del árbol de trabajo.

- `git add [nombre del archivo] o git add .`

Agrega el/los archivo/s al área de staging para el siguiente commit.

- `git commit -m "mensaje"`

Registra los cambios que se encuentran en el área de stagging.

- `git pull`

Incorpora los cambios remotos al repositorio local.

- **git push**

Envia los cambios locales al repositorio remoto.

- **git branch**

Lista las ramas existentes en el repositorio.

- **git branch [nombre de la rama]**

Crea una nueva rama con el nombre indicado.

- **git checkout [nombre de la rama]**

Cambia de rama, a la indicada, y actualiza el directorio de trabajo.

- **git merge [nombre de la rama]**

Merkea la rama indicada con la rama actual.

- **git log**

Muestra el historial de commits.

- **git fetch**

Descarga todas las ramas del repositorio remoto al repositorio local.

## VERSIONADO

---

### Aplicar buenas prácticas de versionado

Aplicar buenas prácticas de versionado en el desarrollo de software implica mantener un control sistemático y ordenado de los cambios realizados en el código fuente y en otros elementos del proyecto a lo largo del tiempo. Algunas implicaciones de esto son:

1. Control de cambios: Las buenas prácticas de versionado permiten realizar un seguimiento de los cambios que se realizan en el código fuente y otros elementos del proyecto. Esto ayuda a los desarrolladores a comprender los cambios realizados en el pasado y a mantener una visión general del estado actual del proyecto.
2. Colaboración: Las herramientas de control de versiones, como Git, permiten que múltiples desarrolladores trabajen en el mismo proyecto simultáneamente. El uso de buenas prácticas de versionado ayuda a evitar conflictos y a mantener la colaboración fluida y organizada.

3. **Reversión:** Las buenas prácticas de versionado permiten a los desarrolladores revertir los cambios realizados en el código fuente y otros elementos del proyecto en caso de que se produzca un error o un problema inesperado. Esto permite una recuperación rápida y eficiente del proyecto en caso de que algo salga mal.
4. **Auditoría:** Las buenas prácticas de versionado proporcionan un historial completo de los cambios realizados en el proyecto, lo que permite a los desarrolladores y otros interesados en el proyecto realizar una auditoría del código fuente y otros elementos. Esto ayuda a garantizar la calidad del código y a detectar cualquier problema potencial.

En resumen, aplicar buenas prácticas de versionado en el desarrollo de software ayuda a mantener un control ordenado y sistemático de los cambios realizados en el proyecto, fomenta la colaboración y permite la recuperación rápida y eficiente del proyecto en caso de problemas.

## Uso de ramas separadas de 'main'

Utilizamos dos ramas principales, develop y main. La rama main se utiliza únicamente para pasar a producción, y la rama develop se utiliza para el desarrollo del proyecto. Cada vez que se finaliza una funcionalidad, que el equipo aprueba, se hace un merge de la rama develop a la rama main, y se sube a producción.

## Resumen de commits y evolución del proyecto

5/8/23, 4:04 PM

Commits · ORT-FIS-2022S2/Obligatorio1-Collazo-Ricca-Salgado

ORT-FIS-2022S2 / Obligatorio1-Collazo-Ricca-Salgado

Private

<> Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

Se

main

Commits on May 8, 2023

Merge branch 'main' of <https://github.com/ORT-FIS-2022S2/proyecto-LCMCR...> ...  
Laura Collazo committed 3 minutes ago

Merge branch 'develop' into main  
Laura Collazo committed 3 minutes ago

Técnicas aplicadas y aprendizajes  
Laura Collazo committed 5 minutes ago

Added: reflexion individual Gastón  
gas234567890 committed 15 minutes ago

Formato Reflexion  
Laura Collazo committed 6 hours ago

Merge branch 'develop' of <https://github.com/ORT-FIS-2022S2/proyecto-...> ...  
Laura Collazo committed 6 hours ago

Reflexion indiv. Laura  
Laura Collazo committed 6 hours ago

correccion de formato observaciones  
Mario committed 6 hours ago

commit mario conclusiones  
Mario committed 6 hours ago

commit observaciones  
Mario committed 6 hours ago

Commits on May 7, 2023

Agregados modelo y usuarios  
Mario committed yesterday

Commits on May 6, 2023

Added: resto de casos de uso detallados, validacion y verificacion  
gas234567890 committed 2 days ago

Prueba tabla  
gas234567890 committed 2 days ago

Corrección en formato en linea 326  
gas234567890 committed 2 days ago

Added: User Stories  
gas234567890 committed 2 days ago

<https://github.com/ORT-FIS-2022S2/Obligatorio1-Collazo-Ricca-Salgado/commits/main>

1/3



localhost:6419

4/24

5/8/23, 4:04 PM

Commits · ORT-FIS-2022S2/Obligatorio1-Collazo-Ricca-Salgado

Commits on May 4, 2023

<b>Modificacion formato</b> Laura Collazo committed 4 days ago
<b>Se agrega UI de form contacto</b> Laura Collazo committed 4 days ago
<b>Add files via upload</b>  LCMCR committed 4 days ago
<b>Carga UI's via link</b> Laura Collazo committed 4 days ago
<b>Add files via upload</b>  LCMCR committed 4 days ago
<b>Cambios formato</b> Laura Collazo committed 4 days ago
<b>Prueba, agregar bocetos UI</b> Laura Collazo committed 4 days ago
<b>Se agregan RF y RNF</b> Laura Collazo committed 4 days ago
<b>Se agregan secciones dentro de Elicitacion</b> Laura Collazo committed 4 days ago

Commits on May 3, 2023

<b>Se agrega detalle Entrevistas</b> Laura Collazo committed 5 days ago
--

Commits on Apr 25, 2023

<b>Actualizacion versionado y layout gral</b> Laura Collazo committed 2 weeks ago
--

Commits on Apr 24, 2023

<b>Se agrega seccion BP Versionado</b> Laura Collazo committed 2 weeks ago
<b>Mas cambios de formato</b> Laura Collazo committed 2 weeks ago
<b>Cambios de formato (markdown)</b> Laura Collazo committed 2 weeks ago
<b>Se agrega la seccion Repos Git</b> Laura Collazo committed 2 weeks ago

Commits on Mar 28, 2023

<b>first commit gaston</b>  gas234567890 committed on Mar 28
<b>prueba develop mario</b>

5/8/23, 4:04 PM

Commits · ORT-FIS-2022S2/Obligatorio1-Collazo-Ricca-Salgado

Mario committed on Mar 28

Prueba Laura

Laura Collazo committed on Mar 28

commit de prueba 1 mario

Mario committed on Mar 28

🔗 Commits on Mar 23, 2023

Add online IDE url

 github-classroom[bot] committed on Mar 23

Newer

Older

# ELICITACION

## Evidencias de actividades de investigación

## Entrevistas

Se realizaron dos entrevistas:

Entrevista 1: se realizó a una madre de tres niños, de los cuales solo uno sigue siendo estudiante de primaria.

Entrevista 2: padre de dos niños, ambos niños practican el judaísmo y uno tiene celiaquía.

Ambos padres envían a sus hijos a escuelas públicas de turno completo donde reciben un almuerzo.

La primera conclusión a la que se llegó es que el sistema de comedores en escuelas públicas funciona bien en su objetivo de alimentar a los niños. De la primera entrevista se vio que en la escuela pública muchas veces el menú semanal no es comunicado con antelación lo cual genera cierta dificultad al preparar las compras en la casa para evitar que sus hijos comieran en sus casas de noche lo mismo que en la escuela.

Por otro lado, se aprendió que la Dirección General de Educación Inicial y Primaria (DGEIP) no puede, por un tema de escala y costos, garantizar opciones para todas las restricciones dietarias, en particular condiciones de carácter filosófico sean religiosas o éticas. En estos casos se le brinda al niño alimentos adecuados dentro de lo posible y se les pide a los padres en esos casos que complementen la alimentación ya sea llevando sus propias viandas a la escuela para cubrir total o parcialmente el almuerzo dependiendo del caso.

Ambos padres encontraron difícil la comunicación con los centros escolares, ya que, a pesar de elaborarse la comida en los centros educativos, la planificación es llevada a cabo para todas las escuelas por un equipo de nutricionistas contratado por la DGEIP

## Observaciones

Al estudiar el funcionamiento de un comedor escolar llegamos a las siguientes conclusiones

- La escuela tiene un sistema de comedor bastante organizado y limpio, con personal encargado de la preparación y distribución de los alimentos.
- Los niños están sentados en mesas largas, compartiendo el espacio con otros estudiantes de diferentes edades, fomentando la socialización y el compañerismo durante la comida.
- Sin embargo, se observa que hay poca variedad de alimentos en el menú, lo que podría resultar monótono y poco atractivo para los niños a largo plazo.
- Además, se nota la falta de opciones para niños con restricciones dietarias, ya sea por motivos religiosos, éticos o médicos.

- No se comunica con suficiente antelación el menú semanal a los padres, lo que dificulta la planificación de las compras de alimentos en casa para aquellos padres que desean complementar la dieta de sus hijos en casa.
- Aunque las recetas son sencillas los encargados del comedor no tienen acceso a la receta con antelación, lo que podría llegar a dificultar la preparación correcta de los platos.
- También se observa que no hay una forma clara para que los padres se comuniquen con el equipo del comedor para informar sobre restricciones dietarias o dar retroalimentación sobre la dieta de sus hijos.

## Lluvia de ideas

A raíz de las entrevistas y luego de discutir el alcance de la aplicación decidimos hacer una lluvia de ideas para ver que funcionalidades debería tener la app.

- Visualización del menú semanal con anticipación para los padres y los estudiantes.
- Plataforma de intercambio entre los padres y los encargados del comedor para dar retroalimentación sobre los alimentos y comunicar restricciones dietarias.
- Funcionalidad para que los padres puedan enviar sus comentarios y sugerencias sobre el menú de la semana y los alimentos servidos en el comedor.
- Notificaciones sobre posibles cambios en el menú semanal o cancelaciones del servicio de comedor.
- Sistema de registro para informar sobre restricciones dietarias de los estudiantes.
- Información nutricional sobre los alimentos servidos en el comedor.
- Posibilidad de solicitar viandas especiales para estudiantes con restricciones dietarias.
- Espacio para que los estudiantes puedan compartir sus opiniones sobre los alimentos servidos en el comedor y dar retroalimentación.

## Referencias a fuentes de información

## Caracterización de usuarios: User Personas



NAME

David



Quote

“

David es un padre trabajador y comprometido con la alimentación saludable de sus hijos. Su hijo mayor tiene celiaquía, lo que ha sido un desafío en términos de asegurar que reciba comidas sin gluten en la escuela. David también quiere asegurarse de que sus hijos reciban comidas saludables y equilibradas durante el día, ya que a menudo comen en la escuela debido a sus horarios ocupados.

”

Background

David es un padre trabajador y comprometido con la alimentación saludable de sus hijos. Su hijo mayor tiene celiaquía, lo que ha sido un desafío en términos de asegurar que reciba comidas sin gluten en la escuela. David también quiere asegurarse de que sus hijos reciban comidas saludables y equilibradas durante el día, ya que a menudo comen en la escuela debido a sus horarios ocupados.

Demographic

♂ Male 42 years

Married

Director de Comunicaciones

3 Cantidad de Hijos

2 Hijos en Escuela

Objetivos

David desea una aplicación que le permita informar sobre la restricción dietética de su hijo celiaco y solicitar viandas especiales. También desea ver el menú semanal del comedor escolar con anticipación para planificar las comidas en casa y asegurarse de que sus hijos estén recibiendo una alimentación saludable y equilibrada durante el día.

Frustraciones

David ha encontrado difícil la comunicación con el centro educativo en cuanto a la restricción dietética de su hijo celiaco. También ha sido difícil planificar las comidas de sus hijos en función del menú del comedor debido a la falta de comunicación anticipada del mismo.

Dispositivos




UXPRESSIA

This persona was built in uxpressia.com

NAME

Maria



Quote

“  
Me gustaría tener una herramienta que me permita estar informada sobre lo que comen mis hijos en el comedor escolar y poder dar retroalimentación sobre su dieta de manera rápida y fácil  
”

Background

María siempre ha sido muy consciente de la importancia de una alimentación saludable y equilibrada. A menudo cocina en casa y prepara almuerzos y meriendas saludables para sus hijos. Sin embargo, ha tenido dificultades para planificar las comidas de sus hijos cuando comen en la escuela porque el menú semanal no se comunica con anticipación. Esto ha resultado en que sus hijos coman lo mismo en casa y en la escuela, lo que le preocupa que puedan estar comiendo en exceso ciertos alimentos. Además, ha tenido dificultades para comunicar las restricciones dietéticas de su hija, que es alérgica a los frutos secos.

Demographic

Female

36

years

Single

Diseñadora Gráfica

2

Cantidad de Hijos

2

Hijos en Escuela



Objetivos



María desea una aplicación que le permita ver el menú semanal del comedor escolar con anticipación para poder planificar las comidas de sus hijos en casa. También desea una plataforma para comunicar las restricciones dietéticas de su hija y recibir notificaciones sobre posibles cambios en el menú.

Frustraciones

María ha encontrado difícil la comunicación con el centro educativo en cuanto a la restricción dietética de su hija. También ha sido difícil planificar las comidas de sus hijos en función del menú del comedor debido a la falta de comunicación anticipada del mismo.

Dispositivos

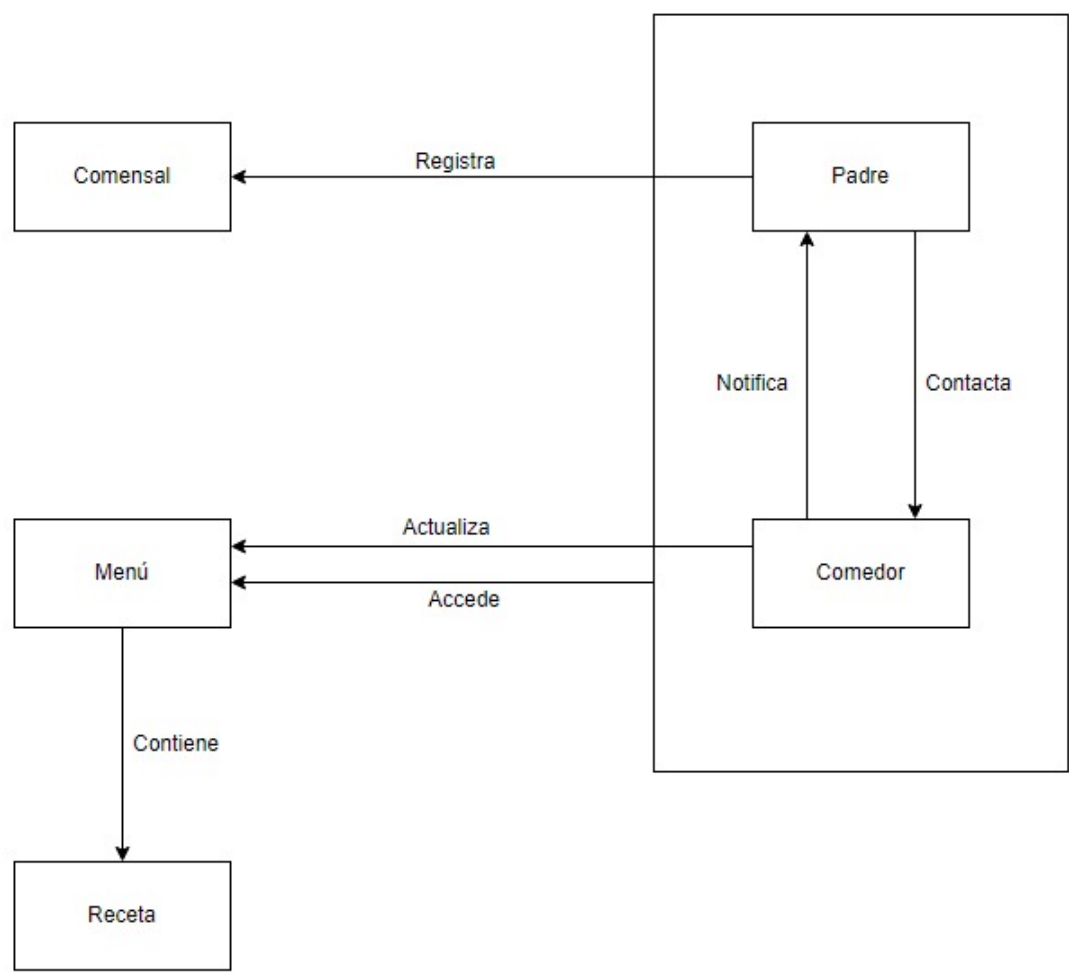




UXPRESSIA

This persona was built in uxpressia.com

Modelo conceptual del problema



# ESPECIFICACION

## Definición de requerimientos funcionales y no funcionales

### Actores

- Padre/Madre
- Encargado del comedor

### Requerimientos Funcionales

- RF1: Registro de comensales

**Actores:** Padre

**Descripción:** El sistema debe permitir a los padres registrar los comensales (hijos) que quiere gestionar con la aplicación.

**Prioridad:** Alta

- **RF2: Administrar comensales**

**Actores:** Padre

**Descripción:** El sistema debe permitir a los padres además de registrar los comensales, eliminar los que ya no quiera gestionar con la aplicación.

**Prioridad:** Alta

- **RF3: Ver el Menú Semanal**

**Actores:** Padre, Encargado del Comedor

**Descripción:** El sistema debe permitir a los padres y encargado del comedor ver los platos del menú para toda la semana con el detalle de que incluye cada uno.

**Prioridad:** Alta

- **RF4: Ingresar una orden por comensal**

**Actores:** Padre

**Descripción:** El sistema debe permitir a los padres cargar una orden en la cual pueden elegir para cada comensal, que días de la semana quieren pedir el menú.

**Prioridad:** Alta

- **RF5: Confirmación de pedido**

**Actores:** Padre

**Descripción:** El sistema debe permitir al padre ver en una pantalla el la orden que ingreso y confirmar la misma de ser correcta. De no ser correcta puede volver para atrás e ingresar la orden nuevamente.

**Prioridad:** Alta

- **RF6: Historial de pedido**

**Actores:** Padre

**Descripción:** El sistema debe permitir al padre ver para un comensal y mes indicado, todos los pedidos realizados.

**Prioridad:** Alta

- **RF7: Formulario de contacto**

**Actor:** Padre

**Descripción:** El sistema debe permitir que los usuarios tipo Padre se comuniquen con el responsable del menu escolar por parte del colegio mediante un formulario donde deben llenar casillas marcando nombre y apellido del niño, correo electrónico y el mensaje.

**Prioridad:** Alta

## Requerimientos no funcionales

- **RNF1: Aplicación responsive**

**Descripción:** El sistema debe ser accesible y adaptable a dispositivos móviles, tablets y computadoras.

**Prioridad:** Alta

- **RNF2: Persistencia**

**Descripción:** El sistema debe poder persistir la información de usuario, comensales y pedidos ingresados.

**Prioridad:** Alta

- **RNF3: Tipografía**

**Descripción:** El sistema debe utilizar la tipografía Gothic A1.

**Prioridad:** Alta

- **RNF4: Paleta de colores**

**Descripción:** El sistema debe respetar, en líneas generales, la paleta de colores seleccionada: #E13A3A, #FFFFFF, #000000, #868686 y #029408.

**Prioridad:** Media

- **RNF5: Google Style Guide**

**Descripción:** El sistema debe cumplir con las buenas practicas de etilos indicada por Google Style Guide para sus diferentes tecnologías.

**Prioridad:** Media

## User stories

### User Stories 1

**Título:** Consulta

**Descripción:** Como padre/madre/tutor quiero poder consultar sobre mis dudas con el menú o con alguna receta en particular para poder tomar una decisión con seguridad.

**Criterios de aceptación:** El padre/madre/tutor deberá llenar un formulario con los campos:

- Nombre
- Apellido
- Correo electrónico
- Mensaje

**User stories 2****Título:** Agregar comensales

**Descripción:** Como padre/madre/tutor quiero poder registrar a mis hijos para poder gestionarlos en la aplicación y agregarles órdenes.

**Criterios de aceptación:** El padre/madre/tutor deberá llenar un formulario con estos campos:

- Nombre
- Apellido
- Edad
- Colegio
- Dieta especial

**User stories 3****Título:** Ingresar orden

**Descripción:** Como padre/madre/tutor quiero poder ingresar órdenes a mis hijos para ya confirmar lo que van a comer en los días que no lleven comida de casa.

**Criterios de aceptación:** El padre/madre/tutor debe tener algún comensal agregado y elegir los días que quiere pedir el menú.

User stories 4

**Título:** Ver menú

**Descripción:** Como padre/madre/tutor quiero poder entrar a la aplicación a ver el menú para decidir qué días voy a pedirlo.

**Criterios de aceptación:** El padre/madre/tutor tiene que entrar a la pestaña ver menú.

Casos de Uso Detallados

Caso de uso 1

**Título:** Consultar

**Actor:** Padre/Madre/Tutor

**Descripción:** Probar el RF7

**Precondición:** Ninguna

**Postcondición:** La consulta es enviada a los responsables del menú

**Curso Normal:**

Interacción de Padre/Madre/Tutor	Respuesta del Sistema
1-Inicia la aplicación	2- El sistema muestra la página de inicio con las 5 opciones disponibles
3-Elije la opción "Contacto"	4- El sistema muestra la página "Contacto"
5-Ingresa nombre, apellido, correo y mensaje y pulsa el botón	6- El sistema envía la consulta y vuelve a la página de inicio

**Caso Alternativo:**

Interacción de Padre/Madre/Tutor	Respuesta del Sistema
5.1-Ingresa erróneamente los datos o falta ingresar un campo	El sistema muestra advertencia del error cometido

Caso de uso 2

**Título:** Registrar Comensales

**Actor:** Padre/Madre/Tutor

**Descripción:** Probar el RF1

**Precondición:** Ninguna

**Postcondición:** Se agrega el comensal

**Curso Normal:**

Interacción de Padre/Madre/Tutor	Respuesta del Sistema
1-Inicia la aplicación	2- El sistema muestra la página de inicio con las 5 opciones disponibles
3-Elije la opción "Administrar Comensales"	4- El sistema muestra la página "Comensales"
5-Pulsa "Registrar Comensal"	6- El sistema muestra la página "Agregar Comensales"
7-Ingresa nombre, apellido, edad, colegio y dieta especial y pulsa "Agregar comensal"	8- El sistema agrega al comensal y vuelve a la página "Comensales"

**Caso Alternativo:**

Interacción de Padre/Madre/Tutor	Respuesta del Sistema
7.1-Ingresa erróneamente los datos o falta ingresar un campo	El sistema muestra advertencia del error cometido

Caso de uso 3

**Título:** Ingresar orden y confirmarla

**Actor:** Padre/Madre/Tutor

**Descripción:** Probar el RF4

**Precondición:** Tener algún comensal agregado

**Postcondición:** Se envía la orden al encargado del menú

**Curso Normal:**

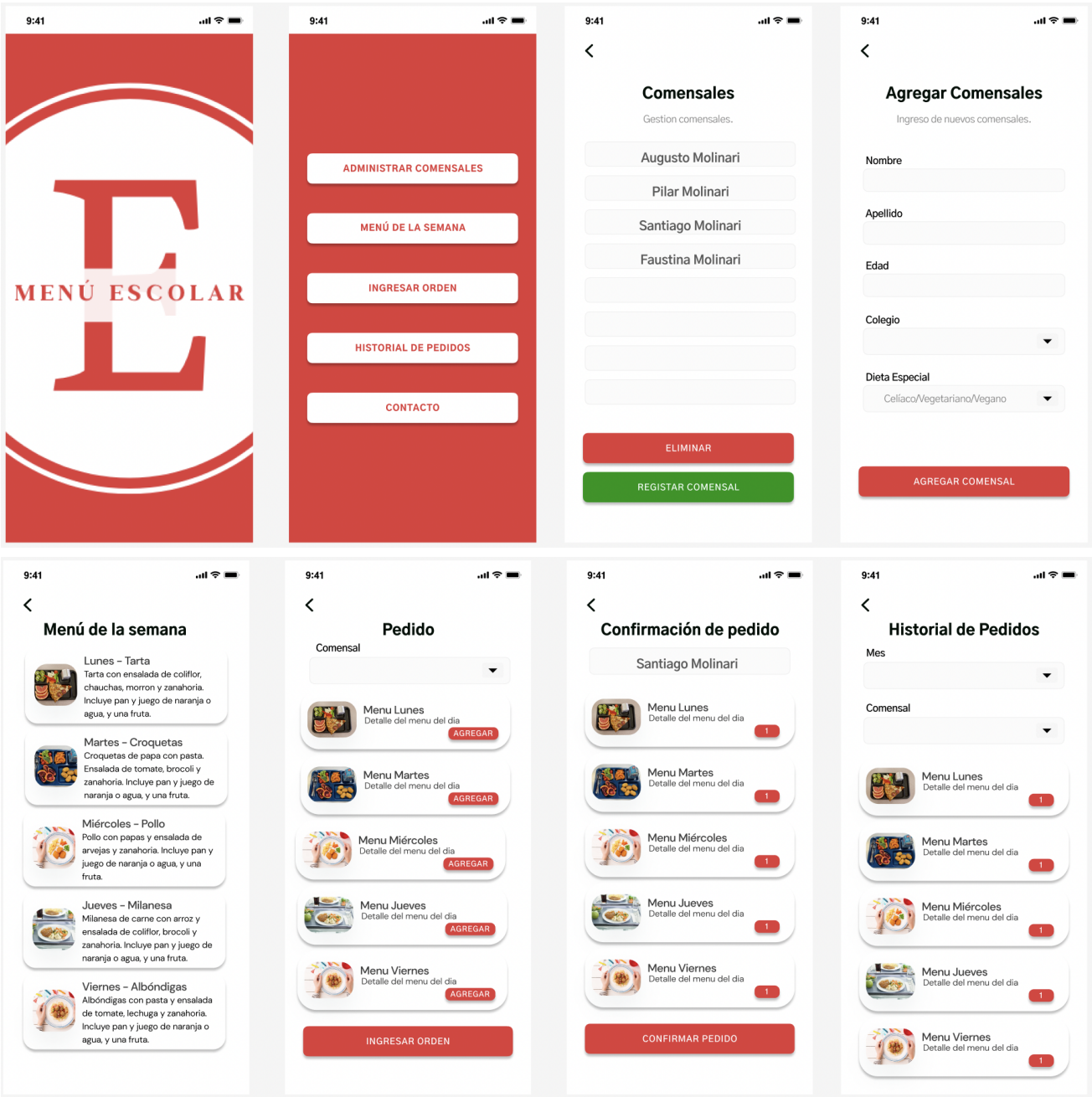


Interacción de Padre/Madre/Tutor	Respuesta del Sistema
1-Inicia la aplicación	2- El sistema muestra la página de inicio con las 5 opciones disponibles
3-Elije la opción "Ingresar Orden"	4- El sistema muestra la página "Pedido"
5-Selecciona el comensal y los días que quiere el menú y presiona "Ingresar Orden"	6- El sistema muestra la página "Confirmación de pedido"
7-Presiona "Confirmar pedido"	8- El sistema envia la orden

Curso Alternativo:

Interacción de Padre/Madre/Tutor	Respuesta del Sistema
5.1-No elige ningún día o ningún comensal	El sistema muestra advertencia del error cometido

Bocetos de IU



9:41

Contacto

Nombre

Apellido

Edad

Mensaje

ENVIAR MAIL

## VALIDACION Y VERIFICACION

### Verificion de los requerimientos

Lista de comprobación:

Requerimiento	RF1	RF2	RF3	RF4	RF5	RF6	RF7	RNF1	RNF2
Consistente	Si	Si	Si	Si	Si	Si	Si	Si	Si
Preciso	Si	Si	No	Si	Si	Si	Si	Si	Si
Verificable	Si	Si	Si	Si	Si	Si	Si	Si	Si
Priorizado	Si	Si	Si	Si	Si	Si	Si	Si	Si
Viable	Si	Si	Si	Si	Si	Si	Si	Si	Si

En el RF3 no explica que diferencia hay si entra el padre o si entra el ecnargado a ver el menú

Validar la solución con personas no involucradas en el proyecto

Estuvimos en una instancia con una madre y le mostramos todos los requerimientos, casos de uso e interfaz. La madre comentó que en base a la interfaz no le quedaba claro si podía elegir un solo día en específico o era necesario elegir toda la semana, al mostrarle los casos de uso le quedo más claro. Otra cosa que no vio fue el precio, y si el mismo varia por cantidad de pedidos.

REFLEXION

Detalle del trabajo individual

- Laura

Definimos en conjunto los temas que iba a llevar adelante cada uno, y se pautaron reuniones para ver avances y comentar sobre posibles cambios. En mi caso, me encargue del detalle de los repositorios Git, del versionado, los requerimientos funcionales y no funcionales, y del diseño de la interfaz de usuario. En cuanto a la interfaz, me pareció interesante el uso de Figma, ya que es una herramienta que no conocía y me resulto muy útil para poder plasmar las ideas que teniamos en mente. En cuanto a los repositorios, me pareció interesante el uso de Git, ya que es una herramienta que no conocía y me resulto muy útil para el trabajo en equipo y por sobre todo para el control de cambios de versiones. Los requerimientos funcionales fueron clave para poder definir el alcance del proyecto, y los no funcionales para poder definir las características que debía tener el sistema.

- **Mario**

Yo me encargué de la parte de elicitación (aunque estuvimos en contacto constante con el resto de los compañeros del obligatorio ya que las partes no eran independientes), en este proceso pude aprender bastante del funcionamiento de los comedores escolares, en los cuales me llevé una grata sorpresa de su buen funcionamiento a pesar de las limitaciones logisticas y de presupuesto. Ese buen funcionamiento nos inspiró a más que reinventar la rueda profundizar en los aspectos a mejorar, en nuestro caso elegimos la comunicación entre la cocina y el hogar ya que creemos que una app puede ser de gran ayuda en esto al centralizar tanto la decisión de recibir o no el menu diario como la comunicación con la escuela en una sola via.

- **Gastón**

Yo estuve a cargo de los User Stories, los Cases detallados, la verificación y la validación. Encargarme de las stories y los casos detallados me hicieron ver que separar el problema en partes ayuda a tener más organizado el proyecto ademas de facilitar la identificación de los problemas principales que hay que probar. Haciendo la verificación me di cuenta de que no siempre se plasma en escrito la idea al 100%, podes omitir un detalle por distracción o porque vos lo entendes bien. Por eso es importante que alguien que no hizo los requerimientos los revise. En la validación la verdad disfrute mostrar la parte del proyecto que habiamos logrado construir y como una persona ajena lo interpretaba.

## **Técnicas aplicadas y aprendizajes**

Aprendimos a utilizar Figma, Git y Github, herramientas que no conociamos y nos resultaron muy útiles para el desarrollo del proyecto. Dentro de la elaboración del readme aprendimos sobre la utilización de Markdown. También utilizamos herramientas como Uxpressia y Draw.io para la elaboración de los user personas y el diagrama de modelo conceptual.

# **FUNDAMENTOS DE INGENIERIA DE SOFTWARE - OBLIGATORIO 2**

---

## **ESTILOS PREDEFINIDOS POR EL EQUIPO**

---

### **JavaScript**

- Los nombres de los archivos deben estar todos en minúscula y pueden contener guiones bajos.
- Los comentarios deben estar en una unica linea y deben estar alineados con el codigo.

- Se debe utilizar camelCase para los nombres de las funciones y las variables.
- Se debe utilizar PascalCase para los nombres de los constructores y las clases.
- Se debe utilizar mayúsculas para las constantes.
- El máximo de caracteres por línea es de 80.
- La indentación debe ser de 2 espacios.
- Se debe dejar una línea en blanco entre los bloques de código.

## HTML

- Los nombres de los elementos de atributos deben estar en minúscula.
- No se debe utilizar JavaScript dentro del HTML, se debe importar.
- El máximo de caracteres por línea es de 80.
- La indentación debe ser de 2 espacios.
- Se debe dejar una línea en blanco entre los bloques de código.

## CASOS DE USO SELECCIONADOS

---

- Caso de uso 2: Registrar Comensales
- Revisar historial pedido

## CONSTRUCCION

---

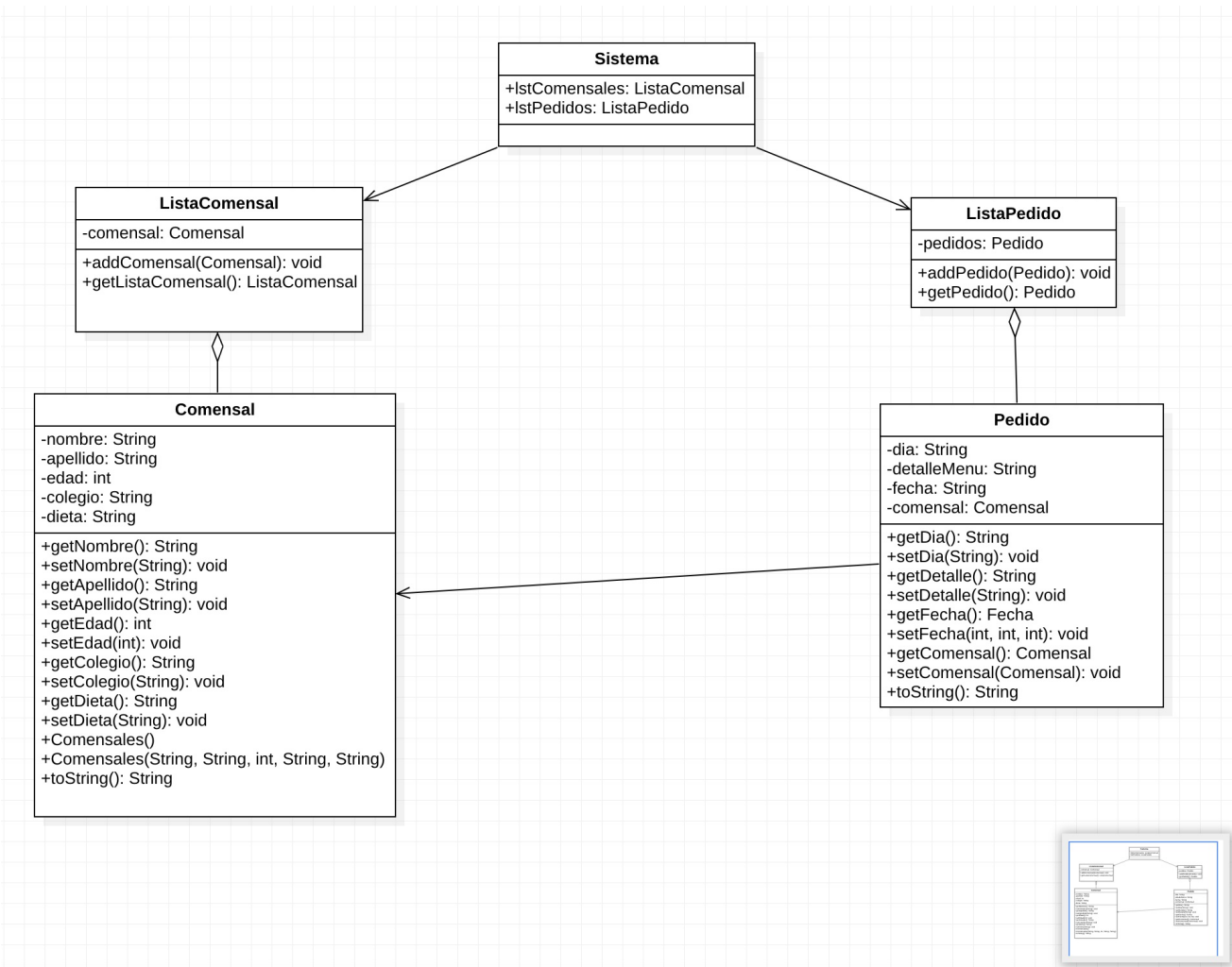
- **Dominio**

Se generan las clases: comensal.js, lista-comensal.js, pedido.js y lista-pedido.js.

- **Interfaz**

Se generan los archivos: index.html, comensales.htm e historial.html. No se utiliza un archivo .css ya que los estilos se aplican directamente en el html por medio del uso de la librería Bootstrap. En main.js se encuentra el código JavaScript que se encarga de la interacción con el usuario.

# UML



## INTERFAZ DE USUARIO

Se respetan las heurísticas de usabilidad de Nielsen. Son una serie de reglas o pautas escritas por Jakob Nielsen en la década del noventa, que se enfocan en evaluar el diseño de la interfaz de usuario. Por ejemplo con el uso de breadcrumbs para favorecer el reconocimiento ante la memorización. Se intento mantener una estética simple y minimalista, para no distraer al usuario de lo que es realmente importante.

## CODIFICACION

Para la codificacion se utilizo el IDE Visual Studio Code, ayudandonos de las extensiones de Prettier y ESLint para mantener un codigo limpio y ordenado, y Live Server para poder visualizar los cambios en tiempo real.

También se utilizo la extensión WAVE Evaluation Tool para verificar la accesibilidad de la pagina.



# TESTS UNITARIOS

Para realizar las pruebas unitarias utilizamos el marco de prueba Jest, para ello descargamos el paquete de Babel para permitir el soporte de ECMAScript, para ello utilizamos el comando "npm install --save-dev @babel/register @babel/preset-env" y creamos un archivo de configuración .babelrc

Los tests unitarios se encuentran en la carpeta "tests" y se ejecutan con el comando "npm run test" y consisten de 2 tipos de pruebas, las pruebas de validez donde se determina si los datos ingresados son válidos y las pruebas donde se determina si los datos ingresados son consistentes con los datos de la base de datos. Por ejemplo si lo genero un comensal con el nombre Mario, el test unitario correspondiente verifica que en el campo "nombre" del comensal se encuentre el nombre Mario. Luego hay una prueba de la funcion toString que verifica que se imprima correctamente el comensal con el formato nombre + apellido.

Al correr el comando npm run test se ejecutan los tests unitarios y se obtiene el siguiente resultado:

```
Administrador@DESKTOP-I1PC2AT MINGW64 /e/Escritorio/fis/Obligatorio1-Collazo-Ricca-Salgado (develop)
$ npm run test

> proyecto-lcmcr@1.0.0 test
> jest --config jest.config.cjs

PASS src/dominio/test/ListaComensal.test.js
PASS src/dominio/test/comensal.test.js
PASS src/dominio/test/pedido.test.js

Test Suites: 3 passed, 3 total
Tests:       22 passed, 22 total
Snapshots:   0 total
Time:        1.562 s, estimated 2 s
Ran all test suites.
```