



BENEMÉRITA UNIVERSIDAD AUTÓNOMA
DE PUEBLA

FACULTAD DE CIENCIAS DE LA
COMPUTACIÓN

INGENIERÍA EN TECNOLOGÍAS DE LA
INFORMACIÓN

MATERIA: PROGRAMACIÓN DE
DISPOSITIVOS MÓVILES

PRÁCTICA 2 COMPONENTES DE IONIC Y
SU USO EN UNA APLICACIÓN

M.C. LUIS Yael MENDEZ – SANCHEZ

ALUMNOS: JORGE LEON VIVANCO

MATRICULA: 202129258

INTERPERIODO 2025

INTRODUCCIÓN

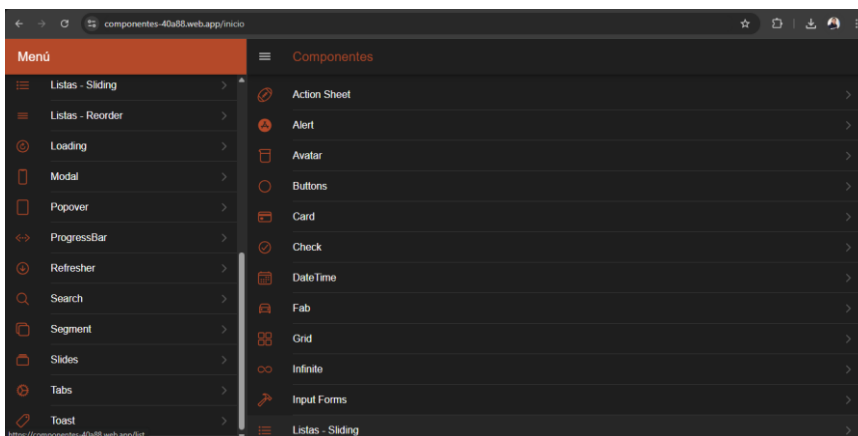
El desarrollo de aplicaciones móviles híbridas se ha convertido en una habilidad esencial en el entorno tecnológico actual, y Ionic destaca como uno de los frameworks más potentes para crear interfaces modernas y responsivas utilizando tecnologías web. Esta práctica tiene como propósito fundamental explorar y aplicar de forma detallada los componentes principales, comprendiendo no solo su configuración básica, sino también su personalización y adaptación a contextos reales.

Durante esta actividad, se abordarán de manera estructurada los componentes disponibles en el framework, organizados alfabéticamente como un glosario práctico. Se implementará cada uno en una aplicación funcional, con ejemplos y mejoras visuales, fomentando la comprensión de su utilidad y versatilidad en el desarrollo de interfaces atractivas. Asimismo, se reforzarán conceptos clave como navegación, estructura modular y adaptación a diferentes dispositivos, sentando así las bases para futuras aplicaciones robustas con Angular e Ionic.

DESARROLLO

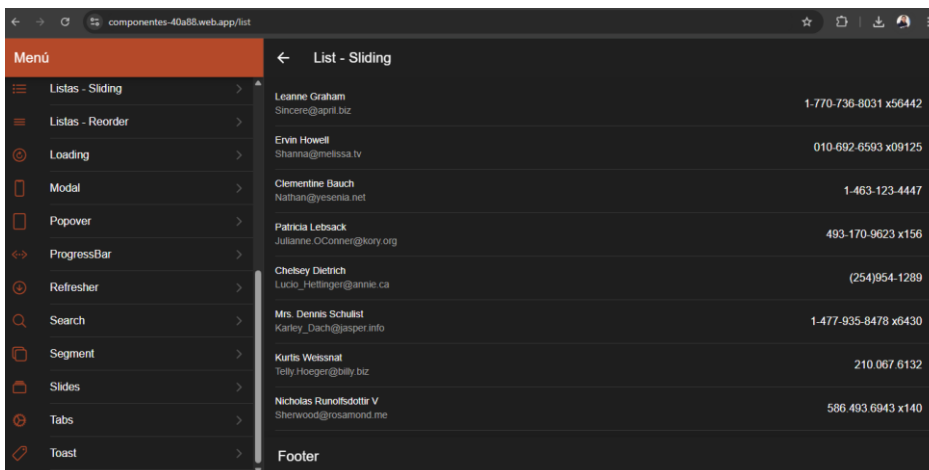
1. Navegación y estructura básica

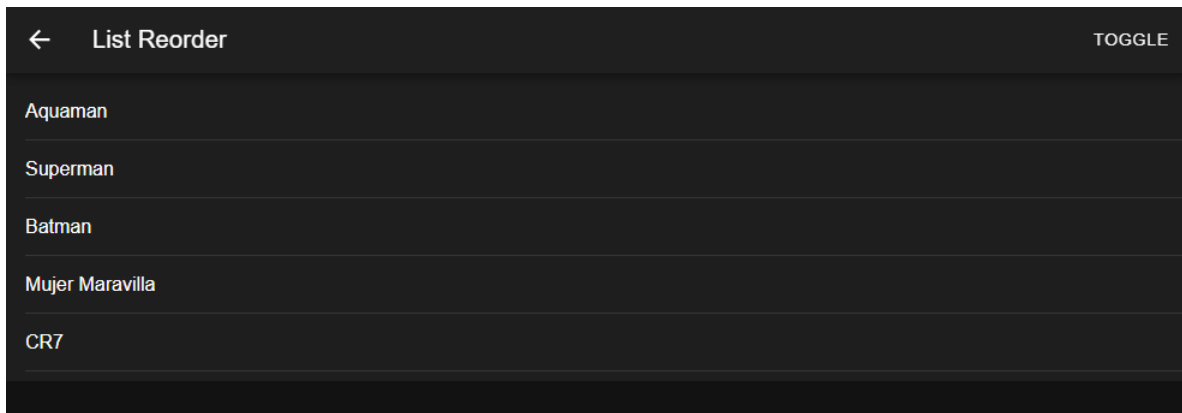
En esta sección se realizó la configuración inicial del proyecto con Ionic, estableciendo los componentes base de la aplicación. Se modificó la pantalla principal (home) y se exploró el sistema de navegación entre páginas utilizando el enrutador de Angular. También se implementó el componente ion-back-button, el cual permite regresar a la página anterior, mejorando la experiencia del usuario en dispositivos móviles. Además, se creó un módulo de componentes compartidos, destacando el Header, que permite un encabezado común y reutilizable en todas las vistas.



2. Listas y manejo de datos

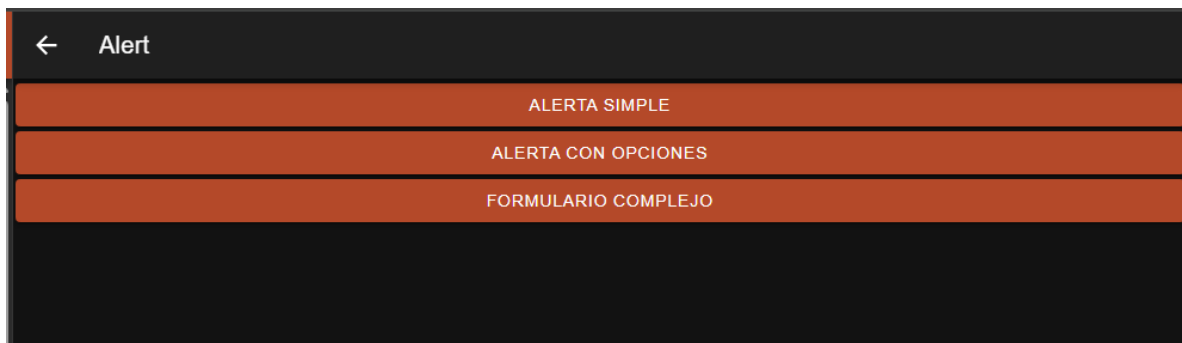
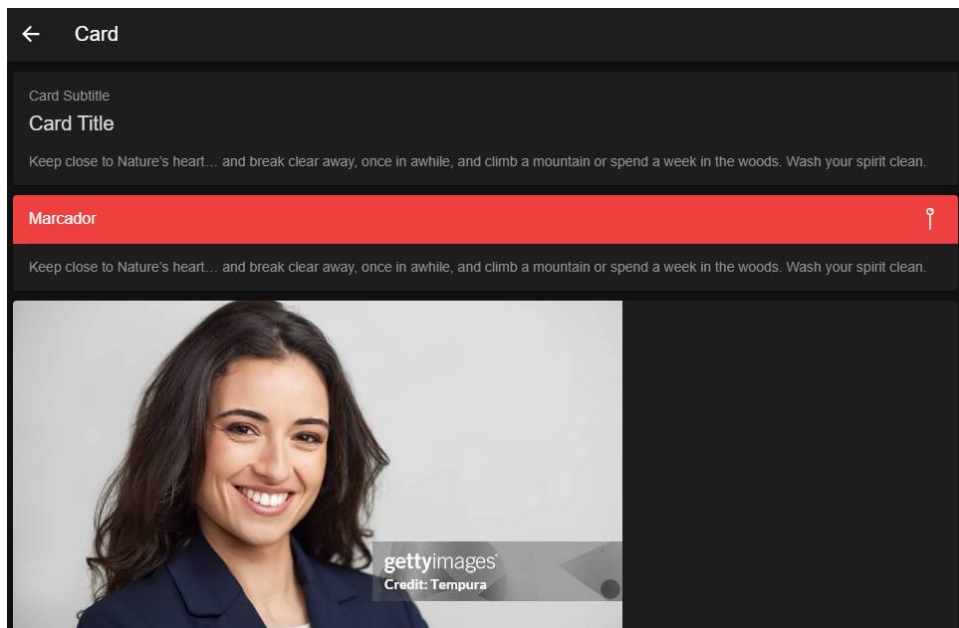
Se trabajó intensamente con el componente ion-list, base para mostrar colecciones de elementos. Se añadieron funcionalidades como deslizamiento de ítems (ion-item-sliding), reordenamiento dinámico (ion-list-reorder), carga infinita (ion-infinite-scroll) y refresco manual (ion-refresher). También se implementó una ion-searchbar para permitir la búsqueda filtrada en tiempo real sobre listas.



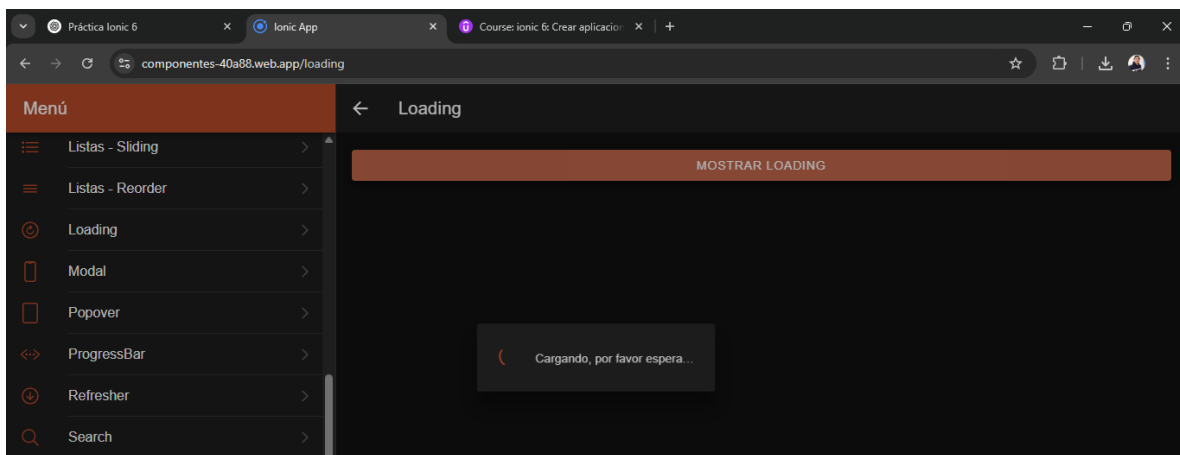
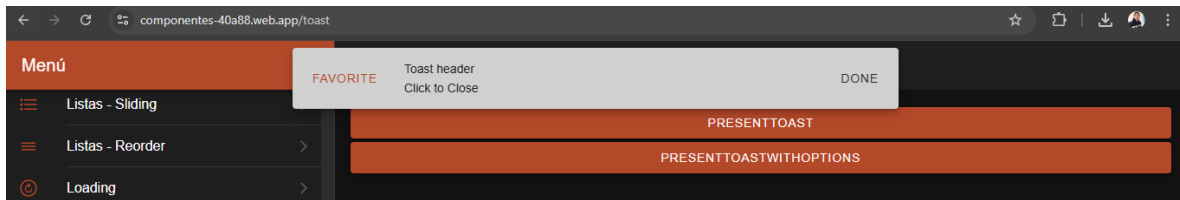


3. Componentes interactivos

Esta sección se centró en los principales componentes visuales e interactivos que Ionic ofrece. Se integraron botones (ion-button), tarjetas (ion-card) y cajas de verificación (ion-checkbox). Se profundizó en el uso de ion-datetime para seleccionar fechas, incluyendo su personalización.

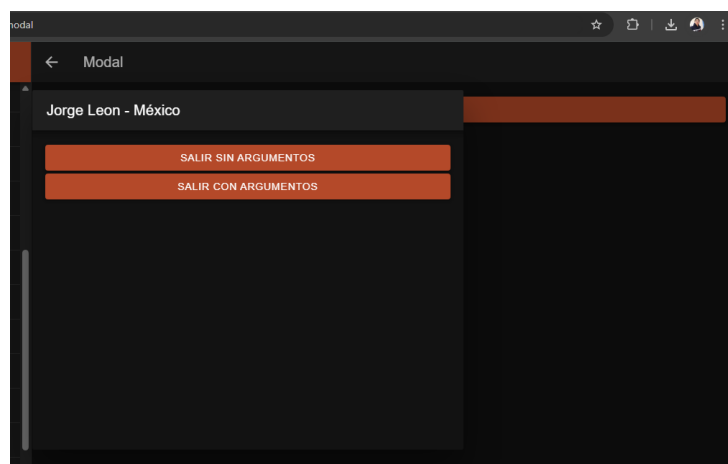


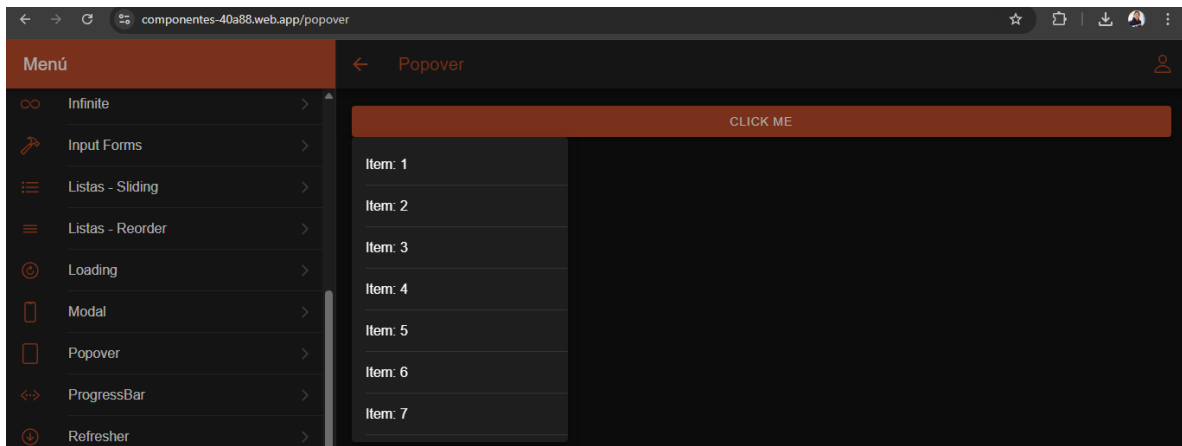
También se exploraron componentes de interacción como ion-alert, ion-action-sheet, ion-loading y ion-toast, útiles para mensajes al usuario. Se agregaron elementos como ion-progress-bar, ion-range y ion-segment para representar progreso y filtrar vistas. Por último, se personalizó la experiencia visual con ion-skeleton-text, ion-avatar y carruseles con ion-slides.



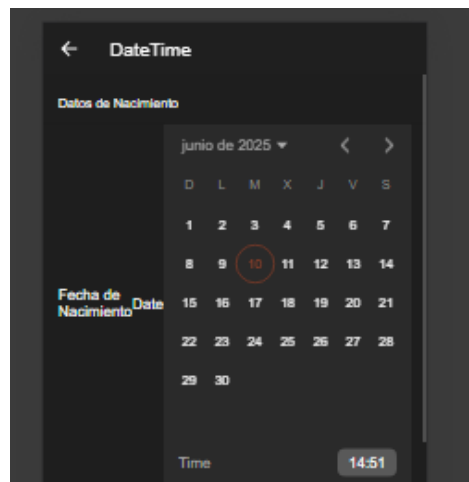
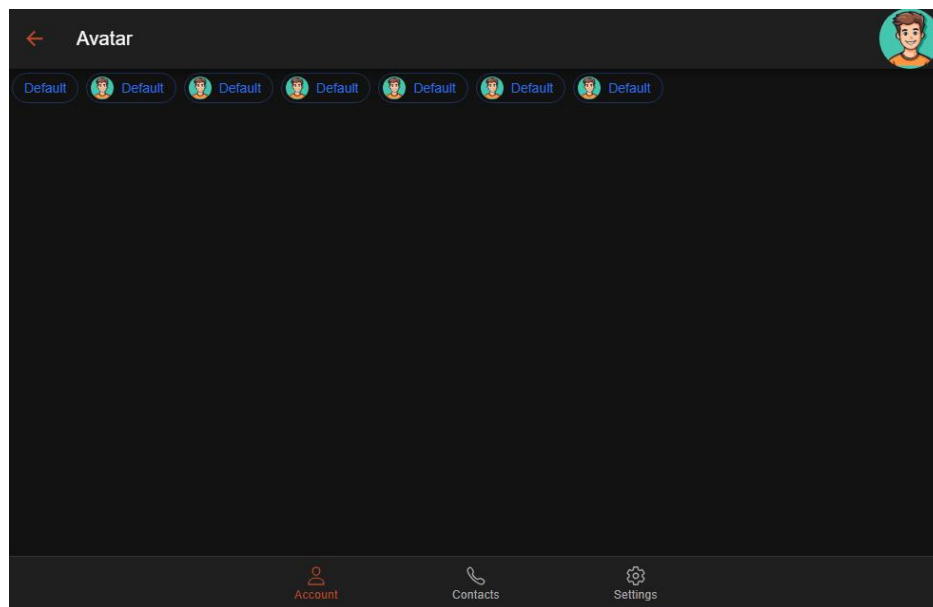
4. Modales y menús

Se integraron modales (ion-modal) que permiten mostrar contenido adicional sin cambiar de página, así como la comunicación entre componentes padre e hijo. También se exploraron ion-popover, ideales para mostrar opciones rápidas contextualizadas, y ion-menu, usado para implementar un menú lateral interactivo.



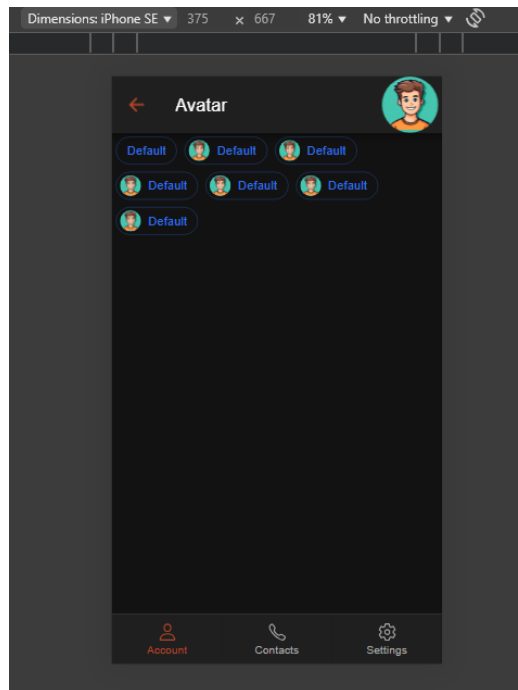


Por último, se añadieron ion-tabs para navegación entre secciones inferiores, y ion-split-pane que permite tener un diseño tipo escritorio con paneles divididos, útil para tablets o pantallas grandes.



5. Maquetación y estilos

En esta última parte se utilizó el sistema de grillas ion-grid, ion-row, ion-col para diseñar estructuras responsivas. Se practicó el control de tamaño por pantalla y se trabajó con variables SCSS para personalizar colores, permitiendo una identidad visual propia. También se incluyó un bonus visual para aplicar transparencia y efectos especiales en dispositivos iOS, ajustando estilos específicos con clases de plataforma.



CONCLUSION

A lo largo de esta práctica, he tenido la oportunidad de adentrarme en el ecosistema de Ionic y experimentar de forma directa el uso e integración de sus principales componentes. Esta experiencia no solo me permitió conocer la configuración básica de cada uno, sino también descubrir cómo personalizarlos y adaptarlos a las necesidades reales de una aplicación móvil. Me di cuenta de que detrás de cada componente existe una lógica bien pensada que busca facilitar la creación de interfaces modernas, responsivas y funcionales, todo con una base sólida en tecnologías web.

Durante el desarrollo de la aplicación, fui aplicando paso a paso lo aprendido en los videos del curso, y eso me permitió entender mejor el propósito de herramientas como el ion-header, los botones, listas, inputs, cards, modales y muchos otros elementos que, aunque simples en apariencia, tienen múltiples propiedades que permiten una personalización completa. Me llamó especialmente la atención el poder de los componentes como el ion-popover o el ion-skeleton-text, ya que permiten enriquecer la experiencia del usuario de manera significativa.

Además, uno de los mayores aprendizajes fue comprender cómo se estructura una aplicación en Ionic utilizando Angular, cómo se gestiona la navegación entre pantallas, cómo puedo agrupar mis propios componentes personalizados en módulos reutilizables y cómo optimizar el diseño para que se vea bien tanto en una computadora como en un teléfono móvil. El uso de Capacitor fue clave para ver los resultados directamente en mi celular, lo que hizo que la práctica se sintiera mucho más real y conectada con lo que podría ser un proyecto en producción.

REFERENCIAS BIBLIOGRÁFICAS

- Ionic Framework. (s.f.). *Ionic Documentation*. Recuperado el 10 de junio de 2025, de <https://ionicframework.com/docs>
- Angular. (s.f.). *Angular Documentation*. Recuperado el 10 de junio de 2025, de <https://angular.io/docs>
- CapacitorJS. (s.f.). *Capacitor Documentation*. Recuperado el 10 de junio de 2025, de <https://capacitorjs.com/docs>
- Ionic Team. (2022). *Ionic 6.0 Release Notes*. Recuperado de <https://ionicframework.com/blog/announcing-ionic-v6>
- Wieruch, R. (2019). *The Road to React: Your journey to master plain yet pragmatic React.js*. Independently published. (*Referente a buenas prácticas en componentes reutilizables, aplicables también en Angular/Ionic*)
- Freeman, A., & Sanderson, P. (2020). *Pro Angular 9*. Apress. <https://doi.org/10.1007/978-1-4842-6037-2>
- Hartl, M. (2020). *Web Development with JavaScript and the Ionic Framework*. Learn Enough. Recuperado de <https://www.learnenough.com/>