



**INSTITUTO TECNOLÓGICO DE BUENOS AIRES - ITBA
ESCUELA DE INGENIERÍA Y GESTIÓN**

Ponte iOS

**TRABAJO FINAL PRESENTADO PARA LA OBTENCIÓN DEL
TÍTULO DE INGENIERO EN INFORMÁTICA, Diciembre 2020.**

De Rienzo, Constanza - 56659

Santoflaminio, Alejandro - 57042

Tutor:

Guerrero, Marcela Alejandra

Fecha: 17 / 12 / 2020

Índice general

Índice de cuadros	IV
Índice de figuras	V
1. Introducción	1
2. Consideraciones y contexto	3
3. Estado del arte	4
4. Etapas de proyecto	7
4.1. Definición de objetivos y requisitos	7
4.2. Cronograma y versionado	8
5. Implementación	10
5.1. Etapa 1: Modelado de la aplicación	10
5.2. Etapa 2: Funcionalidad Ponte Web	23
5.3. Etapa 3: Funcionalidad Adicional	27
5.4. Diseño final	35

6. Casos de prueba	61
6.1. Creación de cuenta como donante	61
6.2. Inicio de sesión	62
6.3. Visualizar los pedidos en Ponte	63
6.4. Visualizar las instituciones registradas con su detalle	64
6.5. Realizar una donación a un pedido	65
6.6. Realizar una donación sin pedido específico	67
6.7. Visualizar las donaciones como donante	69
6.8. Cancelación de donación	70
6.9. Geolocalizar un voluntario	71
6.10. Compartir un pedido	72
6.11. Convertirse en voluntario	72
6.12. Visualizar listado de entregas	74
6.13. Editar contraseña	74
6.14. Editar datos	75
7. Conclusiones	77
8. Trabajo futuro	79
9. Bibliografía	82

Índice de cuadros

6.1. Creación de cuenta como donante	62
6.2. Inicio de sesión	63
6.3. Visualizar los pedidos en Ponte	64
6.4. Visualizar las instituciones registradas con su detalle	65
6.5. Realizar una donación a un pedido	66
6.6. Realizar una donación sin pedido específico	68
6.7. Visualizar las donaciones como donante	69
6.8. Cancelación de donación	70
6.9. Geolocalizar un voluntario	71
6.10. Compartir un pedido	72
6.11. Convertirse en voluntario	73
6.12. Visualizar listado de entregas	74
6.13. Editar contraseña	75
6.14. Editar datos	76

Índice de figuras

3.1. Diagrama indicando la estructura actual del proyecto. En este caso ejemplificado con Ponte iOS, aunque el funcionamiento es el mismo para todas las plataformas.	5
3.2. Diagrama del flujo principal de Ponte con la API actual.	6
4.1. Captura de pantalla del <i>Board de Jira</i> utilizado.	9
5.1. Dos formas de sostener el teléfono móvil con mapa de calor según uso.	12
5.2. Inicio de sesión en versión web.	14
5.3. Perfil de usuario en versión web.	15
5.4. Listado de instituciones en versión web.	15
5.5. Pedidos en versión web.	16
5.6. Perfil en versión Android	16
5.7. Home en versión Android	16
5.8. Instituciones y pedidos en versión Android	17
5.9. Boceto de inicio de sesión.	18
5.10. Boceto de creación de cuenta.	18

5.11. Boceto de pedidos.	19
5.12. Boceto de compartir pedido.	19
5.13. Boceto de realizar donación.	20
5.14. Boceto de nueva donación.	20
5.15. Boceto de instituciones.	21
5.16. Boceto de detalle de institución.	21
5.17. Boceto de perfil.	22
5.18. Boceto de configuración.	22
5.19. Boceto del historial.	23
5.20. Gráfico indicando el funcionamiento de la geolocalización mediante PubNub.	29
5.21. Aguardando ubicación del voluntario.	30
5.22. Voluntario dirigiéndose a domicilio del donante.	30
5.23. Voluntario dirigiéndose a una institución.	31
5.24. Pedido con el ícono de compartir pedido.	33
5.25. Menú para compartir un pedido de donación.	33
5.26. <i>Bottom tab bar</i>	35
5.27. <i>Splash screen</i>	36
5.28. Pantalla de bienvenida.	38
5.29. Creación de usuario.	38

5.30. Error al usar mail ya registrado.	39
5.31. Error al no completar datos.	39
5.32. <i>Home</i> de un donante.	40
5.33. <i>Home</i> de un voluntario.	40
5.34. <i>Pedidos</i>	42
5.35. Realizar una donación.	42
5.36. Error al no completar datos.	43
5.37. Error al elegir horarios repetidos.	43
5.38. Donación confirmada.	44
5.39. <i>Donar</i>	45
5.40. Error al no completar campos.	45
5.41. <i>Instituciones</i>	46
5.42. Detalle de institución.	46
5.43. Perfil de un donante.	47
5.44. Formulario para ser voluntario.	47
5.45. Alerta al solicitar ser voluntario.	48
5.46. Donante con solicitud para ser voluntario.	48
5.47. Perfil de un voluntario.	49
5.48. Menú de configuración.	49

5.49. Historial de un donante.	50
5.50. Historial de un donante sin donaciones previas.	50
5.51. Historial de un voluntario.	51
5.52. Editar datos.	52
5.53. Editar contraseña.	52
5.54. Al interactuar con una donación desactualizada se muestra una advertencia y se recarga el contenido.	53
5.55. Estado CREATED	54
5.56. Estado CREATED con voluntario asignado	54
5.57. Estado BEING-LOOKED-FOR	55
5.58. Estado DELIVERED-BY-DONOR	56
5.59. Estado PICKED-BY-VOLUNTEER	56
5.60. Estado BEING-TAKEN-TO	57
5.61. Estado DELIVERED	57
5.62. Estado CANCELLED	58
5.63. Pantalla indicando error de conexión. El usuario puede hacer un <i>pull to refresh</i> para recargar el contenido.	59
5.64. El indicador de carga consiste en un círculo que se agranda y achica mientras cambia su opacidad.	60

Introducción

Ponte (del latín de puente) es una plataforma que permite conectar instituciones que necesitan donaciones y donantes que las pueden proveer.

Creado en el 2018 por los alumnos del ITBA, Juan Sebastián Vera y Noelia Belén López, Ponte busca funcionar como nexo entre una parte de la sociedad que a menudo queda invisible y otra parte de la sociedad dispuesta a ayudar pero que no siempre cuenta con todos los instrumentos para hacerlo.

Este proyecto nace como una aplicación web en la cual aquellos interesados pueden registrarse como donantes, ya sea de tiempo para actuar de transportistas (voluntario), o de bienes para suplir pedidos (donante) mientras que las asociaciones civiles u ONGs pueden también registrarse y realizar pedidos con sus necesidades.

De esta manera Ponte actúa como una herramienta para facilitar el encuentro de estos dos grupos de gente: aquel con la necesidad y aquel con los recursos para suplirla.

Partiendo de esta idea y este objetivo de facilitar y actuar como herramienta surge la idea de Ponte iOS, que busca ser la versión móvil para la plataforma iOS de este mismo proyecto.

Esta nueva iteración de Ponte se considera sumamente útil ya que actual-

mente los teléfonos móviles son una de las tecnologías más usadas (si no la más usada), por lo que tener una versión de Ponte para iOS puede potencialmente abrir las puertas a muchos nuevos usuarios.

A lo largo del presente informe se registra el proceso de trabajo llevado a cabo para la realización de la aplicación Ponte para el sistema operativo iOS.

Consideraciones y contexto

Habiendo comenzado el proyecto en Marzo, 2020 resulta importante destacar el contexto mundial que se vivía y sigue viviendo por la pandemia de **COVID-19**.

Dicho contexto imposibilitó las reuniones presenciales, obligando a llevar a cabo todo el trabajo de manera remota. Si bien se intentó cumplir con el cronograma de la mejor manera posible también se considera que la situación mundial generó un efecto de desgaste, demandando una mayor inversión de tiempo. A pesar de esto, se logró cumplir con el proyecto dentro del plazo pautado.

Estado del arte

En la actualidad encontramos desplegado el proyecto Ponte únicamente en su versión web. Paralelamente, se estuvo trabajando en Ponte iOS y Android, finalizando dichos desarrollos en diciembre de 2020. Asimismo, existe el proyecto **Ponte 4**, que implicará mejoras sustanciales en el backend que utilizan todas las plataformas.

Con respecto al estado actual del proyecto Ponte, desde noviembre de 2020 se encuentra caído el despliegue de la API del proyecto. Este despliegue se encontraba en **Heroku**, donde se hacía uso de un módulo para interactuar con **MongoDB**. Dicho módulo quedó obsoleto el 10 de noviembre, provocando que deje de funcionar la API desplegada y que se necesite buscar rápidamente una alternativa. Es por ello que al momento de la entrega nos encontramos ejecutando nuestro trabajo con el servidor del proyecto Ponte corriendo de manera local en nuestras computadoras. En cuanto a diferencias de código en nuestro proyecto, las mismas son mínimas y se encuentran explicadas a continuación.

El servidor del proyecto Ponte hace uso de la base de datos MongoDB para el almacenamiento de usuarios, instituciones, donaciones, pedidos, ofertas y voluntarios. Para la autenticación de usuario también se hace uso de **Firestore** de Google. Es por ello que en Ponte web, para la creación de un usuario, el mismo se crea tanto en MongoDB como en Firestore. La interacción con Firestore se realiza

únicamente desde el front end.

Es decir, cuando un usuario se registra mediante algunas de las plataformas de Ponte, se envía la información a Firebase, realizando allí las validaciones correspondientes. En caso de haber un error, el flujo termina ahí. En caso contrario, se envía la respuesta a Ponte, y se prosigue a crear el usuario en la base de datos MongoDB. En caso de que todo ocurra satisfactoriamente, el usuario pasará a estar con su sesión iniciada.

Adicionalmente, se hace uso de una herramienta llamada **Forest Admin**. Esta herramienta permite a los administradores del proyecto realizar las tareas que no son manejadas por el flujo de la aplicación tales como asignar un voluntario a una donación, validación de instituciones, eliminación de datos, entre otros. Es decir, Forest se utiliza como herramienta de back office.

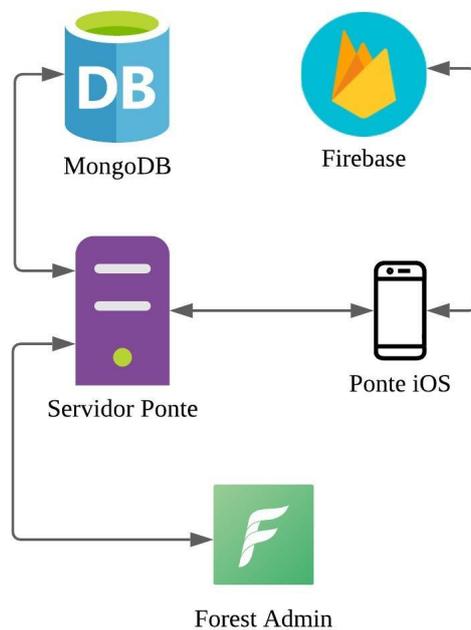


Figura 3.1: Diagrama indicando la estructura actual del proyecto. En este caso ejemplificado con Ponte iOS, aunque el funcionamiento es el mismo para todas las plataformas.

Con respecto al flujo de estados que implica el proceso de donar un bien

se encuentra detallado en el siguiente gráfico. Este flujo es el que se corresponde con el del servidor Ponte en su versión más reciente al día de hoy. Es necesario notar, que previamente a la caída del despliegue, el servidor desplegado contenía una versión anterior, la cual tenía un estado adicional *Volunteer assigned*. Dicho cambio tuvo algunas implicancias menores en el código de Ponte iOS, pero los mismos se encuentran comentados en código, para que en caso de retornar a dicha versión, el trabajo sea mínimo.

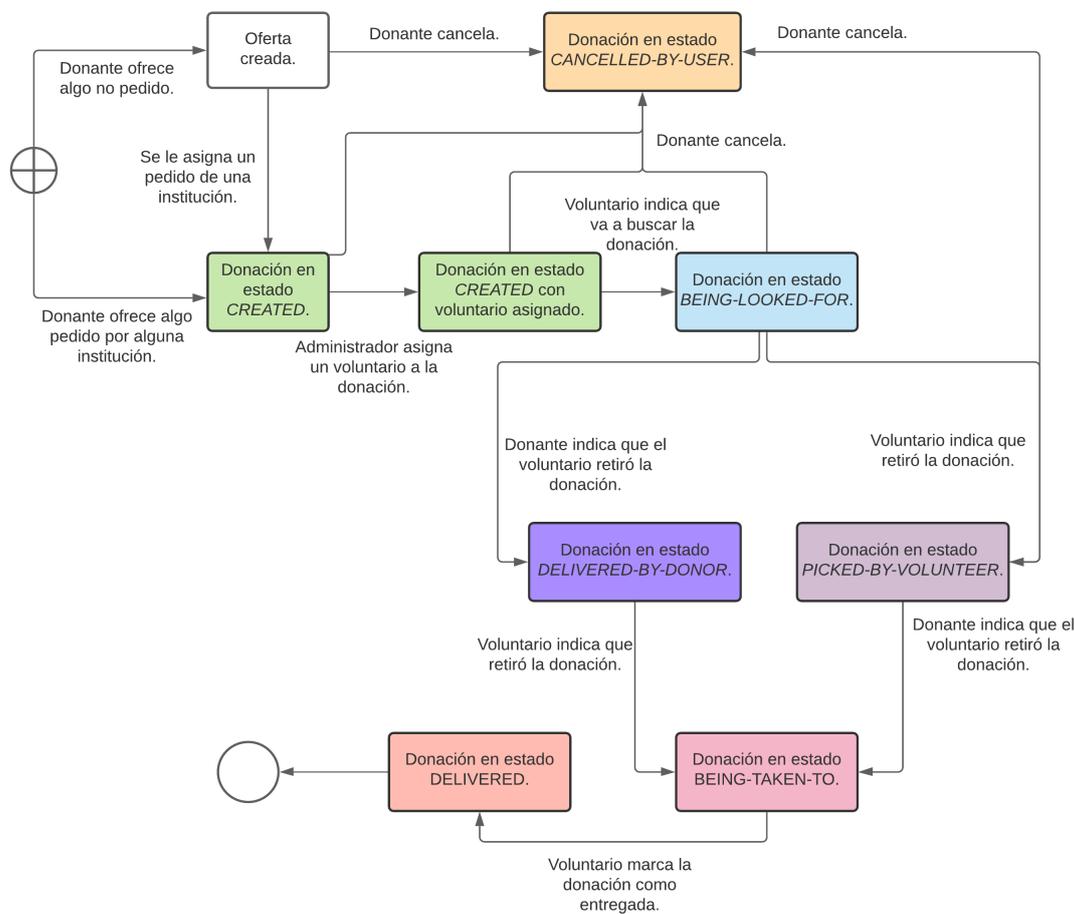


Figura 3.2: Diagrama del flujo principal de Ponte con la API actual.

Etapas de proyecto

4.1. Definición de objetivos y requisitos

Como se mencionó anteriormente, esta iteración de Ponte, parte de la versión web finalizada en el 2019. Dicha versión cuenta con las siguientes funcionalidades que se buscaron replicar:

- Registro e inicio de sesión de usuario. Opción de convertirse en voluntario.
- Perfil del donante con donaciones pendientes y su historial.
- Publicar donaciones o donar a un pedido ya hecho.
- Cancelar publicaciones realizadas.
- Buscar pedidos e instituciones.
- Actualización de los estados de una donación: Entregado, en camino, entre otras.

Para esta iteración de Ponte se tuvieron en cuenta la versión web y junto con la versión móvil en Android. Como tal, uno de los puntos importantes a tener en cuenta fue el de mantener un cierto nivel de similitud en diseño para generar una cohesión entre las tres versiones.

Adicionalmente, se buscó agregar las siguientes funcionalidades (hoy en día no presentes ni en la versión web ni en la de Android):

- Trackeo del estado de la donación en un mapa con geolocalización del transportista.
- Permitir que el donante comparta un pedido de donación a través de sus redes sociales para maximizar el alcance de ese pedido y que se cumpla más rápido el objetivo de que la institución reciba la donación. Esto podría servir por ejemplo si un donante no tiene lo que se pide pero quiere ayudar a que se consiga.

4.2. Cronograma y versionado

Para la realización de Ponte iOS, se pensaron las siguientes 3 etapas:

1. Modelado de la aplicación teniendo en cuenta los lineamientos de las aplicaciones iOS.
2. Implementación de funcionalidades presentes en Ponte web.
3. Implementación de funcionalidades adicionales.

Se buscó aplicar el método SCRUM para el desarrollo, utilizando sprints de dos semanas y la herramienta **Jira** para el seguimiento de las tareas.

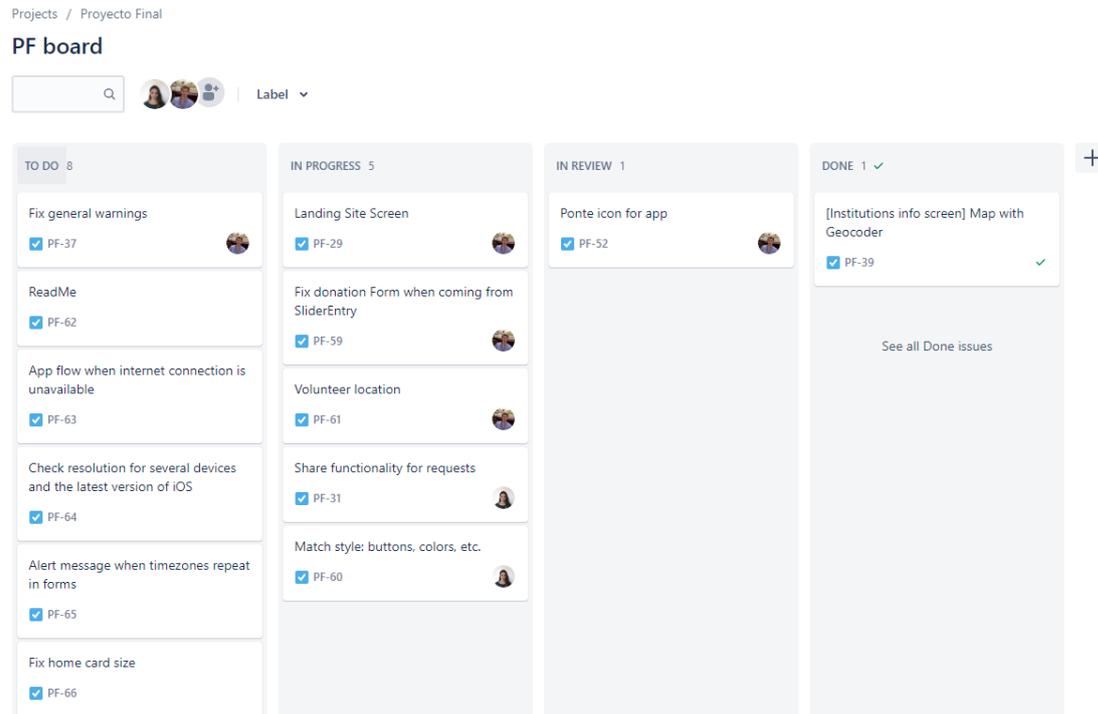


Figura 4.1: Captura de pantalla del *Board de Jira* utilizado.

Adicionalmente, se buscó aplicar buenas prácticas de desarrollo, utilizando branches para cada tarea y manteniendo una misma denominación de las mismas según el tipo de desarrollo que se agregaba (feature, chore o fix). De esta forma cada branch creada mantenía un nombrado uniforme de la siguiente manera:

- Tipo de desarrollo: feature, chore o fix.
- Código de ticket que se resuelve en JIRA.
- Información específica de lo que se resuelve.

Un posible ejemplo de esto sería: *feature/PF-14-login-screen*.

Siguiendo este formato se puede fácilmente relacionar branch a ticket para que luego, quien revise el código pueda verificar si todos los ítem del ticket fueron cumplidos o no.

Implementación

Como se mencionó previamente, el desarrollo se realizó teniendo en cuenta tres etapas, cada una de ellas con su propio conjunto de herramientas y dificultades.

5.1. Etapa 1: Modelado de la aplicación

En una primera instancia se investigaron los lineamientos y detalles característicos a aplicaciones iOS para luego realizar mocks del front end de la aplicación.

Para realizar esto, se utilizó la herramienta **Adobe Xd** del Creative Cloud de Adobe. Mediante la misma, se desarrollaron vistas para cada una de las pantallas de Ponte iOS sobre las cuales se obtuvo *feedback* apuntando a la usabilidad y conceptos de UX.

Finalmente, se iteraron estos diseños según los comentarios obtenidos y se llegó a un diseño preliminar para el front end de Ponte iOS.

5.1.1. ¿Qué caracteriza las aplicaciones iOS?

Tomando de los lineamientos oficiales, los tres temas primarios que diferencian a las aplicaciones iOS son: la claridad, la deferencia y la profundidad. Lo

que esto implica es un foco en la legibilidad y precisión; en utilizar los colores, las fuentes, los gráficos y otros elementos de la interfaz de manera inteligente para resaltar el contenido importante y transmitir interactividad; en movimientos fluidos y niveles distintivos visuales y movimientos realistas para transmitir jerarquía, dar vitalidad y facilitar el entendimiento. Los principios de diseño de iOS son la integridad estética, la consistencia, la manipulación directa, el feedback, el uso de metáforas y el control de usuario.

5.1.2. Consideraciones

Para diseñar la aplicación iOS de Ponte se tuvieron en cuenta diversas reglas importantes que deben seguir las aplicaciones móviles. Por un lado, la idea principal es que la aplicación debe ser simple de tal forma que todo el contenido sea fácilmente alcanzable para el usuario.

Por otro lado, se tuvo en cuenta que el 49 por ciento de las personas utilizan solo un pulgar para manejar su teléfono celular. En la siguiente figura se pueden observar en verde, las áreas fáciles de llegar con el pulgar, en amarillo, aquellas para las cuales debe hacer un esfuerzo adicional y en rojo, aquellas que requieren un cambio en la forma en que el usuario sostiene el teléfono.

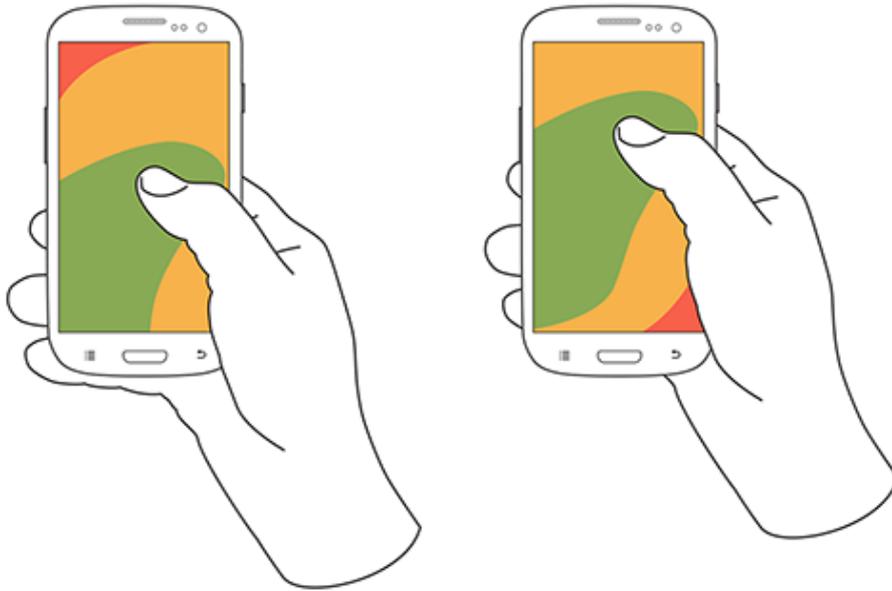


Figura 5.1: Dos formas de sostener el teléfono móvil con mapa de calor según uso¹.

Al diseñar una interfaz de usuario es importante mantener las acciones más frecuentes en la parte inferior de la pantalla. Con dicho fin se decidió implementar lo que se conoce como bottom tab bar. Consiste en una barra en la parte inferior de la pantalla que permitirá al usuario acceder a las funcionalidades centrales con un solo toque. La misma debe evitar tener más de cinco funcionalidades y debe mostrar la ubicación actual dentro de la misma.

Más allá de estos lineamientos establecidos la idea es siempre que la aplicación tenga similitud con su par en Android y con Ponte en su versión web. De tal manera que un usuario que migre de una aplicación a la otra no tenga dificultades en usar esta versión y pueda ahorrarse el tiempo de capacitación.

¹blogs.adobe.com. 2020. Mobile navigation tab bar - Adobe. Disponible en: <https://blogs.adobe.com/creativecloud/mobile-navigation-tab-bar-and-pictorial-icons/> [Visitado el 20 de abril 2020].

5.1.3. Navegación

Navigation Bars

Siguiendo los lineamientos iOS, se cuenta con una barra en la parte inferior de la aplicación que permite la navegación por una serie de pantalla jerárquicas². En dicho sector se muestra el título de la ubicación actual lo cual ayuda a los usuarios entender lo que están viendo, así como también un botón estándar para ir para atrás. El uso de un botón estándar permite que los usuarios lo reconozcan fácilmente y entiendan su función. Adicionalmente, cabe aclarar que este botón cumple una sola función por lo que no se encuentran breadcrumbs de múltiples segmentos.

Search Bars

También se implementan search bars en lugares apropiados para facilitar la búsqueda entre listas de potencialmente muchos elementos³. Este search bar cuenta con la opción de ‘Clear’ que borra lo escrito y con un placeholder como un recordatorio de la posibilidad de buscar en ese contexto.

Tab Bars

Como se mencionó, el diseño preliminar de Ponte implementaba una botom tab bar⁴. La misma tenía cuatro funcionalidades: Pedidos, Donar, Instituciones y Perfil.

²Developer.apple.com. 2020. Navigation Bars - Bars - Ios - Human Interface Guidelines - Apple Developer. Disponible en: <https://developer.apple.com/design/human-interface-guidelines/ios/bars/navigation-bars/> [Visitado el 20 de abril 2020].

³Developer.apple.com. 2020. Search Bars - Bars - Ios - Human Interface Guidelines - Apple Developer. Disponible en: <https://developer.apple.com/design/human-interface-guidelines/ios/bars/search-bars/> [Visitado el 20 de abril 2020].

⁴Developer.apple.com. 2020. Tab Bars - Bars - Ios - Human Interface Guidelines - Apple Developer. Disponible en: <https://developer.apple.com/design/human-interface-guidelines/ios/bars/tab-bars/> [Visitado el 20 de abril de 2020].

Este aspecto luego se modificó para agregar la sección de Home. En esta tab bar se muestra la ubicación actual marcando el ícono correspondiente en negro frente al resto que es gris.

5.1.4. Diseño de referencia (Web y Android)

Para lograr mantener una uniformidad entre las versiones de Ponte fue importante realizar un relevamiento de las mismas, analizando su diseño y características principales. De esta forma permitiendo que un usuario pueda utilizar cualquiera de ellas y reconocerla como Ponte.

A continuación, se incluyen algunas capturas de ambas versiones a modo de referencia.

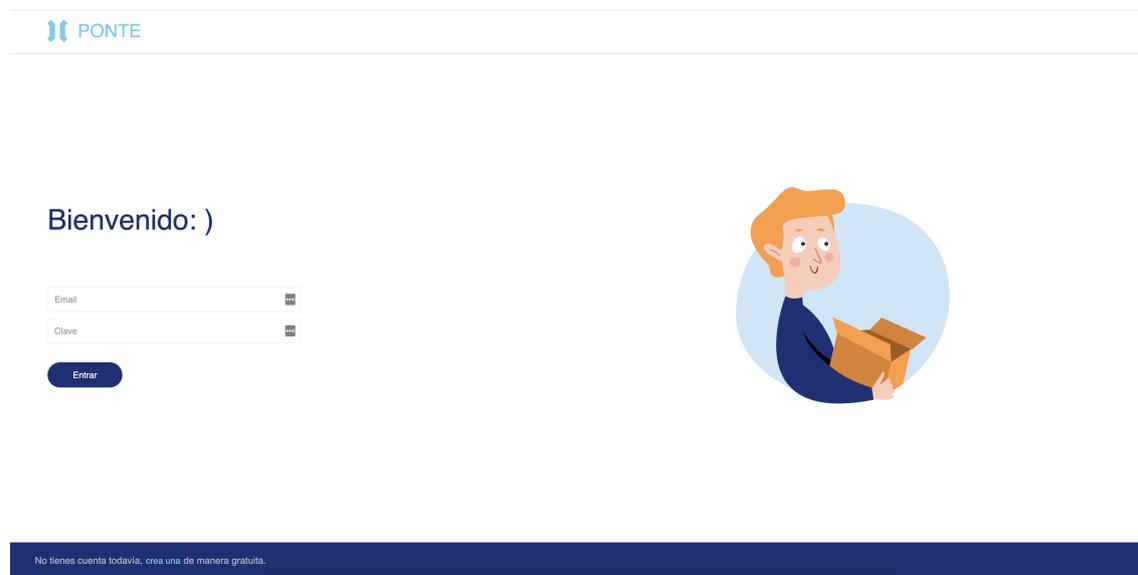


Figura 5.2: Inicio de sesión en versión web.

5.1. ETAPA 1: MODELADO DE LA APLICACIÓN



Figura 5.3: Perfil de usuario en versión web.

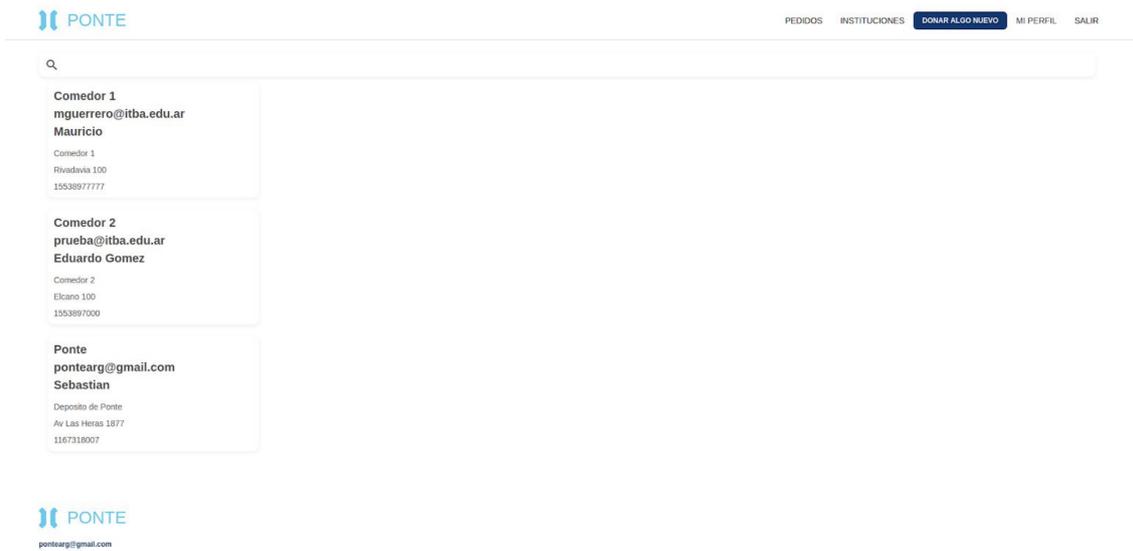


Figura 5.4: Listado de instituciones en versión web.

5.1. ETAPA 1: MODELADO DE LA APLICACIÓN

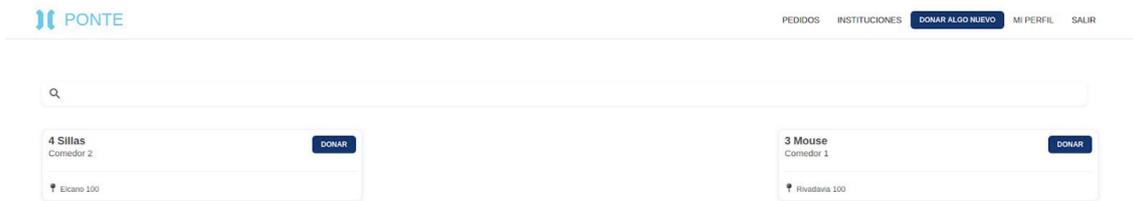


Figura 5.5: Pedidos en versión web.

Algunos estilos a notar que se mantuvieron son la paleta de colores, el logo que se muestra en la pantalla de Inicio de sesión junto con el *Bienvenido :)* y el formato general de las Cards para pedidos e instituciones.

A continuación, encontramos algunas capturas de la versión Android. Esta versión si bien se tomó en consideración, no contaba con todo el desarrollo realizado al momento del relevamiento por lo cual no se tuvo la versión final sino una intermedia.



Figura 5.6: Perfil en versión Android



Figura 5.7: Home en versión Android

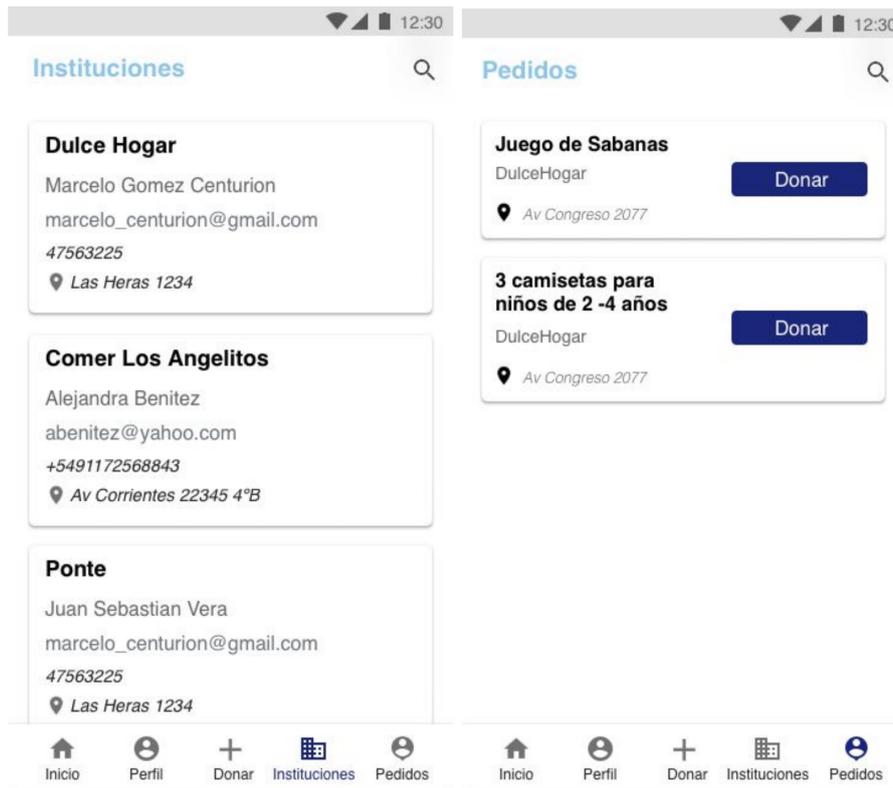


Figura 5.8: Instituciones y pedidos en versión Android

Aquí se puede ver un mismo estilo en el *Home* y *Perfil* junto con las Cards que también se encuentran presentes en la versión web.

5.1.5. Diseño preliminar

A continuación, se muestran capturas del diseño preliminar de Ponte iOS.

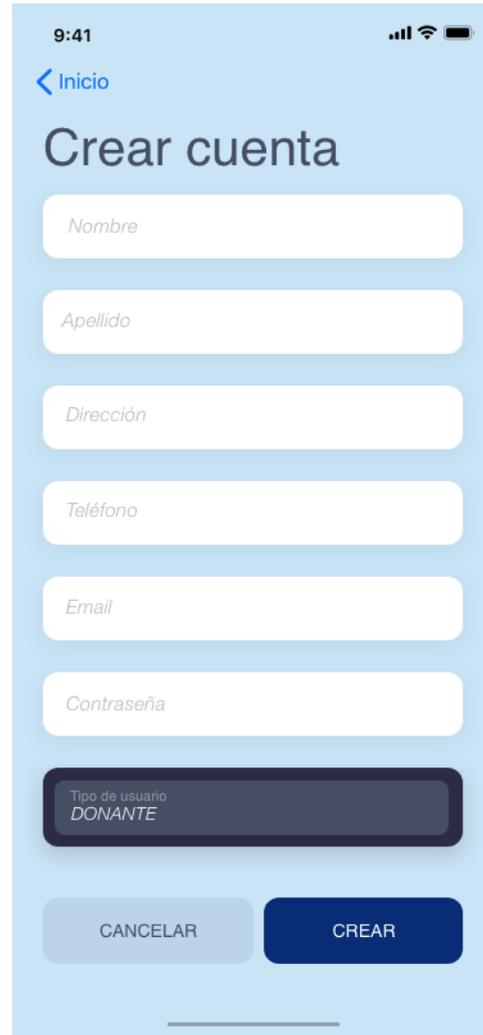
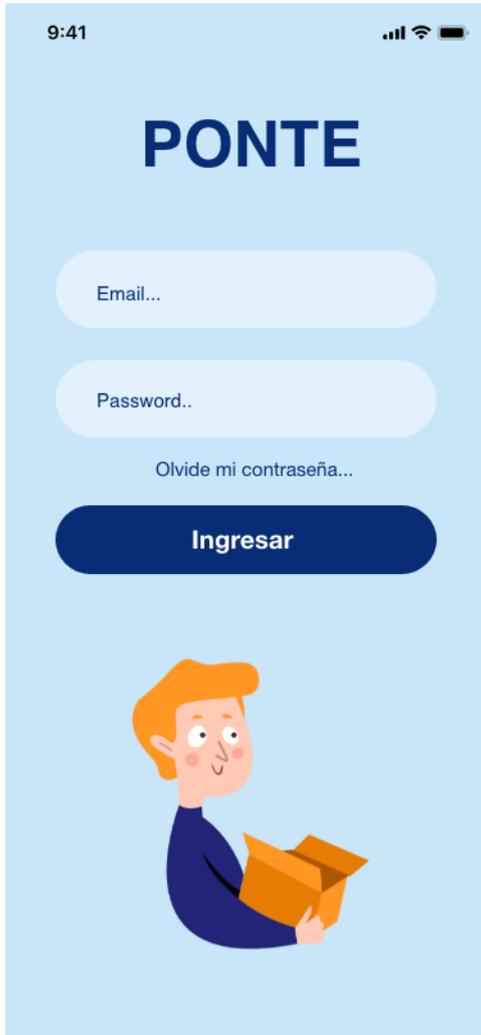


Figura 5.9: Boceto de inicio de sesión. Figura 5.10: Boceto de creación de cuenta.

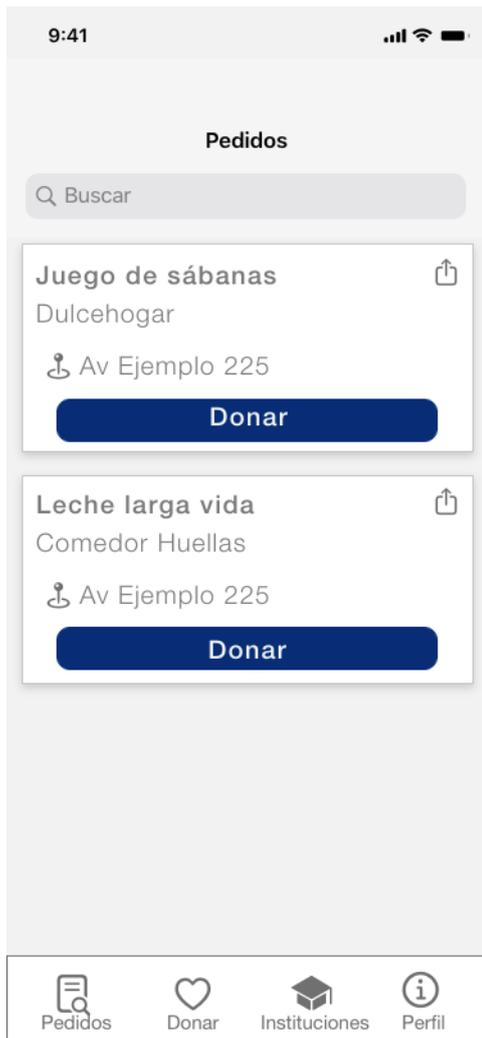


Figura 5.11: Boceto de pedidos.



Figura 5.12: Boceto de compartir pedido.

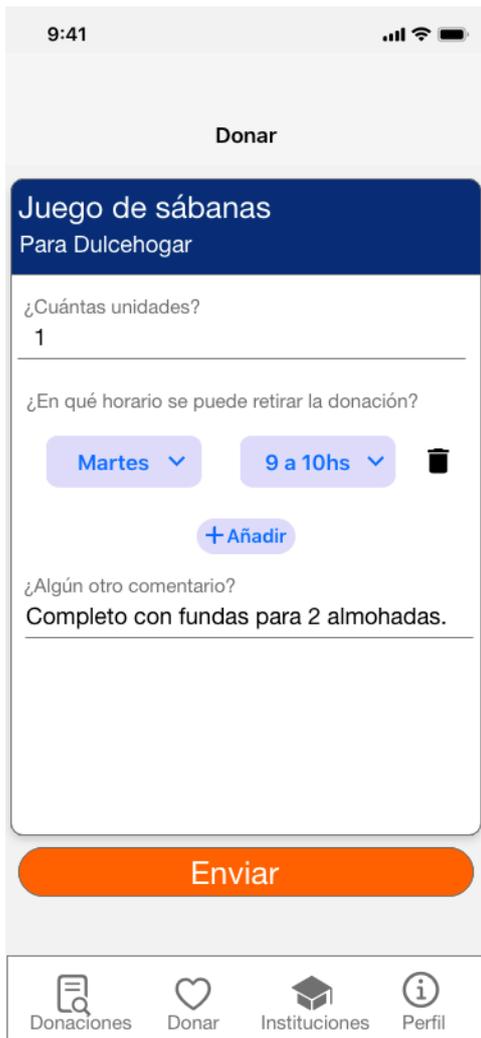


Figura 5.13: Boceto de realizar donación. Figura 5.14: Boceto de nueva donación.



Figura 5.15: Boceto de instituciones.

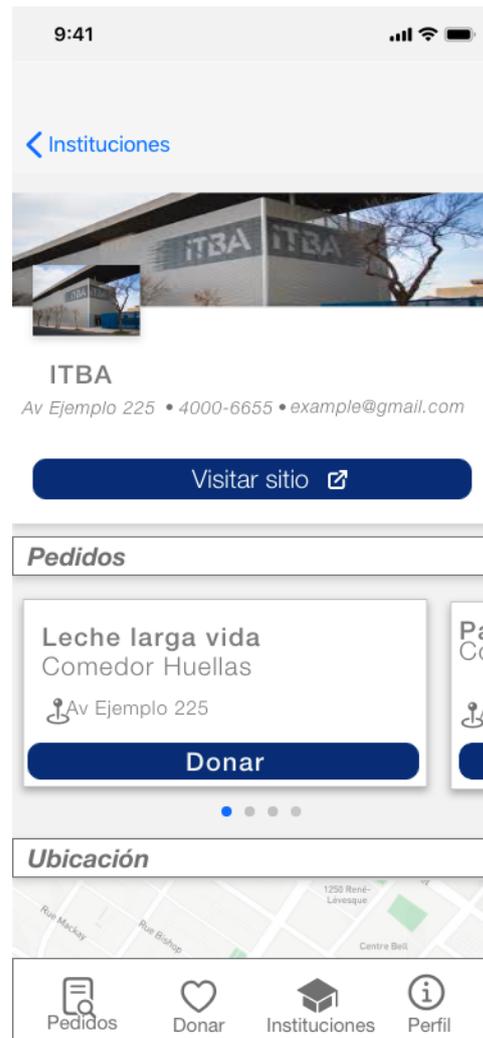


Figura 5.16: Boceto de detalle de institución.

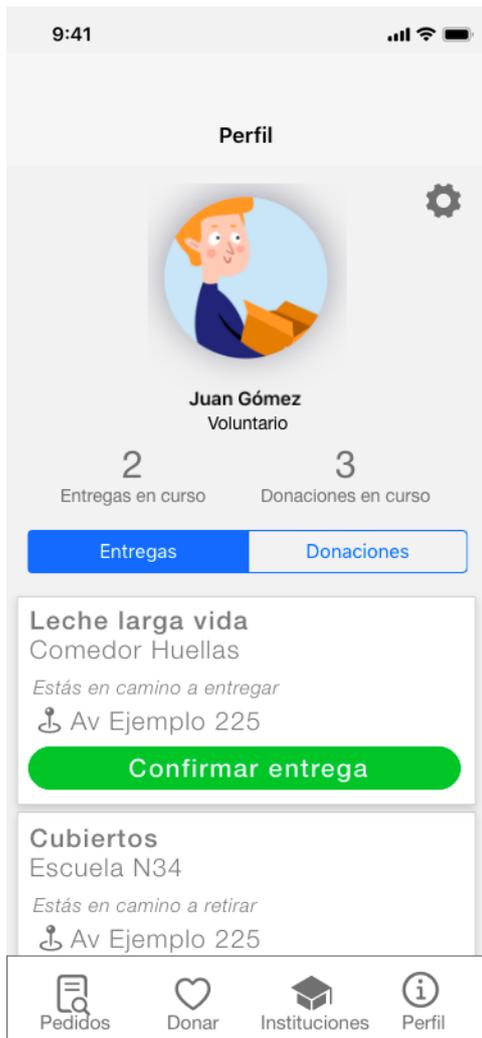


Figura 5.17: Boceto de perfil.

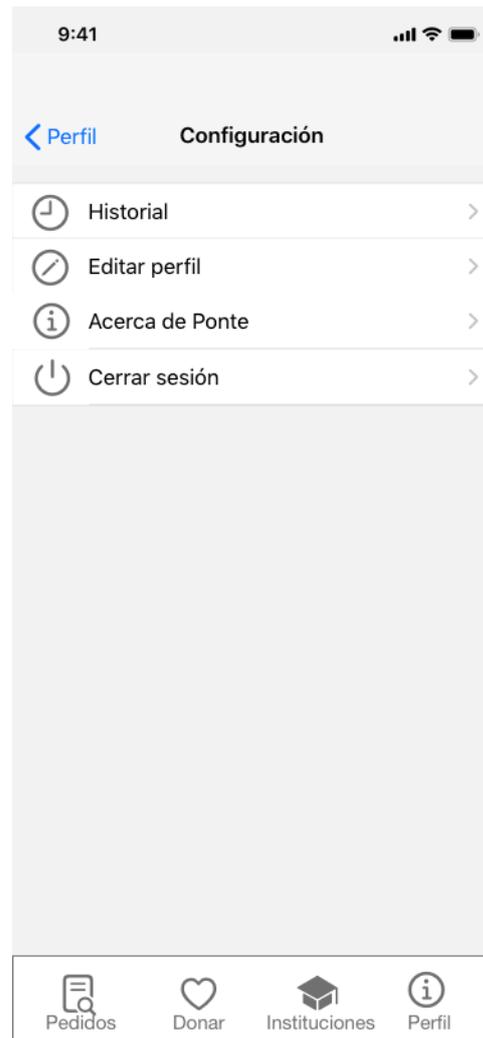


Figura 5.18: Boceto de configuración.



Figura 5.19: Boceto del historial.

5.2. Etapa 2: Funcionalidad Ponte Web

Desde un comienzo el objetivo de la aplicación iOS fue el de complementar y expandir la versión web. Es por esto que se desarrollaron todas las funcionalidades allí presentes y que los requerimientos de ambas versiones cuentan con puntos en común.

A continuación se detallan los requisitos funcionales y no funcionales que se desprenden de la versión web.

5.2.1. Funcionales

Autenticación

La aplicación debe permitir la creación de cuentas por parte de los usuarios. Esto se realiza ingresando nombre, apellido, dirección, email y contraseña. Luego, el sistema le enviará un correo electrónico a la casilla de email proporcionada para que éste active su nueva cuenta.

Una vez registrado, ese nuevo usuario contará con una cuenta de donante y deberá poder ingresar a la aplicación.

Adicionalmente, este usuario podrá pedir mediante un formulario, ser voluntario. Dentro de este formulario deberá ingresar las franjas horarias según su disponibilidad junto con el tamaño de bien que puede transportar (chico, mediano, grande) y algún comentario adicional si así lo desea.

Una vez enviada dicha solicitud el usuario deberá esperar a que la misma sea aprobada por algún administrador del sitio. Cuando esto sucede el usuario pasará a ser voluntario y deberá poder ingresar a la aplicación bajo ese estado.

Pedidos, ofertas y donaciones

La aplicación gira en torno a tres elementos: los pedidos que realizan las instituciones, las ofertas hechas por donantes y, en caso de ser voluntario, la donaciones que éste tiene asignadas.

Los usuarios, tanto voluntarios como donantes deben poder realizar una oferta que puede corresponderse a un pedido o no. En caso de que se corresponda se deberá ingresar las unidades que se desea donar, franjas horarias de disponibilidad y algún comentario adicional si así lo desea. En caso de que no se corresponda

con ningún pedido, se denomina donación y se debe ingresar todo lo mencionado previamente, agregando qué bien desea donar.

Las donaciones pasan por varios estados que serán detallados más adelante. Tanto donante como voluntario deberán poder interactuar con sus respectivas donaciones.

Los donantes deberán poder marcar las donaciones como entregadas al voluntario o cancelarlas en todo momento (excepto luego de entregarla). Los voluntarios a su vez deberán poder marcar cuando están yendo a retirar la donación, cuando la retiró y cuando la entregó a la institución correspondiente.

Instituciones

La aplicación deberá permitir acceder a un listado de instituciones. En el caso de Ponte iOS se agregó una vista con más información de cada institución así como también un listado de sus pedidos y un mapa con la ubicación. Esta funcionalidad se detalla con más profundidad en el sector de Funcionalidad Adicional.

5.2.2. No funcionales

Partiendo de Ponte Web y los requerimientos funcionales previamente definidos, se determinaron los siguientes atributos de calidad.

Usabilidad

Uno de los atributos que se considera más importante es el de usabilidad. Este atributo refiere a la facilidad de uso del sistema, cuan intuitiva es. Adicionalmente, considera atributos de accesibilidad que también se determinaron relevantes

dado el componente social de la aplicación.

Mantenibilidad

Dado que se sabe que va a haber cambios en el backend ya que hay un proyecto final en desarrollo y adicionalmente se estaba contemplando la posibilidad de lanzar Ponte al público se consideró importante la mantenibilidad del proyecto para facilitar los posibles cambios en la aplicación a futuro.

Confiabilidad

La aplicación debe ser tolerante a fallos. El atributo de confiabilidad indica la capacidad de la aplicación de seguir funcionando ante una falla de componentes del sistema. En caso de fallar algún componente, la aplicación debe informar de errores relevantes al usuario.

Performance

Adicionalmente, se consideró importante el atributo de Performance ya que este indica el grado de respuesta de la aplicación dentro de un intervalo de tiempo dado. Esto se tuvo en cuenta a la hora de los pedidos a la API, buscando optimizar la cantidad de llamadas para así mejorar los tiempos de respuesta y carga.

Vale aclarar que dado que se contaba con una API ya definida y limitada hubo algunos casos donde no se logró el estado ideal que se buscaba, principalmente en aspectos de paginación en los pedidos o filtrado de la información lo cual hubiera hecho que algunos pedidos sean más performantes.

Seguridad

Finalmente, se tomó el atributo de seguridad para prevenir la pérdida de información particularmente de las cuentas de los usuarios y el uso indebido de la aplicación. De esta forma no se permite usar la aplicación sin antes ingresar a una cuenta.

5.3. Etapa 3: Funcionalidad Adicional

Las funcionalidades adicionales implementadas en la versión final de Ponte iOS fueron las siguientes:

- Geolocalización del voluntario: Trackeo del estado de la donación en un mapa con geolocalización del transportista cuando el mismo se encuentra buscando la donación en la dirección del donante o llevando la misma hacia la institución correspondiente.
- Compartir pedido: Permitir que el donante comparta un pedido de donación a través de sus redes sociales para maximizar el alcance de ese pedido y que se cumpla más rápido el objetivo de que la institución reciba la donación.
- Edición de usuario: Permite a un usuario editar sus datos personales o su contraseña.
- Accesibilidad: Permite a usuarios utilizar la herramienta de VoiceOver para una lectura de las secciones de la aplicación.
- Detalles de la institución: Información detallada de una institución con un listado de sus pedidos y un mapa incorporado en la información de la institución mostrando su ubicación.

Estas funcionalidades actualmente, se encuentran únicamente en Ponte iOS. Sería ideal incorporarlas eventualmente a Ponte Web y Ponte Android. A continuación se detalla cada una de ellas.

5.3.1. Geolocalización del voluntario

El objetivo de esta funcionalidad es que el donante pueda geolocalizar al voluntario cuando el mismo esta yendo a buscar la donación en su dirección (estado BEING-LOOKED-FOR) o cuando la está llevando a la institución correspondiente (estado BEING-TAKEN-TO).

Para ello se hizo uso de la API de **PubNub**. PubNub es un servicio que permite el envío de datos en tiempo real mediante la creación de canales. En este caso, lo utilizamos para enviar las coordenadas del voluntario hacia el donante.

Cuando un voluntario utiliza la aplicación, publicará su ubicación en un canal. Este canal será la dirección de mail de dicho voluntario. A partir de ahí, cada 20 segundos, enviará su actualización de ubicación por dicho canal que correrá en segundo plano. Cabe destacar que para obtener la ubicación del voluntario, se utiliza el paquete `GetLocation` para React Native y cuando se quiera enviar la ubicación de usuario por primera vez, al mismo le aparecerá una alerta indicando que debe autorizar dicha operación.

Por otro lado, cuando un donante vea que su donación está siendo buscada en su domicilio verá un ícono que le indica que puede acceder a la pantalla de rastreo del voluntario. Cuando la donación haya sido retirada, y el voluntario la esté llevando al lugar de entrega, el donante también dispondrá de dicho ícono hasta que la donación sea entregada.

Al ingresar al ícono, la pantalla de Rastreo mostrará un mapa centrado en

la ubicación del voluntario simbolizado con el ícono de un vehículo. Un ícono de una casa mostrará la dirección del voluntario mientras que un ícono de una institución mostrará la dirección de la misma según el caso que corresponda.

Con respecto al funcionamiento en sí de esta pantalla, en este caso mediante el uso de PubNub, se hace una suscripción al canal del voluntario. Es decir, se actualizará el mapa cada vez que se publique una nueva coordenada en el canal correspondiente al email del voluntario asignado a dicha donación.

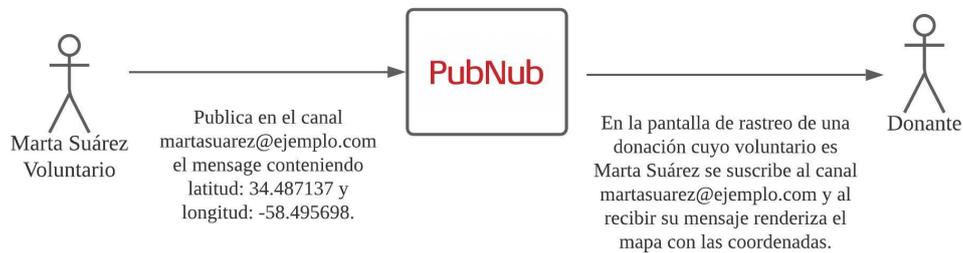


Figura 5.20: Gráfico indicando el funcionamiento de la geolocalización mediante PubNub.

Es importante notar que estamos utilizando una versión de prueba de PubNub que permite hasta un total de un millón de transacciones. Eventualmente, en caso de querer lanzar la aplicación, será necesario abonar la suscripción paga.

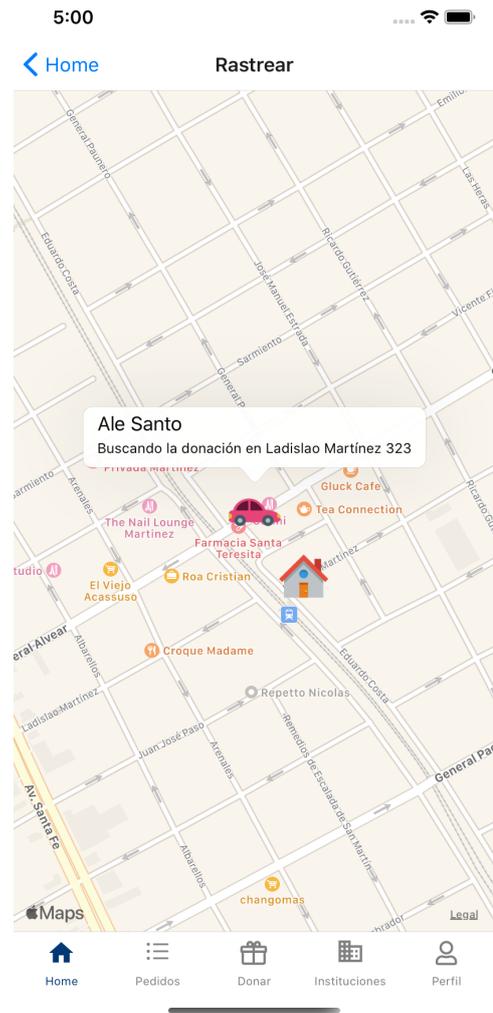


Figura 5.21: Aguardando ubicación del voluntario. Figura 5.22: Voluntario dirigiéndose a domicilio del donante.



Figura 5.23: Voluntario dirigiéndose a una institución.

5.3.2. Compartir pedido

Dicha funcionalidad se consideró con el objetivo de aumentar el alcance de la aplicación, y de esta manera facilitar y fomentar la realización de donaciones ya que si una persona no tiene el bien que se pide puede compartir el pedido con sus conocidos y posiblemente alguno de ellos lo tenga.

Para realizarlo se utiliza la funcionalidad *Share* de la librería React Native Share la cual abre el modal nativo de iOS. De esta manera se logra compartir mediante cualquier aplicación que se encuentre descargada en el celular del usuario

y que soporte esta funcionalidad. Particularmente se comparte solo texto ya que en la base de datos no se cuenta con imágenes. Sin embargo, en un futuro esto puede fácilmente incluirse.

Adicionalmente, una posible mejora que podría agregarse a Ponte en caso de lanzar la aplicación dentro del **App Store** sería la de incorporar los Dynamic Links provistos por Firebase. Este tipo de link permite ser abierto desde cualquier plataforma (móvil o PC) y lleva al usuario a la aplicación si se cuenta con esta descargada o a la tienda en caso negativo. Sin embargo, para implementar esto en iOS habría que ser miembro del **Apple Developer Program**, el cual requiere de una suscripción paga. Una vez hecho esto se puede crear un link dinámico en Firebase compatible con la aplicación iOS.



Figura 5.24: Pedido con el ícono de compartir pedido.

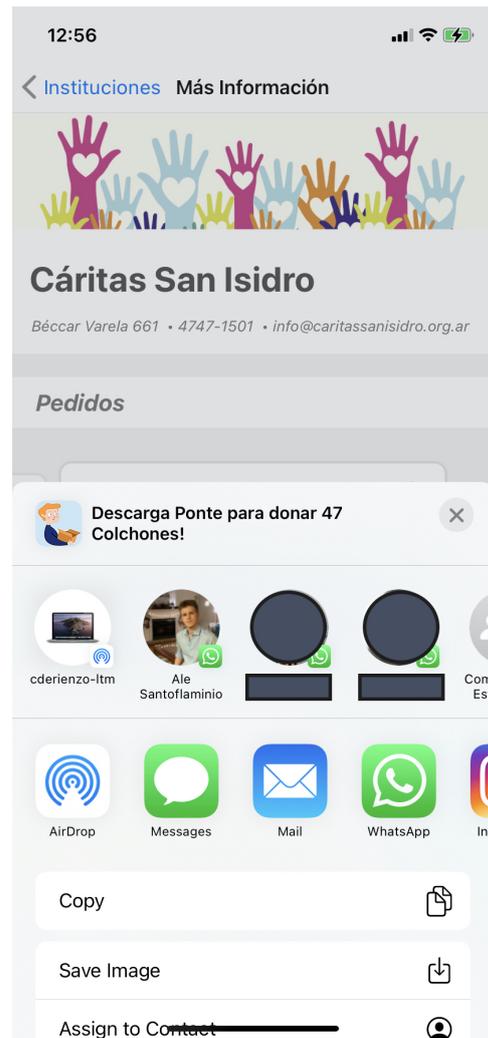


Figura 5.25: Menú para compartir un pedido de donación.

5.3.3. Edición de usuario

Otra funcionalidad incorporada, que no se encuentra en las otras plataformas de Ponte fue la edición de datos del usuario.

Aprovechando los métodos ya existentes sobre la API, se agregó la pantalla de 'Editar datos'. Esta pantalla se accede desde Perfil, ingresando en el ícono de configuración y permite al usuario cambiar su nombre, apellido, dirección y/o teléfono.

Por otro lado, también desde configuración dentro de Perfil, se puede acceder a Cambiar contraseña. Permite al usuario cambiar su contraseña, previamente ingresando la actual. Dicho cambio se hace efectivo en Firebase.

5.3.4. Accesibilidad

Esta funcionalidad se consideró relevante teniendo en cuenta la naturaleza de la aplicación y el componente social que esta tiene.

Para llevar a cabo dicha mejora se utiliza el **accessibilityLabel** dentro de los componentes principales de la aplicación para asegurar que el flujo de la misma se pueda realizar utilizando la herramienta de **VoiceOver** que se encuentra integrada dentro de iOS.

5.3.5. Detalle de una institución

Se agrega una vista detallada de una institución incluyendo un listado de sus pedidos junto con un mapa indicando su ubicación.

La funcionalidad se incorpora con el objetivo de completar la pantalla de institución y permitir que el donante pueda obtener más información sobre una institución de interés.

Para lograr indicar en un mapa la ubicación se debe llevar a cabo lo que se denomina *Geocoding*. Dicho proceso consiste en tomar una dirección e interpretarla a latitud y longitud. Esto se realiza a través de una API que provee **HERE Maps**.

Actualmente, se cuenta con una suscripción dentro de un Free Tier. Esta suscripción provee hasta 250k transacciones por mes y 5k usuarios activos. En caso de querer lanzar la aplicación con esta funcionalidad se deberá o pasar a una

suscripción superior o buscar alguna otra solución alternativa

5.4. Diseño final

En esta sección se detalla la implementación final de Ponte iOS, la cual difiere levemente de los diseños preliminares mostrados anteriormente.

5.4.1. Navegación

En concordancia con lo diseñado en una primera etapa, la navegación en Ponte iOS se logra mediante la *bottom tab bar*. A diferencia del diseño preliminar se decidió agregar un quinto elemento a la misma, el Home. Esta sección estaba presente en Ponte Android, y para limitar las diferencias entre ambas aplicaciones se decidió incorporarla, además de considerarse que es un agregado importante para la aplicación.



Figura 5.26: *Bottom tab bar*.

Cada sección de la *bottom tab bar* tiene su propio stack. Esto quiere decir que por ejemplo, si navegamos dentro de *Pedidos* hacia alguno en particular, y luego tocamos en *Perfil* y vamos hacia el historial del usuario, al volver a tocar en *Pedidos* nos encontraremos en el pedido al que habíamos navegado en dicha sección. Es decir que se respeta el stack de cada pestaña. Este comportamiento es similar al de aplicaciones como Facebook o Instagram.

5.4.2. Bienvenida

Al lanzar la aplicación de Ponte se presentará una *splash screen* por un breve tiempo hasta que cargue la aplicación. La función de una pantalla de carga es la de dar a los usuarios la impresión de una aplicación rápida y *responsive*, permitiendo a su vez la carga del contenido inicial.



Figura 5.27: *Splash screen*.

5.4.3. Inicio de sesión y creación de usuario

Luego de la *splash screen*, se presentará una pantalla donde el usuario puede ingresar su correo y su contraseña para ingresar a la aplicación, o bien proceder a crearse un usuario al tocar en *Crearse una cuenta*.

Cabe destacar, que si el usuario previamente había iniciado sesión en la aplicación y no la había cerrado, esta pantalla no se mostrará e ingresará directo a *Home*.

En caso de ingresar correctamente su correo y contraseña, el usuario ingresará a la aplicación ubicándose en la pestaña *Home*. En caso de algún error, se le mostrará un mensaje al usuario.

Por otra parte, si el usuario no tiene cuenta, puede proceder a crearse una. Dicha pantalla consta de un formulario con los campos requeridos por el servidor de Ponte para cargar un usuario a la base de datos. Los campos se encuentran validados correspondientemente para indicarle los errores que pueden surgir al usuario. En caso de crearse una cuenta con éxito, procederá a iniciar sesión automáticamente y posicionarse en *Home*.



Figura 5.28: Pantalla de bienvenida.



Figura 5.29: Creación de usuario.

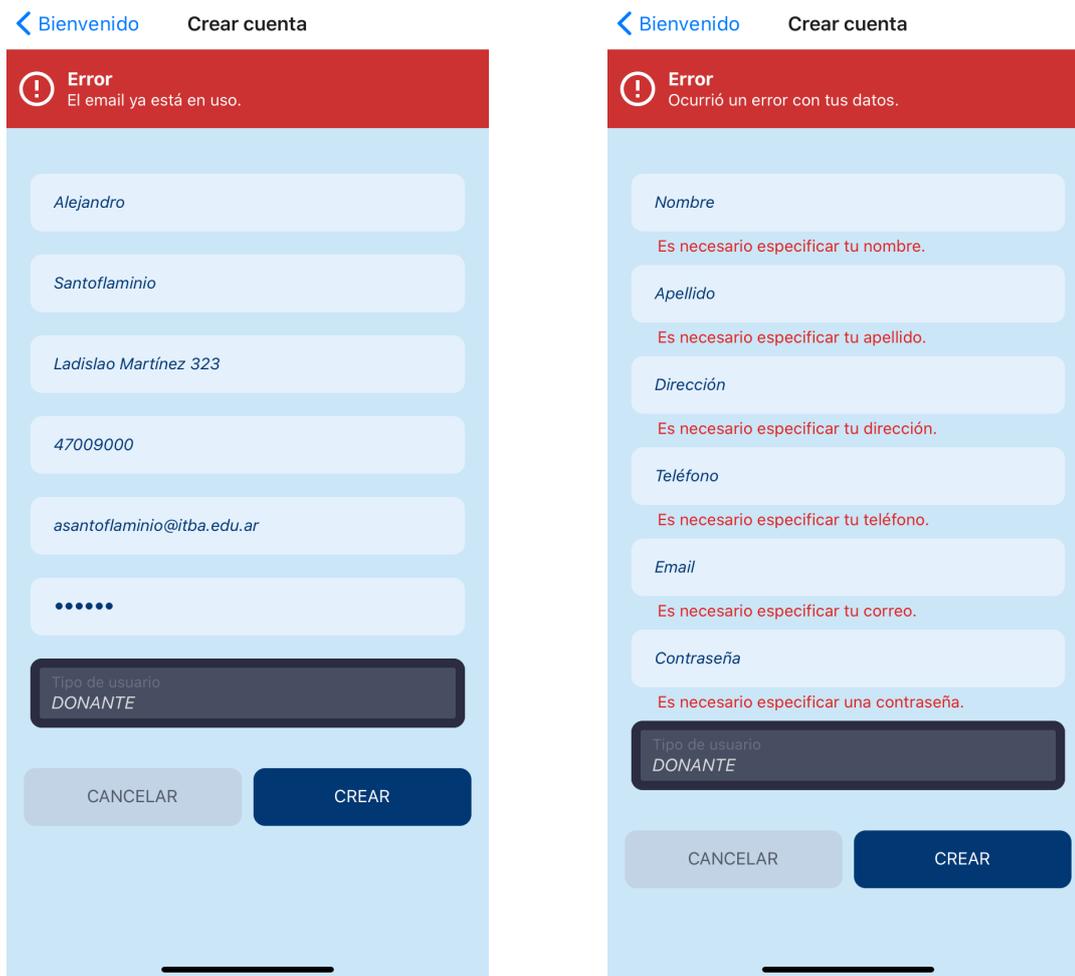


Figura 5.30: Error al usar mail ya registrado. Figura 5.31: Error al no completar datos.

5.4.4. Home

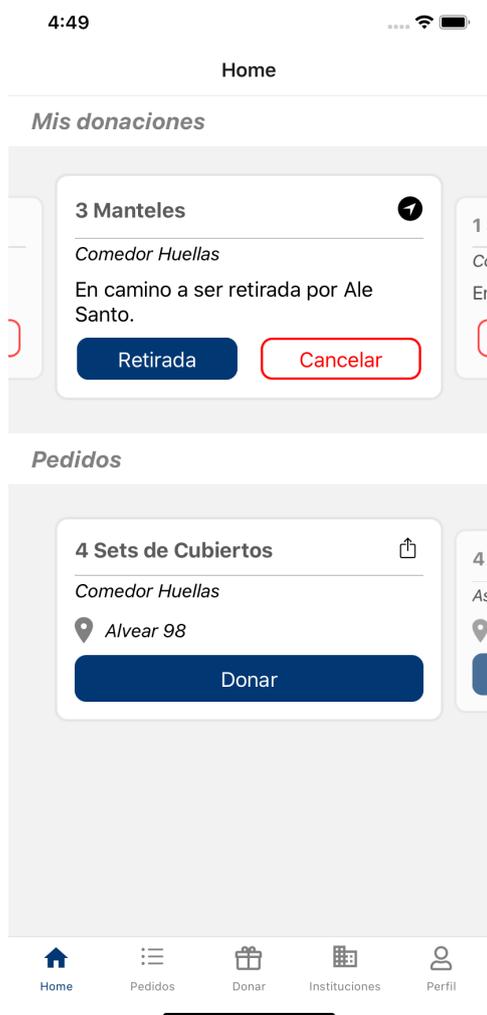
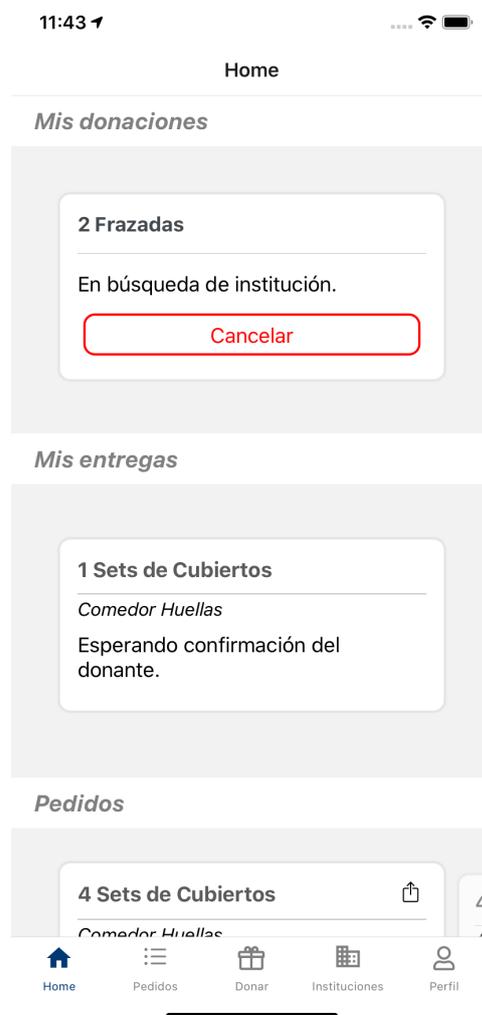
Como se mencionó, *Home* es la pestaña donde se ubica el usuario al ingresar a Ponte, una vez iniciada su sesión.

Home representa un resumen de información para el usuario. En caso de tratarse de un donante, verá en esta sección las donaciones que tiene en curso así como los pedidos que existen actualmente en Ponte. Si se trata de un voluntario también podrá ver sus entregas en curso.

Cada donación o entrega en curso tiene su tarjeta, y se muestran al usuario

en forma de carrusel. Los estados y las opciones que muestra cada tarjeta dependiendo del tipo de usuario se encuentran explicados más adelante, pero es importante notar que desde *Home* el usuario puede interactuar con sus donaciones o entregas para llevar a cabo el flujo de las mismas.

Con respecto a los pedidos, también cada uno tiene su tarjeta y están mostrados en forma de carrusel. Se puede ingresar a una en particular tocando en *Donar*, lo cual luego mostrará un formulario y en caso de éxito una pantalla de confirmación de donación. Estas pantallas las veremos en la próxima sección (*Pedidos*). Cada tarjeta tiene su funcionalidad de compartir incorporada como se mencionó anteriormente.

Figura 5.32: *Home* de un donante.Figura 5.33: *Home* de un voluntario.

5.4.5. Pedidos

En esta pestaña se le muestra inicialmente al usuario un listado completo de todos los pedidos que existen en Ponte. Dispone de una barra de búsqueda que le permite al usuario buscar alguna solicitud en particular.

De la misma forma que en *Home*, cada pedido de donación tiene su tarjeta, pero en este caso dispuesto a modo de lista. Nuevamente cada una con su funcionalidad de compartir.

En caso de tocar en *Donar*, se procederá a un formulario donde el usuario debe indicar la cantidad de unidades que desea donar, así como los horarios en los que desea que un voluntario pase a buscar la donación. Adicionalmente, puede indicar un comentario si así lo desea. Dicho formulario cuenta con las validaciones correspondientes.

Al tocar en el botón naranja de *Donar*, se mostrará una pantalla de confirmación de la donación, o bien una alerta con el error correspondiente.

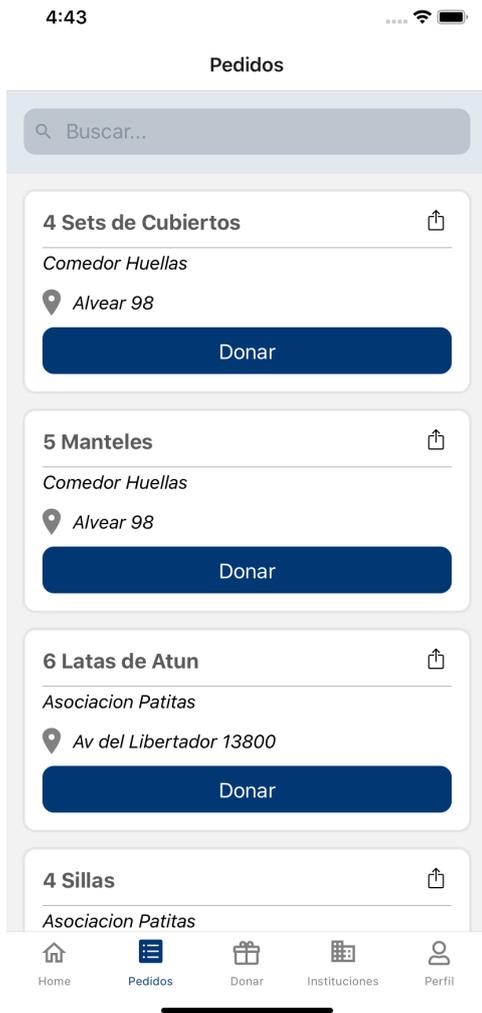


Figura 5.34: *Pedidos*.



Figura 5.35: Realizar una donación.

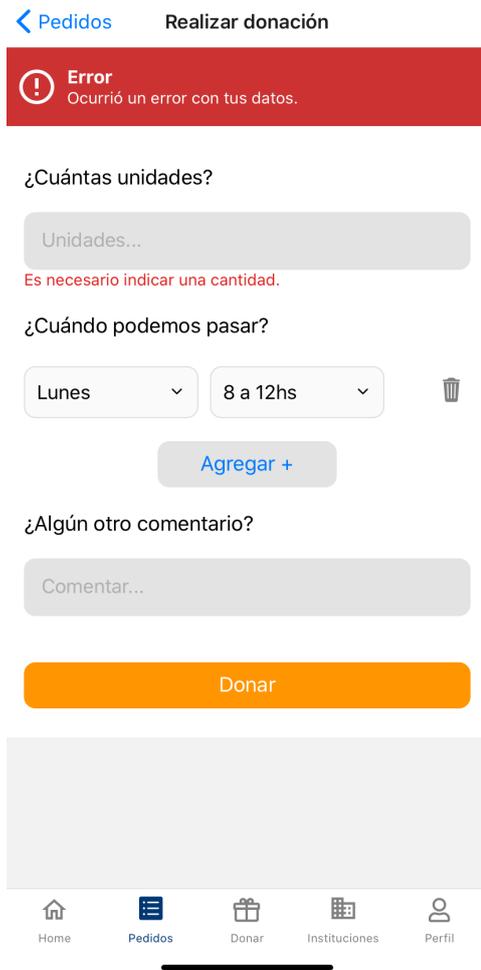


Figura 5.36: Error al no completar datos. Figura 5.37: Error al elegir horarios repetidos.

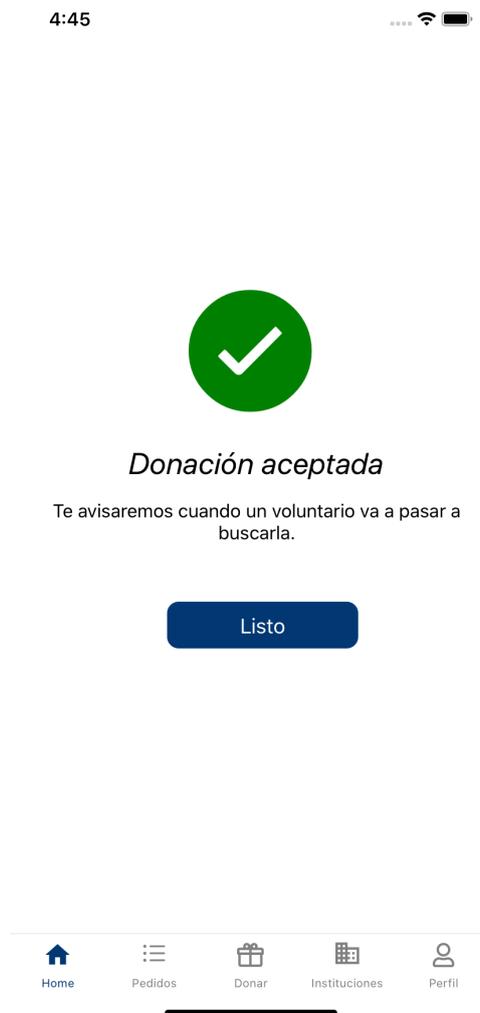


Figura 5.38: Donación confirmada.

5.4.6. Donar

Desde *Donar* un usuario puede realizar una donación que no se corresponde con un pedido realizado por una institución. Es decir, el donante puede donar lo que desea especificando el bien a donar, la cantidad de unidades, sus horarios disponibles y un comentario. Nuevamente se cuentan con las validaciones correspondientes.

Al realizar esta oferta sin pedido, la donación se encontrará a la espera de una institución para poder continuar con su flujo normal. También podría ocurrir

que un administrador la acepte y se guarde en depósito.

Figura 5.39: *Donar*.

Figura 5.40: Error al no completar campos.

5.4.7. Instituciones

En *Instituciones* se muestra un listado completo de instituciones registradas en Ponte. Al navegar dentro de cada una de ellas podemos visualizar detalles de la misma. Allí veremos además, los pedidos específicos de esa institución así como su ubicación en un mapa.

Como se mencionó previamente, la ubicación en el mapa se logra mediante el uso de la API provista por HERE Maps.

En caso de no poder ubicarla en el mapa ya sea por que la API utilizada está caída o porque la dirección ingresada es insuficiente o errónea, se muestra un mensaje de error.

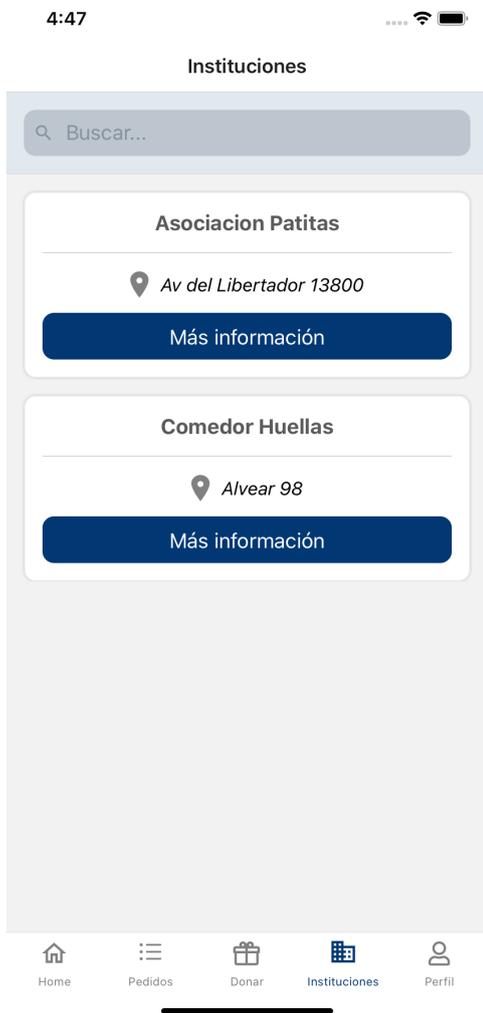
Figura 5.41: *Instituciones*.

Figura 5.42: Detalle de institución.

5.4.8. Perfil

Perfil muestra al usuario en su pantalla inicial información acerca del mismo.

En la parte inferior veremos el listado de donaciones correspondientes a ese usuario, nuevamente con las tarjetas similares a las que encontramos en *Home*

y cuyo comportamiento explicaremos en la próxima sección. En caso de tratarse de un voluntario, dispondra de una tab segmentada para alternar entre sus donaciones o sus entregas. Si un donante quisiera convertirse en voluntario, dispone del botón 'Quiero ser voluntario' para realizar dicha solicitud.

Mediante la tuerca de la parte superior derecha, el usuario navega hacia configuración. Desde configuración el usuario puede acceder a su historial, a la pantalla de editar sus datos o su contraseña, o por otra parte, cerrar su sesión.

Desde historial, puede visualizar un listado de donaciones o entregas según corresponda.

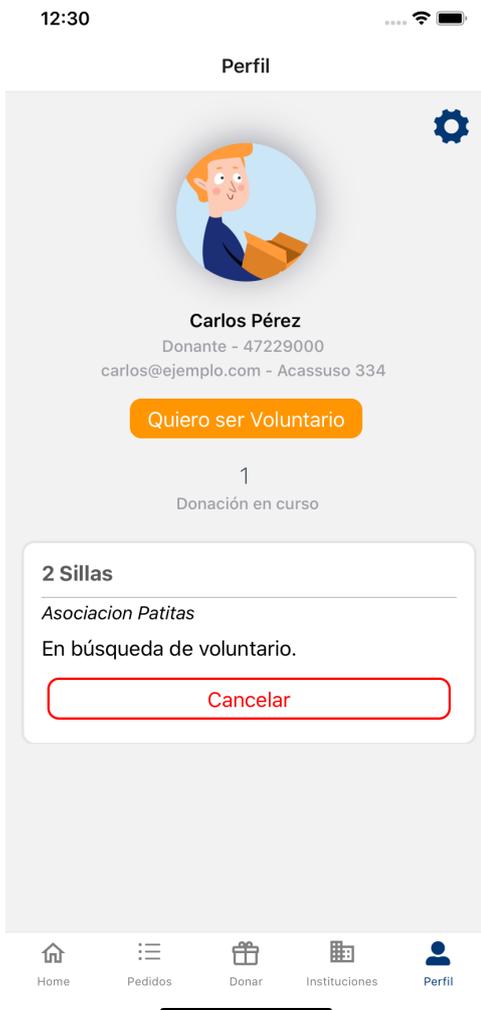


Figura 5.43: Perfil de un donante.



Figura 5.44: Formulario para ser voluntario.

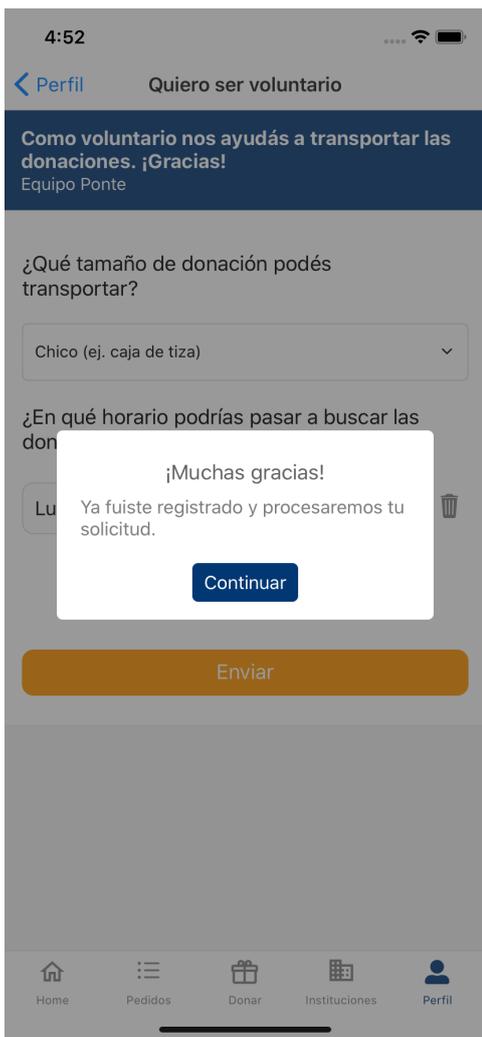


Figura 5.45: Alerta al solicitar ser voluntario.



Figura 5.46: Donante con solicitud para ser voluntario.

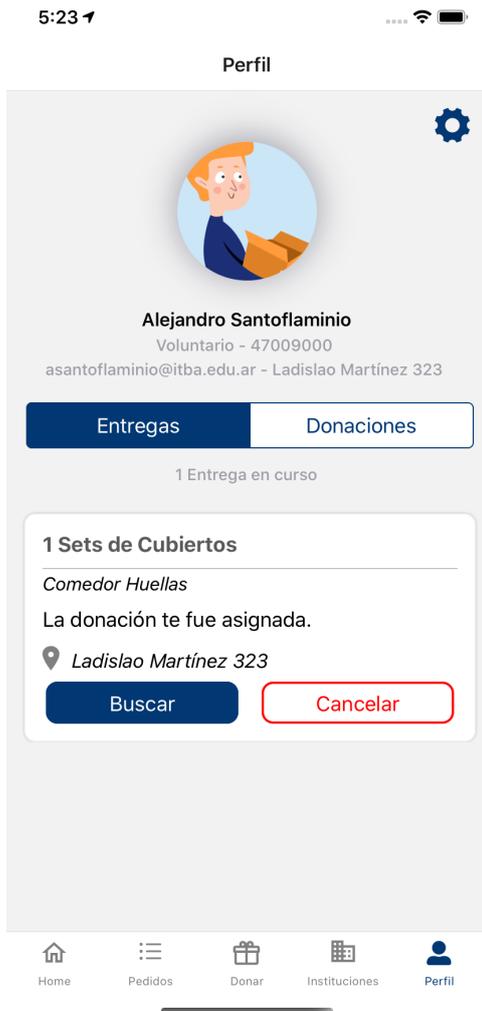


Figura 5.47: Perfil de un voluntario.

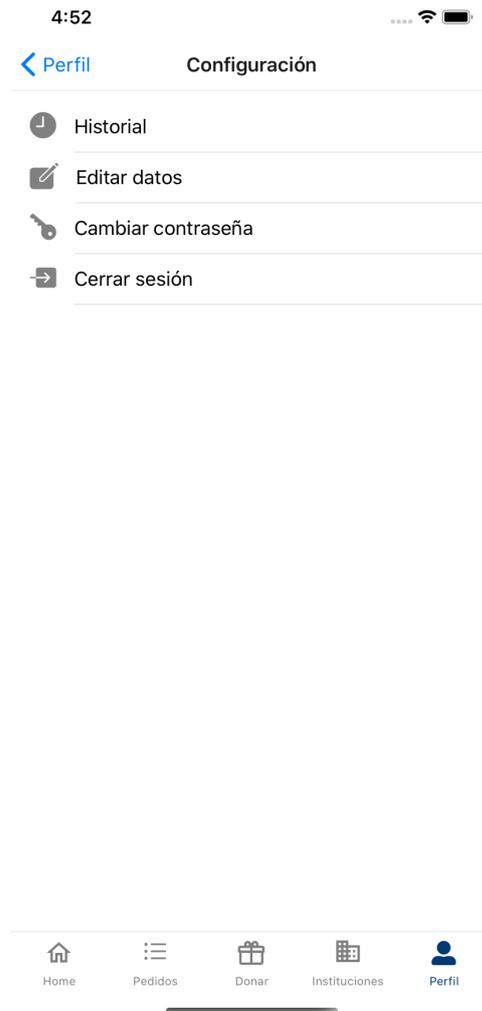


Figura 5.48: Menú de configuración.

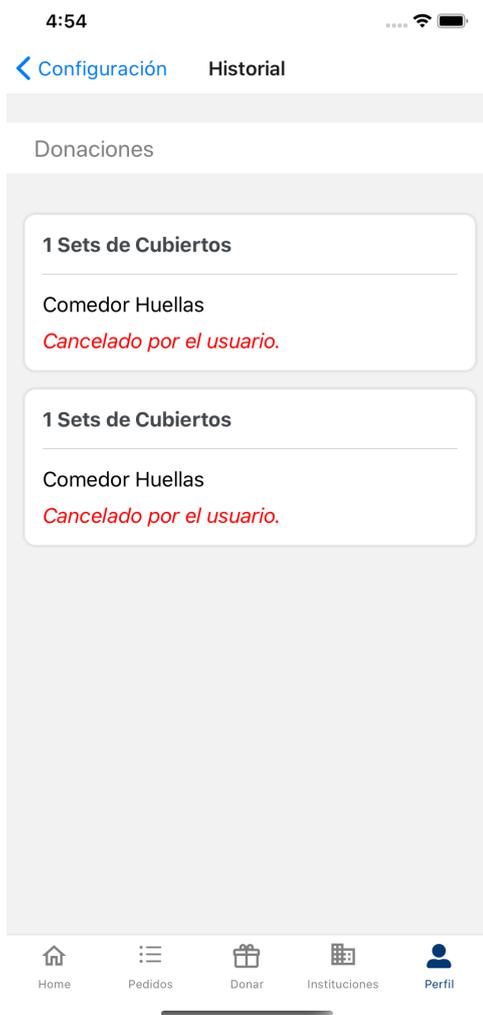


Figura 5.49: Historial de un donante.

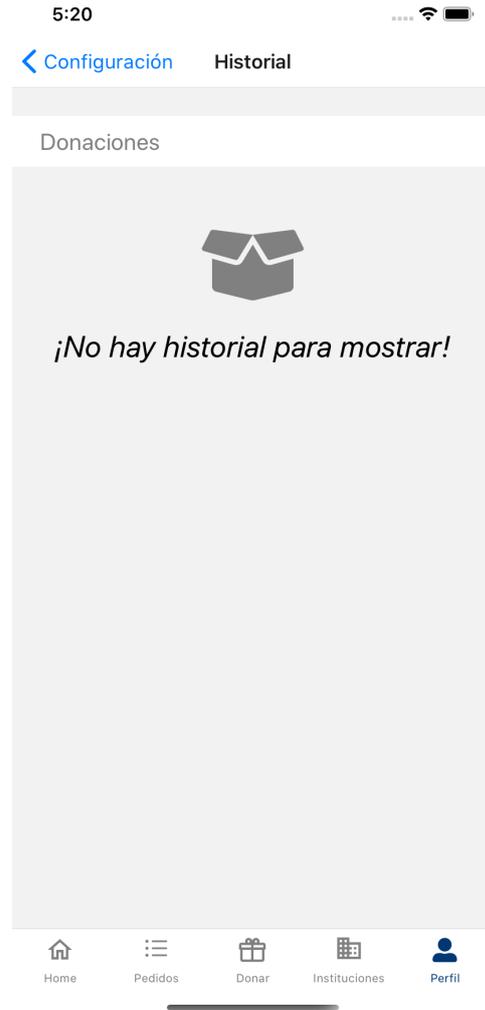


Figura 5.50: Historial de un donante sin donaciones previas.

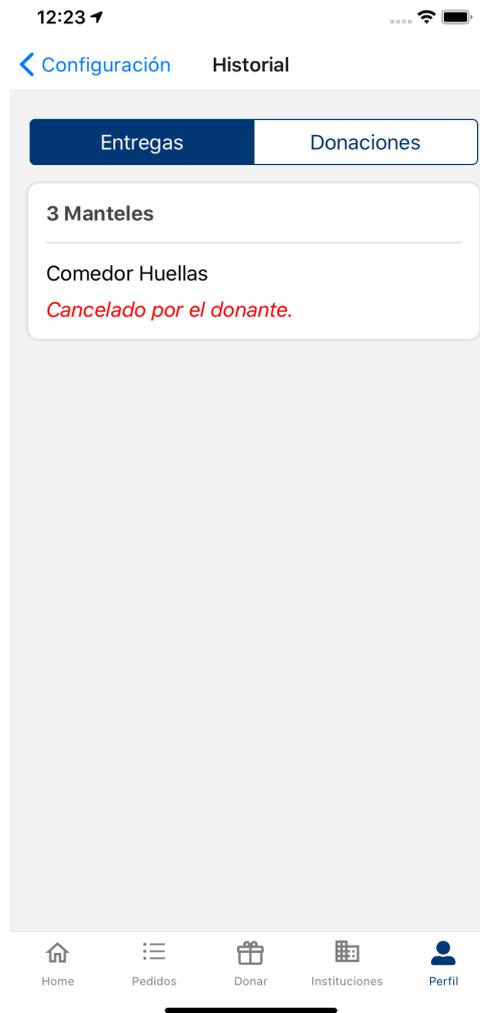


Figura 5.51: Historial de un voluntario.

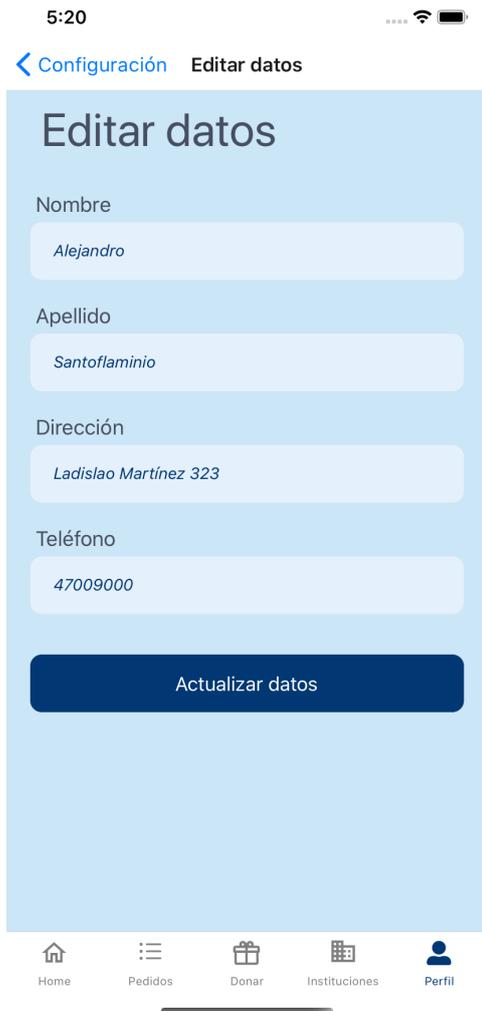


Figura 5.52: Editar datos.

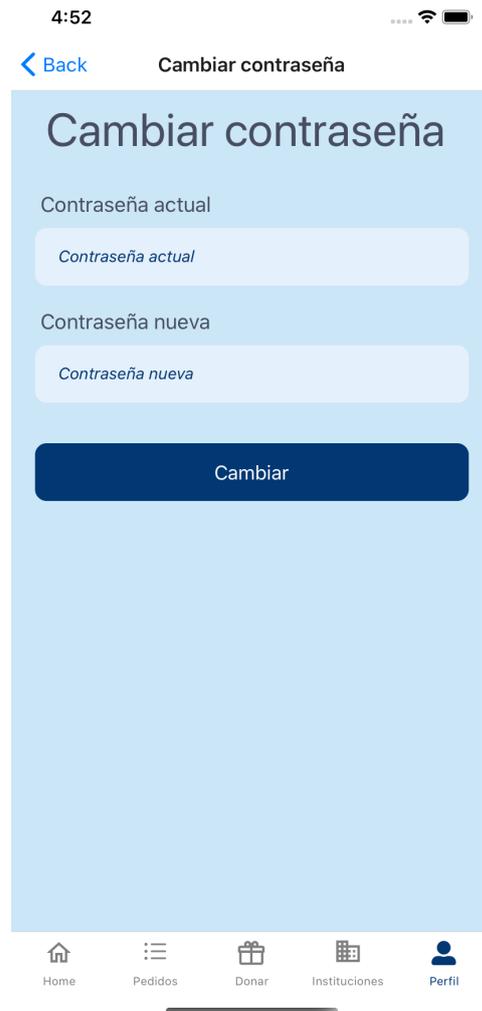


Figura 5.53: Editar contraseña.

5.4.9. Estados de una donación

Como se mostró previamente, las donaciones atraviesan distintos estados. En esta sección iremos viendo cada uno y mostrando en comparación para una misma donación como se le muestra al donante, y como se le muestra al voluntario.

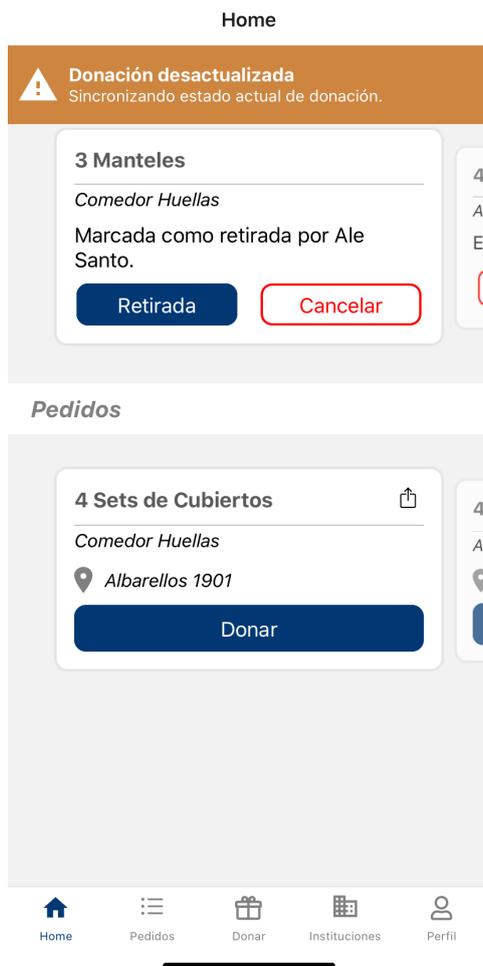


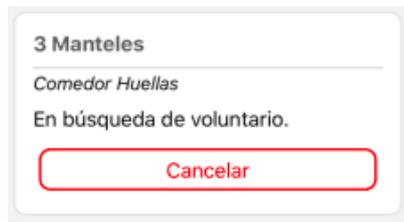
Figura 5.54: Al interactuar con una donación desactualizada se muestra una advertencia y se recarga el contenido.

CREATED

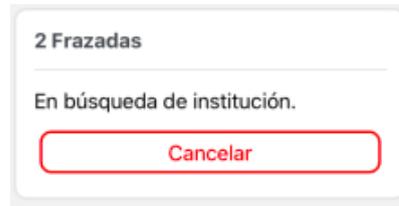
Al realizar una oferta de manera exitosa a un pedido, el donante verá su donación en estado **CREATED** durante el cual se le indicará que se está buscando un voluntario para asignarle a dicha donación. En caso de hacer una donación sin pedido específico, la misma se mostrará indicando que está en búsqueda de institución (en tal caso se deberá esperar a que un administrador le asigne un pedido de una donación).

En esta etapa todavía no está involucrado el voluntario. El donante en este

momento solo puede optar por cancelar la donación.



(a) Tarjeta mostrada al donante al crear una donación a un pedido.



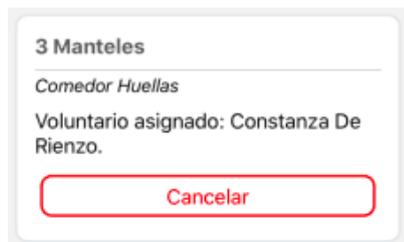
(b) Tarjeta mostrada al donante al crear una donación sin un pedido específico.

Figura 5.55: Estado CREATED

CREATED con voluntario asignado

Como se mencionó, en la versión desplegada de la API hasta noviembre de 2020, existía el estado VOLUNTEER-ASSIGNED. En la última versión que se encontraba en el repositorio de Ponte, éste ya no se encuentra debido a la modificación de estados que está llevando a cabo el equipo encargado de continuar el proyecto Ponte. Sin embargo, la ausencia de este estado no implica grandes cambios al funcionamiento. Simplemente se pasó a utilizar el estado CREATED de la misma manera que se utilizaba el estado VOLUNTEER-ASSIGNED cuando éste tenía un voluntario asignado.

En este momento el donante tiene la posibilidad únicamente de cancelar la donación. Desde el lado del voluntario, el mismo puede o bien cancelar su asignación a dicha donación (en tal caso volverá a estado CREATED sin voluntario) o bien pasar a buscarla (estado BEING-LOOKED-FOR).



(a) Tarjeta mostrada al donante.



(b) Tarjeta mostrada al voluntario.

Figura 5.56: Estado CREATED con voluntario asignado

BEING-LOOKED-FOR

En esta instancia, el voluntario se encuentra buscando la donación en el domicilio del donante. Como se mencionó previamente, el donante puede empezar a geolocalizar al voluntario accediendo al ícono de localización.

El donante puede en este caso, cancelar la donación o bien marcarla como retirada (la donación pasa a DELIVERED-BY-DONOR). Desde el lado del voluntario, el mismo puede cancelar su asignación a dicha donación (nuevamente pasa a estar CREATED sin voluntario) o marcarla como retirada (PICKED-BY-VOLUNTEER).



(a) Tarjeta mostrada al donante. (b) Tarjeta mostrada al voluntario.

Figura 5.57: Estado BEING-LOOKED-FOR

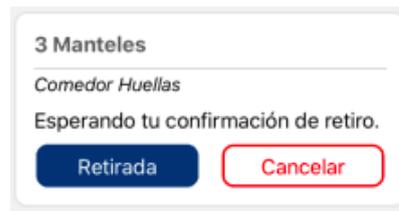
DELIVERED-BY-DONOR

En caso de que el donante haya marcado que el voluntario ya retiró la donación en su domicilio, la misma pasará a DELIVERED-BY-DONOR. En este punto, el donante no tiene ninguna opción para cambiar el flujo de la donación, ya que debe esperar a que el voluntario confirme su retiro.

Por parte del voluntario, el mismo puede indicar el retiro de la donación (en tal caso pasará a BEING-TAKEN-TO) o bien cancelarla. Sería interesante debatir si permitirle cancelar al voluntario una donación en esta etapa sería erróneo, ya que sucedería que el donante ya la marcó como entregada y el voluntario la canceló. Sin embargo, se decidió mantener estas opciones para ser consistentes con la versión web de Ponte.



(a) Tarjeta mostrada al donante.



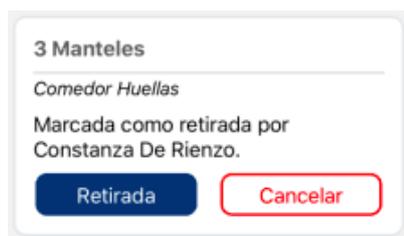
(b) Tarjeta mostrada al voluntario.

Figura 5.58: Estado DELIVERED-BY-DONOR

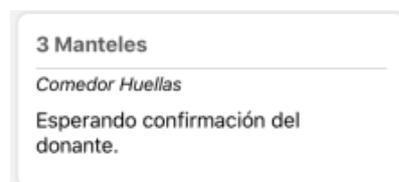
PICKED-BY-VOLUNTEER

Cuando una donación se encuentra en esta etapa, quiere decir que el voluntario indicó que ya buscó la donación en el domicilio del donante, pero el donante aún no confirmó dicha acción. El voluntario no puede hacer más que esperar la confirmación del donante, el cual tiene la opción de indicar la donación como retirada o bien cancelarla. Nuevamente es cuestionable, la acción de permitir cancelar la donación siendo que el voluntario ya la indicó como retirada, pero se mantuvo en consistencia con Ponte web.

Cuando el donante marca que la donación fue retirada, la misma pasa al estado BEING-TAKEN-TO.



(a) Tarjeta mostrada al donante.



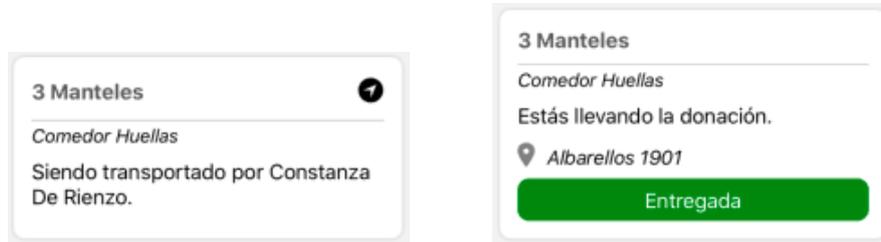
(b) Tarjeta mostrada al voluntario.

Figura 5.59: Estado PICKED-BY-VOLUNTEER

BEING-TAKEN-TO

En este punto, tanto el donante como el voluntario indicaron que la donación fue retirada, por lo tanto el voluntario se encuentra en camino a la institución.

El donante tiene la posibilidad únicamente de seguirlo a través de la geolocalización mientras que el voluntario podrá indicar la donación como entregada (sin posibilidad de cancelarla).



(a) Tarjeta mostrada al donante. (b) Tarjeta mostrada al voluntario.

Figura 5.60: Estado BEING-TAKEN-TO

DELIVERED

En esta etapa, el voluntario indicó que ya entregó la donación en la institución correspondiente.

Tanto el voluntario como el donante, solo podrán a partir de ahora ver la donación en su historial.



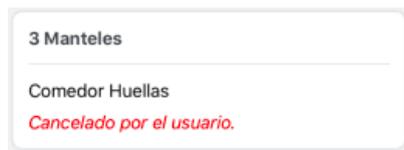
(a) Tarjeta mostrada en el historial de donaciones del donante. (b) Tarjeta mostrada en el historial de entregas del voluntario.

Figura 5.61: Estado DELIVERED

CANCELLED

En caso de que el donante cancele la donación en alguno de los estados en los cuales tiene dicha posibilidad, la misma pasará al estado CANCELLED.

Nuevamente, tanto el voluntario como el donante, solo podrán a partir de ahora ver la donación en su historial.



(a) Tarjeta mostrada en el historial de donaciones del donante.



(b) Tarjeta mostrada en el historial de entregas del voluntario.

Figura 5.62: Estado CANCELLED

5.4.10. Flujo sin conexión

A lo largo de las secciones anteriores se mencionaron y mostraron las validaciones y muestra de errores que realiza la aplicación. Otro aspecto a tener en cuenta dentro de la aplicación es el de una posible falla en la conexión. Es por ello que al fallar una solicitud a la API por problemas de conexión se muestra un mensaje correspondiente.

Cabe destacar que el usuario puede hacer un *pull to refresh* para volver a cargar el contenido, es decir deslizar la pantalla hacia abajo para desencadenar una recarga.

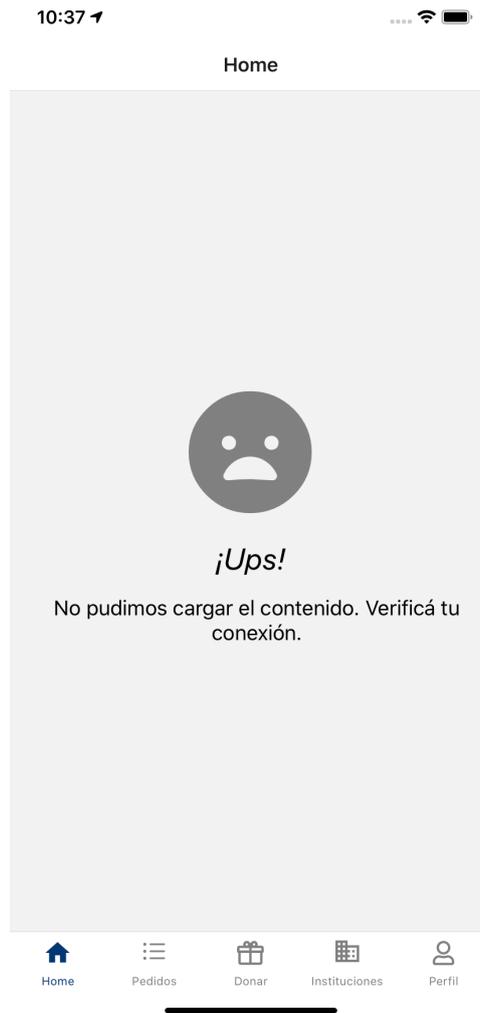


Figura 5.63: Pantalla indicando error de conexión. El usuario puede hacer un *pull to refresh* para recargar el contenido.

5.4.11. Indicador de carga

Durante la carga de contenido de las diferentes pantallas se muestra un indicador de carga. Esto es para que el usuario tenga noción de que la aplicación efectivamente está cargando el contenido.

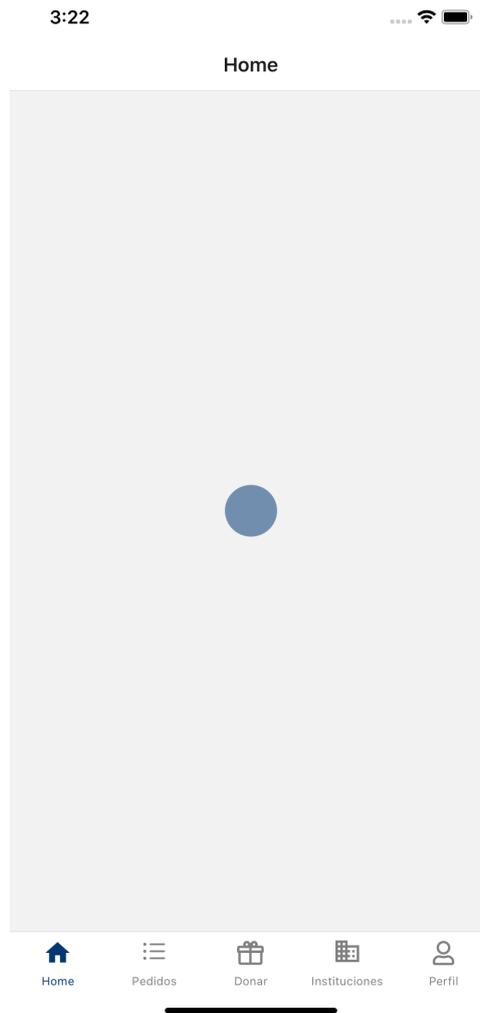


Figura 5.64: El indicador de carga consiste en un círculo que se agranda y achica mientras cambia su opacidad.

Casos de prueba

A continuación se detallan los casos de prueba realizados junto con los resultados obtenidos para cada uno de ellos.

6.1. Creación de cuenta como donante

- Objetivo: Un usuario puede ingresar a la aplicación de Ponte y crearse una cuenta como donante.
- Precondiciones: -
- Entradas: Nombre, Apellido, Dirección, Teléfono, Email y Contraseña.

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Ingresa a la aplicación	Se muestra pantalla de inicio de sesión	Pasó
2	Selecciona 'Crear una cuenta'	Se muestra el formulario de creación de cuenta	Pasó
3	No completa todos los campos y selecciona 'Crear'	Se muestra una alerta en forma de dropdown y un error por cada campo no completado	Pasó
4	Completa todos los datos pero utiliza un email ya registrado en Ponte	Se muestra una alerta en forma de dropdown indicando que el email ya está en uso	Pasó
5	Completa todos los datos correctamente	La cuenta se crea e inicia sesión de forma exitosa procediendo al <i>Home</i>	Pasó

Cuadro 6.1: Creación de cuenta como donante

6.2. Inicio de sesión

- Objetivo: Un usuario puede ingresar a la aplicación de Ponte ya teniendo una cuenta creada.
- Precondiciones: Tener una cuenta creada en Ponte.
- Entradas: Email y contraseña.

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Ingresa a la aplicación	Se muestra pantalla de inicio de sesión	Pasó
2	No completa todos los campos y selecciona 'Ingresar'.	Se muestra alerta indicando que debe ingresar todos los campos para ingresar	Pasó
3	Ingresa un email no registrado	Se muestra alerta indicando que los datos ingresados son inválidos	Pasó
4	Ingresa un email registrado con su contraseña correspondiente	Inicia sesión y se muestra la pantalla de <i>Home</i>	Pasó

Cuadro 6.2: Inicio de sesión

6.3. Visualizar los pedidos en Ponte

- Objetivo: Un usuario puede ver todos los pedidos hechos por las instituciones en Ponte que requieran donaciones.
- Precondiciones: Haber ingresado en la aplicación.
- Entradas: -

6.4. VISUALIZAR LAS INSTITUCIONES REGISTRADAS CON SU DETALLE

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación y se posiciona en la pestaña <i>Home</i>	Se muestran todos los pedidos en forma de carrusel	Pasó
2	Se posiciona en <i>Pedidos</i>	Se muestran todos los pedidos pero en forma de lista	Pasó

Cuadro 6.3: Visualizar los pedidos en Ponte

6.4. Visualizar las instituciones registradas con su detalle

- Objetivo: Un usuario puede ver todas las instituciones registradas en Ponte y acceder al detalle de cada una.
- Precondiciones: Haber ingresado en la aplicación.
- Entradas: -

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación	Se muestra la pantalla <i>Home</i>	Pasó
2	Se posiciona en <i>Instituciones</i>	Se muestran todas las instituciones en forma de lista	Pasó
3	Escribe en la barra de búsqueda una institución en particular	Se muestra el listado filtrado con la búsqueda correspondiente	Pasó
4	Ingresa a dicha institución	Se muestra una pantalla con los detalles: dirección, teléfono, email, carrusel de pedidos y ubicación en el mapa.	Pasó

Cuadro 6.4: Visualizar las instituciones registradas con su detalle

6.5. Realizar una donación a un pedido

- Objetivo: Un usuario puede realizar una donación a un pedido listado.
- Precondiciones: Haber ingresado en la aplicación.
- Entradas: Cantidad de unidades a donar, disponibilidad horaria, comentario (opcional).

6.5. REALIZAR UNA DONACIÓN A UN PEDIDO

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación	Se muestra la pantalla <i>Home</i>	Pasó
2	Elige un pedido desde el carrusel de pedidos	Se muestra el formulario para realizar la donación	Pasó
3	No completa el campo 'Cantidad'	Se muestra un error indicando que debe indicar la cantidad	Pasó
4	Ingresa una cantidad igual o menor a cero	Se muestra un error indicando que debe indicar un número mayor a cero	Pasó
5	Ingresa un número mayor a la cantidad de bienes pedidos	Se muestra un error indicando que la cantidad debe ser menor o igual a la cantidad pedida	Pasó
6	No indica ningún horario disponible	Se muestra una alerta en forma de dropdown indicando que es necesario al menos un horario	Pasó
7	Indica horarios repetidos	Se muestra una alerta en forma de dropdown indicando que hay horarios repetidos	Pasó
8	Ingresa datos correctos	Se muestra la pantalla de confirmación de donación	Pasó
9	Presiona el botón 'Listo'	Se lo redirige a la pantalla inicial del stack correspondiente (en este caso la pantalla inicial de <i>Home</i>) y aparece la donación dentro de 'Mis donaciones'	Pasó

Cuadro 6.5: Realizar una donación a un pedido

6.6. Realizar una donación sin pedido específico

- Objetivo: Un usuario puede realizar una donación que no tenga un pedido de una institución asociado.
- Precondiciones: Haber ingresado en la aplicación.
- Entradas: Bien a donar, cantidad de unidades a donar, disponibilidad horaria, comentario (opcional).

6.6. REALIZAR UNA DONACIÓN SIN PEDIDO ESPECÍFICO

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación	Se muestra la pantalla <i>Home</i>	Pasó
2	Elige la pestaña <i>Donar</i>	Se muestra el formulario para realizar una donación sin pedido	Pasó
3	No completa los campos obligatorios	Se muestra un error por cada campo obligatorio no completado	Pasó
4	No completa el campo 'Donación'	Se muestra un error indicando que debe indicar el bien a donar	Pasó
5	Ingresar una cantidad igual o menor a cero	Se muestra un error indicando que debe indicar un número mayor a cero	Pasó
6	Indica horarios repetidos	Se muestra una alerta en forma de dropdown indicando que hay horarios repetidos	Pasó
7	Ingresar datos correctos	Se muestra la pantalla de confirmación de donación	Pasó
8	Presiona el botón 'Listo'	Se lo redirige a la pantalla inicial del stack, es decir el formulario de <i>Donar</i> en limpio	Pasó
9	Se dirige a <i>Home</i>	Puede visualizar su donación con el mensaje <i>En búsqueda de institución</i>	Pasó

Cuadro 6.6: Realizar una donación sin pedido específico

6.7. Visualizar las donaciones como donante

- Objetivo: Un usuario puede ver aquellas donaciones a las que realizó una oferta con su estado.
- Precondiciones: Haber ingresado en la aplicación y realizado alguna oferta.
- Entradas: -

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación y se posiciona en la pestaña <i>Home</i>	Se muestran todas las donaciones en transcurso que involucran al donante en forma de carrusel	Pasó
2	Se posiciona en <i>Perfil</i>	Se muestran todas las donaciones en transcurso que involucran al donante en forma de lista	Pasó
3	Selecciona el icono con forma de tuerca	Se muestra la pantalla de configuración	Pasó
4	Selecciona la opción historial	Se muestra la pantalla de historial y puede visualizar aquellas donaciones ya completadas satisfactoriamente o canceladas	Pasó

Cuadro 6.7: Visualizar las donaciones como donante

6.8. Cancelación de donación

- Objetivo: Un usuario puede cancelar una donación en transcurso a la cual haya realizado una oferta.
- Precondiciones: Haber ingresado en la aplicación y tener una donación en curso.
- Entradas: -

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación y se posiciona en la pestaña <i>Home</i>	Se muestran todas las donaciones en transcurso que involucran al donante en forma de carrusel	Pasó
2	Selecciona el botón 'Cancelar' en una donación en transcurso	Se recarga la página y la donación deja de mostrarse	Pasó
3	Se dirige a <i>Perfil</i>	Se muestra la pantalla correspondiente con las donaciones en transcurso	Pasó
4	Selecciona el icono con forma de tuerca	Se muestra la pantalla de configuración	Pasó
5	Selecciona la opción historial	Se muestra la pantalla de historial y puede visualizar la donación cancelada	Pasó

Cuadro 6.8: Cancelación de donación

6.9. Geolocalizar un voluntario

- **Objetivo:** Un usuario puede geolocalizar un voluntario cuando está yendo a su domicilio o llevando la donación a la institución.
- **Precondiciones:** Haber ingresado en la aplicación y tener una donación en curso con estado BEING-LOOKED-FOR o BEING-TAKEN-TO.
- **Entradas:** -

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación y se posiciona en la pestaña <i>Home</i>	Se muestran todas las donaciones en transcurso que involucran al donante en forma de carrusel	Pasó
2	Elige una donación en estado BEING-LOOKED-FOR o BEING-TAKEN-TO y selecciona el icono de geolocalización	Se muestra la pantalla de geolocalización con el mensaje 'Aguardando ubicación'	Pasó
3	El voluntario está usando la aplicación y compartiendo su ubicación	Al donante se le muestra el mapa con la ubicación del voluntario.	Pasó
4	El dispositivo del voluntario cambia sus coordenadas	El símbolo del vehículo simbolizando al voluntario se desplaza al lugar correspondiente en el mapa que visualiza el donante.	Pasó

Cuadro 6.9: Geolocalizar un voluntario

6.10. Compartir un pedido

- Objetivo: Un usuario puede compartir un pedido listado en sus redes sociales.
- Precondiciones: Debe existir al menos un pedido en transcurso de alguna institución y el usuario debe haber ingresado en la aplicación. Para compartir un pedido en WhatsApp u otras redes sociales debe utilizar un iPhone real (no un simulador).
- Entradas: -

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación y se posiciona en la pestaña <i>Home</i>	Se muestra el listado de pedidos en forma de carrusel	Pasó
2	Elige un pedido y selecciona el botón de compartir	Se despliega el menú de compartir de iOS	Pasó
3	El usuario elige la aplicación correspondiente, en este caso WhatsApp	El menú de compartir ofrece al usuario elegir el contacto destino	Pasó
4	Elige un contacto	Se muestra el mensaje prearmado para compartir el pedido	Pasó
5	Selecciona enviar	Se envía el mensaje al usuario seleccionado	Pasó

Cuadro 6.10: Compartir un pedido

6.11. Convertirse en voluntario

- Objetivo: Un usuario donante puede convertirse en voluntario.

6.11. CONVERTIRSE EN VOLUNTARIO

- Precondiciones: Usuario registrado como donante con sesión iniciada en la aplicación.
- Entradas: Tamaño del vehículo, franja horaria y comentario (opcional).

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación y se posiciona en la pestaña <i>Home</i>	Se muestra la pantalla inicial de <i>Home</i>	Pasó
2	Se dirige a <i>Perfil</i>	Se muestra la pantalla inicial de <i>Perfil</i>	Pasó
3	Selecciona el botón 'Quiero ser voluntario'	Se lo redirige al formulario para convertirse en voluntario	Pasó
4	Completa los datos correctamente	Se muestra una alerta indicando que su solicitud fue registrada	Pasó
5	Acepta la alerta	Se lo redirige a la página inicial de <i>Perfil</i> y se muestra un mensaje indicando que la solicitud para convertirse en voluntario está en progreso	Pasó
6	Un administrador aprueba la solicitud y el usuario refresca la página de perfil	Se muestra al usuario como Voluntario permitiéndole ver sus entregas además de sus donaciones	Pasó

Cuadro 6.11: Convertirse en voluntario

6.12. Visualizar listado de entregas

- **Objetivo:** Un voluntario puede visualizar las entregas pendientes que tiene asignadas.
- **Precondiciones:** Usuario registrado como voluntario con sesión iniciada en la aplicación y con una donación asignada por un administrador.
- **Entradas:** -

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación y se posiciona en la pestaña <i>Home</i>	Se muestra la pantalla inicial de <i>Home</i> con las entregas asignadas dentro de 'Mis entregas'	Pasó
2	Se dirige a <i>Perfil</i>	Se muestra la pantalla inicial de <i>Perfil</i> donde también puede ver sus entregas pendientes	Pasó

Cuadro 6.12: Visualizar listado de entregas

6.13. Editar contraseña

- **Objetivo:** Un usuario puede cambiar su contraseña.
- **Precondiciones:** Usuario con sesión iniciada.
- **Entradas:** Contraseña actual y contraseña nueva.

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación y se posiciona en la pestaña <i>Home</i>	Se muestra la pantalla inicial de <i>Home</i>	Pasó
2	Se dirige a <i>Perfil</i>	Se muestra la pantalla inicial de <i>Perfil</i>	Pasó
3	Selecciona el icono de la tuerca de configuración	Se muestra la pantalla de configuración	Pasó
4	Selecciona la opción 'Editar contraseña'	Se muestra la pantalla de 'Editar contraseña'	Pasó
5	Indica una contraseña actual incorrecta	Se muestra una alerta en forma de dropdown indicando que la contraseña actual es incorrecta	Pasó
6	Indica la contraseña actual y la nueva de forma correcta	Se muestra una alerta indicando el cambio exitoso	Pasó
7	Selecciona continuar	Se muestra la pantalla inicial de <i>Perfil</i>	Pasó

Cuadro 6.13: Editar contraseña

6.14. Editar datos

- Objetivo: Un usuario puede editar sus datos personales (nombre, apellido, teléfono y/o dirección).
- Precondiciones: Usuario con sesión iniciada.
- Entradas: Dato o datos a cambiar (nombre, apellido, teléfono y/o dirección).

Procedimiento			
#Paso	Descripción	Resultado	Pasó/Falló
1	Abre la aplicación y se posiciona en la pestaña <i>Home</i>	Se muestra la pantalla inicial de <i>Home</i>	Pasó
2	Se dirige a <i>Perfil</i>	Se muestra la pantalla inicial de <i>Perfil</i>	Pasó
3	Selecciona el icono de la tuerca de configuración	Se muestra la pantalla de configuración	Pasó
4	Selecciona la opción 'Editar datos'	Se muestra la pantalla de 'Editar datos'	Pasó
5	Indica un teléfono con caracteres inválidos y selecciona 'Editar datos'	Se muestra un error indicando que el teléfono es inválido	Pasó
6	Indica un teléfono válido y selecciona 'Editar datos'	Se muestra una alerta indicando que los datos fueron actualizados correctamente	Pasó
7	Selecciona 'Continuar'	Se muestra la pantalla inicial de <i>Perfil</i> con el teléfono actualizado	Pasó

Cuadro 6.14: Editar datos

Conclusiones

Consideramos que fue posible lograr el objetivo que nos habíamos propuesto creando una aplicación robusta, funcional y que mantenga coherencia y consistencia con las previas dos versiones existentes así como también un componente característico a las aplicaciones iOS.

Adicionalmente, este proyecto tomó especial relevancia en el contexto actual de pandemia donde la economía se vio afectada y aquellos con necesidades se vieron en una situación aún más precaria. Es por esto que soluciones sociales como lo es Ponte se tornan cada vez más importantes y en nuestro caso particular van más allá del ámbito académico para contemplar un lanzamiento real.

Creemos que la decisión de realizarlo en React Native agilizó el desarrollo, no solo por el gran soporte en el ambiente sino también por su similitud con React, lenguaje que ambos integrantes del equipo utilizaron previamente.

A lo largo del desarrollo se fueron adquiriendo varias herramientas y cementando otras ya existentes permitiendo ver la creación de una aplicación de comienzo a fin y todo lo que eso conlleva. Se trabajaron conceptos de usabilidad, experiencia de usuario, bases de datos y arquitectura del software entre otros.

En una primer etapa, consistió en un desarrollo más bien front end, comenzando por los mock y luego la estructura inicial de la aplicación en React Native.

Luego se empezó a trabajar con las herramientas de back office, la base de datos y API presentes. Esto requirió un proceso de relevamiento ya que se trataba de código y herramientas con las que no se había trabajado. En ciertos momentos el hecho de que la API ya esté realizada de cierta forma consistió en una limitación ya que algunas ideas no se pudieron llevar a cabo. Un ejemplo de esto fue el diseño inicial de la pantalla de una institución que contaba con dos imágenes: un logo y un banner. Esta idea no se logró utilizar debido a que las tablas en la base de datos no contemplaban un campo de imagen.

Estas limitaciones de back end afortunadamente pueden ser analizadas y, si así se desea, mejoradas por el equipo que nos continúa (Ponte 4) permitiendo una constante mejora del proyecto de Ponte como un todo. El hecho de que Ponte vaya iterando, creciendo y mejorando permite simular un ambiente de trabajo con un producto que se va revisando y mejorando.

Creemos que el agregado de la versión móvil iOS a este proyecto que es Ponte trae muchos beneficios y aumenta considerablemente el público al que se puede alcanzar, por lo que consideramos satisfactorio el trabajo realizado.

Trabajo futuro

Es importante saber qué mejoras se le pueden realizar al proyecto Ponte para que en el futuro pueda lanzarse al público evitando posibles inconvenientes.

Muchas de las limitaciones que encontramos en Ponte iOS parten de desperfectos en la API, pero dicha tarea está siendo llevada a cabo por Bianca Ritorto y Clara Guzzetti, quienes están a cargo de Ponte 4. Una vez concluido dicho proyecto, va a ser necesario retocar código en ambas aplicaciones móviles (tanto Ponte iOS como Android) teniendo en cuenta que probablemente cambie la forma de realizar algunos pedidos a la API.

Asimismo, ya que Ponte iOS incorpora funcionalidades adicionales, sería imprescindible incorporar dichas funcionalidades a Ponte Android y Web para que ninguna plataforma tenga más limitaciones que otra.

Con respecto al estado actual de la API, las mejoras que podrían realizarse son varias. En primer lugar, la API no ofrece ningún tipo de paginación provocando que listados grandes de datos sean cargados completamente de una. A mayor cantidad de datos, esto produciría una carga muy lenta de páginas como Pedidos, Home o Instituciones. Dichas páginas utilizan en Ponte iOS componentes como FlatList, haciendo que una vez incorporada la paginación, el cambio en la API será muy simple de implementar en la aplicación. También sería necesario mejorar el filtrado

de algunos pedidos a la API. Ocurre que por ejemplo la API no permite filtrar los pedidos hechos por una institución en específico. Por otro lado tampoco devuelve los datos de forma ordenada. Ambas cosas producen sobrecarga desde el lado del frontend, provocando demoras en el usuario final. Adicionalmente, se puede considerar la creación de endpoints más específicos utilizando el estilo REST que permitirían expandir las funcionalidades tanto de la versión web como las versiones móviles.

Respecto a la base de datos, como se mencionó en las conclusiones, se podría considerar de agregar campos de imágenes tanto para el perfil del usuario como para el de instituciones para permitir una personalización mayor.

Otro aspecto relevante a considerar son las notificaciones. Actualmente, el servidor no lo contempla, pero en el futuro se podrían incorporar notificaciones al mismo, por ejemplo, cada vez que haya una actualización de estado de donación donde se vea involucrado el usuario. Desde el lado de la geolocalización, si se hiciera uso de la versión paga de PubNub también podrían incorporarse notificaciones en base a la ubicación del voluntario.

Una funcionalidad que no se encuentra en ninguna de las plataformas de Ponte, es la de recuperar contraseña en caso de haberla olvidado. Dicha funcionalidad también está siendo llevada a cabo por Ponte 4.

Adicionalmente, como se mencionó previamente si se lanza la aplicación iOS al Apple Store se podrían agregar links dinámicos dentro de la funcionalidad de compartir. Esto permitiría que se redirija a la aplicación, si ya se cuenta con la misma, o a la tienda para descargarla en caso contrario.

Desde el lado de la seguridad también hay que notar que actualmente los endpoints de la API son públicos. Esto último cambiará también con la realización de Ponte 4 que incorporará mejoras en la seguridad del proyecto.

Por último, sería un cambio beneficioso, delegar menor cantidad de ope-

raciones en el backoffice (Forest). Por ejemplo, sería útil que la asignación de un voluntario a una donación sea automática teniendo en cuenta la disponibilidad horaria tanto del donante como del propio voluntario.

Bibliografía

- [blogs.adobe.com](https://blogs.adobe.com/creativecloud/mobile-navigation-tab-bar-and-pictorial-icons/). 2020. Mobile navigation tab bar - Adobe. Disponible en: <https://blogs.adobe.com/creativecloud/mobile-navigation-tab-bar-and-pictorial-icons/> [Visitado el 20 de abril 2020].
- [Developer.apple.com](https://developer.apple.com/design/human-interface-guidelines/ios/bars/navigation-bars/). 2020. Navigation Bars - Bars - Ios - Human Interface Guidelines - Apple Developer. Disponible en: <https://developer.apple.com/design/human-interface-guidelines/ios/bars/navigation-bars/> [Visitado el 20 de abril 2020].
- [Developer.apple.com](https://developer.apple.com/design/human-interface-guidelines/ios/bars/search-bars/). 2020. Search Bars - Bars - Ios - Human Interface Guidelines - Apple Developer. Disponible en: <https://developer.apple.com/design/human-interface-guidelines/ios/bars/search-bars/> [Visitado el 20 de abril 2020].
- [Developer.apple.com](https://developer.apple.com/design/human-interface-guidelines/ios/bars/tab-bars/). 2020. Tab Bars - Bars - Ios - Human Interface Guidelines - Apple Developer. Disponible en: <https://developer.apple.com/design/human-interface-guidelines/ios/bars/tab-bars/> [Visitado el 20 de abril de 2020].

Referencias

- Repositorio del proyecto: <https://bitbucket.org/itba/ponte-ios/src/master/>