



# Guía de Pantallas - Napphy Services

Esta guía describe todas las pantallas de la aplicación, su función, componentes y flujos de navegación.



## Índice

1. [Pantallas de Autenticación](#)
2. [Pantallas de Niñeras](#)
3. [Pantallas de Padres](#)
4. [Pantallas de Administrador](#)
5. [Pantallas de Chat](#)
6. [Pantallas Comunes](#)

## Pantallas de Autenticación

### 1. Splash Screen 🌟

**Ruta:** /

**Archivo:** `lib/screens/common/splash_screen.dart`

**Descripción:**

Pantalla inicial que se muestra al abrir la aplicación. Verifica el estado de autenticación y redirige al usuario a la pantalla correspondiente.

**Componentes:**

- Logo de la aplicación
- Nombre de la aplicación
- Indicador de carga
- Gradiente de fondo (primary → secondary)

**Flujo:**

```
App inicia
  ↓
Muestra Splash Screen (2 segundos)
  ↓
Verifica autenticación
  ↓
¿Usuario autenticado?
├─ Sí → Redirige según rol
│   ├── Niñera → Nanny Home
│   ├── Padre → Parent Home
│   └── Admin → Admin Dashboard
└─ No → Login Screen
```

## 2. Login Screen

**Ruta:** /login

**Archivo:** lib/screens/auth/login\_screen.dart

### Descripción:

Pantalla de inicio de sesión donde los usuarios ingresan sus credenciales.

### Campos del Formulario:

- Email (validación de formato)
- Contraseña (validación de longitud mínima)

### Componentes:

- Logo de la aplicación
- Campos de texto para email y contraseña
- Botón "Iniciar sesión"
- Link "¿Olvidaste tu contraseña?"
- Link "Regístrate" (redirige a Role Selection)
- Indicador de carga durante la autenticación

### Validaciones:

- Email no vacío y formato válido
- Contraseña mínimo 6 caracteres

### Flujo de Éxito:

```
Usuario ingresa credenciales
↓
Click en "Iniciar sesión"
↓
Validación de formulario
↓
AuthService.signInWithEmail()
↓
Firebase Authentication
↓
Carga datos del usuario desde Firestore
↓
Redirige según rol del usuario
```

### Manejo de Errores:

- Email no encontrado
- Contraseña incorrecta
- Demasiados intentos fallidos
- Problemas de conexión

---

## 3. Role Selection Screen

**Ruta:** /role-selection

**Archivo:** lib/screens/auth/role\_selection\_screen.dart

### Descripción:

Pantalla donde el usuario selecciona su rol antes de registrarse (Niñera o Padre).

**Componentes:**

- Título “¿Cómo deseas usar Napphy?”
- Card para “Soy Niñera/Cuidador”
- Icono: child\_care
- Descripción: “Quiero ofrecer mis servicios de cuidado”
- Card para “Soy Padre/Tutor”
- Icono: family\_restroom
- Descripción: “Busco contratar servicios de cuidado”
- Link “¿Ya tienes cuenta? Inicia sesión”

**Flujo:**

```

Usuario sin cuenta accede
↓
Selecciona rol (Niñera o Padre)
↓
Redirige a Register Screen
  (pasando el rol como argumento)

```

## 4. Register Screen

**Ruta:** /register**Archivo:** lib/screens/auth/register\_screen.dart**Descripción:**

Pantalla de registro que adapta sus campos según el rol seleccionado.

**Campos Comunes:**

- Nombre completo
- Email
- Contraseña
- Confirmar contraseña

**Campos Adicionales para Niñeras:**

- Edad (validación: mayor de 18)
- Dirección
- Tarifa por hora
- Biografía (opcional)

**Componentes:**

- Formulario adaptativo según rol
- Botón “Registrarse”
- Link “¿Ya tienes cuenta? Inicia sesión”
- Indicador de carga

**Validaciones:**

- Nombre no vacío
- Email válido y único
- Contraseñas coinciden
- Contraseña mínimo 6 caracteres

- Edad mayor a 18 (niñeras)
- Tarifa positiva (niñeras)

#### Flujo de Registro:

```

Usuario completa formulario
↓
Click en "Registrarse"
↓
Validación local
↓
AuthService.registerWithEmail()
↓
Firebase crea usuario
↓
Crea documento en /users/
↓
Si es niñera:
  [ ] Crea documento en /nannies/
  [ ] Estado: isApproved = false
Si es padre:
  [ ] Crea documento en /parents/
↓
Redirige a Home correspondiente

```

## Pantallas de Niñeras

### 5. Nanny Home Screen 🏠

**Ruta:** /nanny/home

**Archivo:** lib/screens/nanny/nanny\_home\_screen.dart

#### Descripción:

Dashboard principal para niñeras con dos pestañas: Contrataciones y Perfil.

#### Tab 1: Contrataciones

Muestra lista de contrataciones con diferentes estados:



#### Estados de Contratación:

- **Pendiente:** Esperando aceptación/rechazo
- **Aceptada:** Confirmada por la niñera
- **En Progreso:** Servicio en curso
- **Completada:** Servicio finalizado
- **Rechazada/Cancelada:** No se realizó

#### Información de cada Booking:

- Fecha del servicio
- Horario (inicio - fin)
- Dirección
- Monto total
- Instrucciones especiales
- Estado actual
- Botones de acción (si está pendiente)

**Acciones en Bookings Pendientes:**

-  Aceptar: Cambia estado a "accepted"
-  Rechazar: Cambia estado a "rejected"

**Tab 2: Perfil**

Lista de opciones:

- Editar perfil
- Configuración
- Ayuda y soporte
- Cerrar sesión

**AppBar:**

- Título: "Napphy - Niñera"
- Icono de notificaciones
- Icono de chat

**Bottom Navigation:**

- Contrataciones
- Perfil

## 6. Nanny Profile Screen

**Ruta:** `/nanny/profile`

**Archivo:** `lib/screens/nanny/nanny_profile_screen.dart`

**Descripción:**

Pantalla donde las niñeras pueden ver y editar su perfil profesional.

**Componentes:****1. Avatar Circular**

- Foto de perfil (si existe)
- Botón de edición (cámara)

**1. Información Básica** (read-only)

- Nombre completo
- Email

**2. Estado de Aprobación**

- Banner naranja si está pendiente de aprobación

**3. Switch de Disponibilidad**

- Disponible / No disponible
- Actualiza en tiempo real

**4. Estadísticas**

- Rating promedio (estrellas)
- Número de reseñas

**5. Campos Editables**

- Biografía (multilinea)
- Dirección

- Tarifa por hora
- Años de experiencia

#### Flujo de Actualización:

```

Niñera edita campos
↓
Click en "Guardar cambios"
↓
Validación del formulario
↓
FirestoreService.updateNannyProfile()
↓
Actualiza en Firestore
↓
Muestra mensaje de éxito
↓
Vuelve a pantalla anterior

```

## Pantallas de Padres

### 7. Parent Home Screen 🏠

**Ruta:** /parent/home

**Archivo:** lib/screens/parent/parent\_home\_screen.dart

#### Descripción:

Dashboard principal para padres con tres pestañas.

#### Tab 1: Inicio

- Banner Principal**
  - Mensaje de bienvenida
  - Descripción del servicio
  - Botón "Buscar Niñeras"
- Grid de Servicios (2x2)**
  - Cuidado infantil
  - Ayuda con tareas
  - Primeros auxilios
  - Cuidado flexible
- Sección "¿Cómo funciona?"**
  - Paso 1: Busca
  - Paso 2: Conecta
  - Paso 3: Contrata

#### Tab 2: Mis Reservas

Lista de contrataciones del padre:

- Fecha del servicio
- Horario
- Monto total

- Estado actual
- Click para ver detalles

### Tab 3: Perfil

Opciones:

- Configuración
- Ayuda y soporte
- Cerrar sesión

### Bottom Navigation:

- Inicio
- Mis Reservas
- Perfil

## 8. Search Nannies Screen

**Ruta:** `/parent/search`

**Archivo:** `lib/screens/parent/search_nannies_screen.dart`

### Descripción:

Pantalla de búsqueda de niñeras con filtros avanzados.

### Filtros Disponibles (Panel desplegable):

- Tarifa máxima por hora (slider: \$0 - \$200)
- Calificación mínima (slider: 0 - 5 estrellas)

### Lista de Resultados:

Cada card muestra:

- Avatar/foto de perfil
- ID de niñera (primeros 6 caracteres)
- Badge "Disponible" (si aplica)
- Rating (estrellas + número de reseñas)
- Años de experiencia
- Dirección
- Tarifa por hora (destacada)
- Botón "Ver perfil"

### Ordenamiento:

- Por defecto: Mayor a menor rating

### Flujo:

```

Padre accede a búsqueda
↓
Ajusta filtros (opcional)
↓
Ve lista de niñeras aprobadas y disponibles
↓
Click en una niñera
↓
Navega a Nanny Detail Screen
  
```

---

## 9. Nanny Detail Screen

**Ruta:** `/parent/nanny-detail`

**Archivo:** `lib/screens/parent/nanny_detail_screen.dart`

### Descripción:

Perfil completo de una niñera con opción de contratar o enviar mensaje.

### AppBar Expandible:

- Foto de perfil grande
- Nombre de la niñera
- Gradiente de fondo

### Sección de Estadísticas:

- Rating (con ícono de estrella)
- Años de experiencia
- Número de reseñas

### Sección de Tarifa:

- Precio por hora (destacado)

### Sobre mí:

- Biografía de la niñera

### Certificaciones (si hay):

- Lista con checkmarks verdes
- Nombres de certificaciones

### Idiomas:

- Chips con los idiomas que habla

### Reseñas:

- Muestra hasta 3 reseñas recientes
- Rating con estrellas
- Comentario de cada reseña

### Bottom Sheet:

- Botón "Mensaje" (outline)
  - Botón "Contratar" (filled)
- 

## 10. Booking Form (Modal)

**Ubicación:** Parte de `nanny_detail_screen.dart`

### Descripción:

Formulario modal para crear una solicitud de contratación.

### Campos:

1. **Fecha del servicio**



- Date picker
- Validación: no puede ser en el pasado

#### 1. Hora de inicio

- Time picker

#### 2. Hora de fin

- Time picker
- Validación: debe ser después de hora de inicio

#### 3. Dirección

- Campo de texto
- Validación: no vacío

#### 4. Instrucciones especiales (opcional)

- Campo multilinea
- Placeholder: "Alergias, rutinas, etc."

#### Resumen:

- Horas totales (calculado)
- Tarifa por hora
- Total a pagar

#### Flujo de Contratación:

```

Padre completa formulario
↓
Click en "Enviar solicitud"
↓
Validación local
↓
Calcula horas y monto total
↓
FirestoreService.createBooking()
↓
Crea documento en /bookings/
(estado: pending)
↓
Cierra modal
↓
Muestra mensaje de éxito

```

## Pantallas de Administrador

### 11. Admin Dashboard Screen

**Ruta:** /admin/dashboard

**Archivo:** lib/screens/admin/admin\_dashboard\_screen.dart

#### Descripción:

Panel administrativo con tres pestañas para gestión completa.

#### Tab 1: Dashboard

**Grid de Estadísticas (2x2):**

- 👥 Usuarios Totales: 150
- 🧑 Niñeras Activas: 45
- 👨 Padres: 105
- ⌚ Pendientes: 8

**Actividad Reciente:**

Lista de eventos:

- Nueva niñera registrada
- Contratación completada
- Reporte recibido

**Tab 2: Pendientes**

Lista de niñeras pendientes de aprobación:

Cada card muestra:

- Avatar
- ID de niñera
- Edad
- Tarifa por hora
- Biografía
- Dirección
- Certificaciones (si hay)

**Acciones:**

- ✅ **Aprobar:**
  - Actualiza isApproved = true
  - La niñera aparece en búsquedas
- ❌ **Rechazar:**
  - Muestra diálogo de confirmación
  - (Funcionalidad para implementar)

**Tab 3: Usuarios**

Placeholder para gestión de usuarios:

- Lista de todos los usuarios
- Opciones de edición/suspensión
- (Por implementar completamente)

**Bottom Navigation:**

- Dashboard
- Pendientes
- Usuarios

## Pantallas de Chat

### 12. Chat List Screen

**Ruta:** /chat/list

**Archivo:** lib/screens/chat/chat\_list\_screen.dart

**Descripción:**

Lista de todas las conversaciones del usuario.

**Cada Conversación Muestra:**

- Avatar del otro usuario
- Nombre del otro usuario
- Último mensaje (preview)
- Hora del último mensaje
- Hoy: HH:mm
- Ayer: "Ayer"
- Esta semana: día de la semana
- Más antiguo: DD/MM/YY
- Badge con número de mensajes no leídos (si hay)

**Ordenamiento:**

- Por fecha del último mensaje (más reciente primero)

**Estado Vacío:**

- Icono de chat
- "No tienes conversaciones"

**Flujo:**

```

Usuario accede a chats
↓
ChatService.getUserChats() (stream)
↓
Muestra lista de conversaciones
↓
Click en una conversación
↓
Navega a Chat Screen

```

## 13. Chat Screen

**Ruta:** /chat

**Archivo:** lib/screens/chat/chat\_screen.dart

**Descripción:**

Conversación individual con mensajería en tiempo real.

**AppBar:**

- Nombre del otro usuario
- Botón de información (opcional)

**Área de Mensajes:**

- Scroll invertido (nuevos abajo)
- Burbujas de mensajes:
- Mensajes propios: Alineados a la derecha, color primary
- Mensajes recibidos: Alineados a la izquierda, color gris
- Cada mensaje muestra:

- Contenido del mensaje
- Hora de envío (HH:mm)

#### Input de Mensaje:

- Botón “+” (adjuntar - no implementado)
- Campo de texto expandible
- Botón de envío (ícono de avión)

#### Funcionalidades:

- Envío de mensajes en tiempo real
- Actualización automática de nuevos mensajes
- Marcado automático como leído al visualizar
- Scroll automático a nuevo mensaje

#### Flujo de Envío:

```

Usuario escribe mensaje
↓
Click en botón de enviar
↓
ChatService.sendMessage()
↓
Añade mensaje a Firestore
↓
Actualiza lastMessage del chat
↓
Incrementa unreadCount del receptor
↓
Mensaje aparece instantáneamente
↓
Limpia campo de texto

```

## Pantallas Comunes

### 14. Notifications Screen






**Ruta:** /notifications

**Archivo:** lib/screens/common/notifications\_screen.dart

#### Descripción:

Lista de todas las notificaciones del usuario.

#### Tipos de Notificaciones:

-  Nueva solicitud de contratación
-  Mensaje nuevo
-  Reserva confirmada
-  Nueva reseña
-  Alertas del sistema

#### Cada Notificación Muestra:

- Icono según tipo
- Título

- Mensaje descriptivo
- Tiempo relativo ("2 horas atrás", "Ayer")

**AppBar:**

- Título: "Notificaciones"
- Botón: "Marcar todas como leídas"

**Estado Vacío:**

- Icono de campana
- "No tienes notificaciones"

**Nota:** Datos de ejemplo en la versión actual. Debe conectarse con Firebase Messaging en producción.

---

## 15. Settings Screen

**Ruta:** `/settings`

**Archivo:** `lib/screens/common/settings_screen.dart`

**Descripción:**

Pantalla de configuración con múltiples opciones organizadas por secciones.

**Sección: Notificaciones**

- Switch: Habilitar notificaciones
- Switch: Notificaciones por email
- Switch: Notificaciones push

**Sección: Apariencia**

- Switch: Modo oscuro (funcionalidad pendiente)

**Sección: Cuenta**

- Editar perfil
- Cambiar contraseña
- Privacidad

**Sección: Soporte**

- Ayuda (modal con contactos)
- Acerca de (modal con info de la app)
- Enviar comentarios (modal con formulario)

**Versión:**

- Muestra versión de la app en la parte inferior
- 



## Componentes Reutilizables

---

Aunque no son pantallas completas, estos componentes se usan en múltiples lugares:

### BookingCard

- Card que muestra información de una contratación
- Usado en: NannyHomeScreen, ParentHomeScreen
- Props: booking (BookingModel)

## **\_NannyCard**

- Card que muestra preview de una niñera
- Usado en: SearchNanniesScreen
- Props: nanny (NannyModel)

## **\_StatItem**

- Widget para mostrar estadísticas con ícono
- Usado en: NannyDetailScreen, AdminDashboardScreen
- Props: icon, value, label, color

## **\_MessageBubble**

- Burbuja de mensaje en el chat
- Usado en: ChatScreen
- Props: message (MessageModel), isMe (bool)



## **Flujos de Navegación Principales**

### **Flujo de Contratación Completo**

```

Padre busca niñeras
↓
SearchNanniesScreen
↓
Click en niñera
↓
NannyDetailScreen
↓
Click en "Contratar"
↓
BookingForm (modal)
↓
Completa y envía formulario
↓
Booking creado (estado: pending)
↓
Niñera recibe notificación
↓
Niñera ve en NannyHomeScreen
↓
Acepta o rechaza
↓
Estado actualizado
↓
Padre recibe notificación
↓
Ambos ven el booking actualizado
  
```

## Flujo de Chat

```

Usuario A quiere chatear con Usuario B
↓
Click en botón "Mensaje"
↓
ChatService.getOrCreateChat()
↓
Busca chat existente o crea nuevo
↓
Navega a ChatScreen
↓
Usuario A envía mensaje
↓
Mensaje guardado en Firestore
↓
Usuario B ve notificación
↓
Usuario B abre chat
↓
Mensajes marcados como leídos
↓
Conversación en tiempo real

```



## Diseño y UX

### Colores Principales

- Primary: #6B4EFF (Morado)
- Secondary: #FF6B9D (Rosa)
- Accent: #FFC107 (Amarillo)
- Background: #F8F9FA (Gris claro)
- Text: #2D3748 (Gris oscuro)

### Tipografía

- Font Family: Poppins
- Tamaños:
  - H1: 32px (títulos principales)
  - H2: 24px (títulos de sección)
  - H3: 20px (subtítulos)
  - Body: 16px (texto normal)
  - Caption: 14px (texto secundario)
  - Small: 12px (texto pequeño)

### Espaciado

- Padding de pantallas: 16px
- Espaciado entre elementos: 8px, 16px, 24px
- Border radius: 12px (botones, cards)
- Border radius grande: 16px (cards destacados)

## Iconografía

- Tamaño estándar: 24px
- Tamaño grande: 40px, 80px
- Material Icons

---

Esta guía proporciona una visión completa de todas las pantallas de la aplicación y cómo interactúan entre sí. 🎨