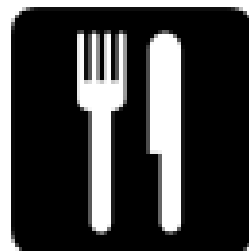


Where2Eat



Aplicación: Where2Eat

Motivación: Brindar un servicio de búsqueda de restaurants por nombre y tipo de comida cercanos a la posición del usuario.

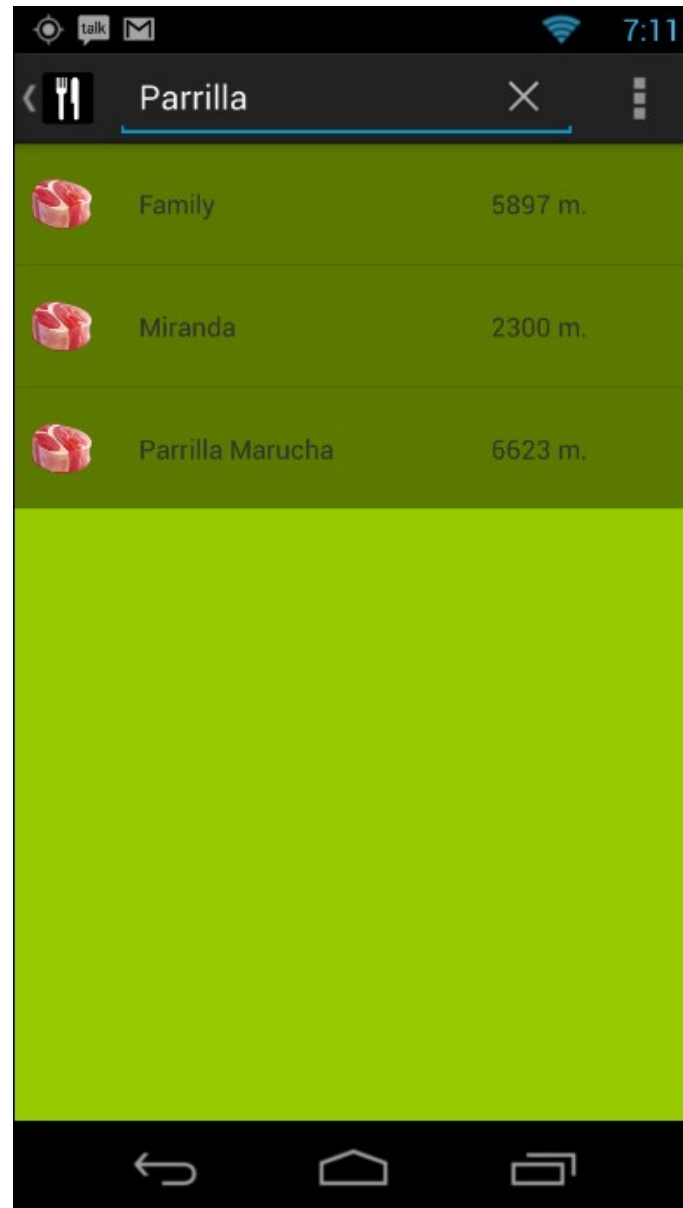
Contexto: Una aplicación móvil que corra en dispositivos Android.

Público objetivo: Usuarios con smatphones que suelen salir a comer frecuentemente.

Caso de uso típico: búsqueda de restaurant por tipo de comida

- El usuario escribe en la barra de búsqueda el nombre de un restaurant o un tipo de comida.
- La aplicación devuelve una lista de restaurants ordenados por cercanía a la ubicación del usuario.

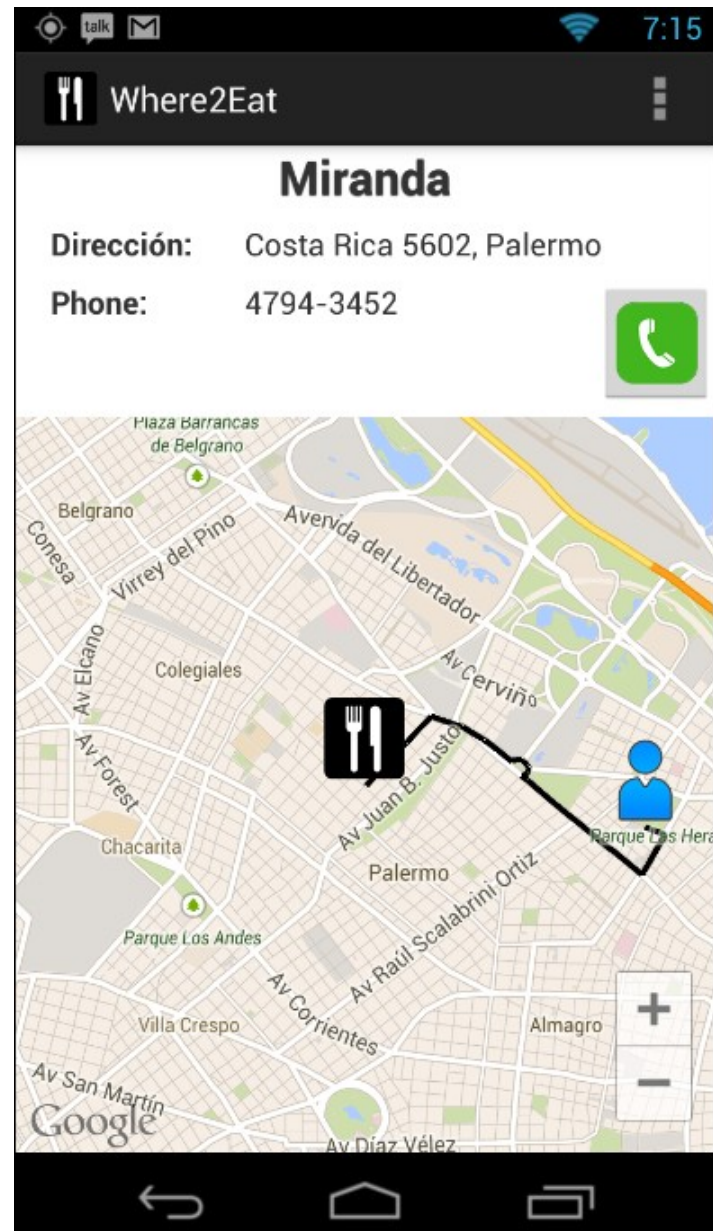
Vista resultado de búsqueda



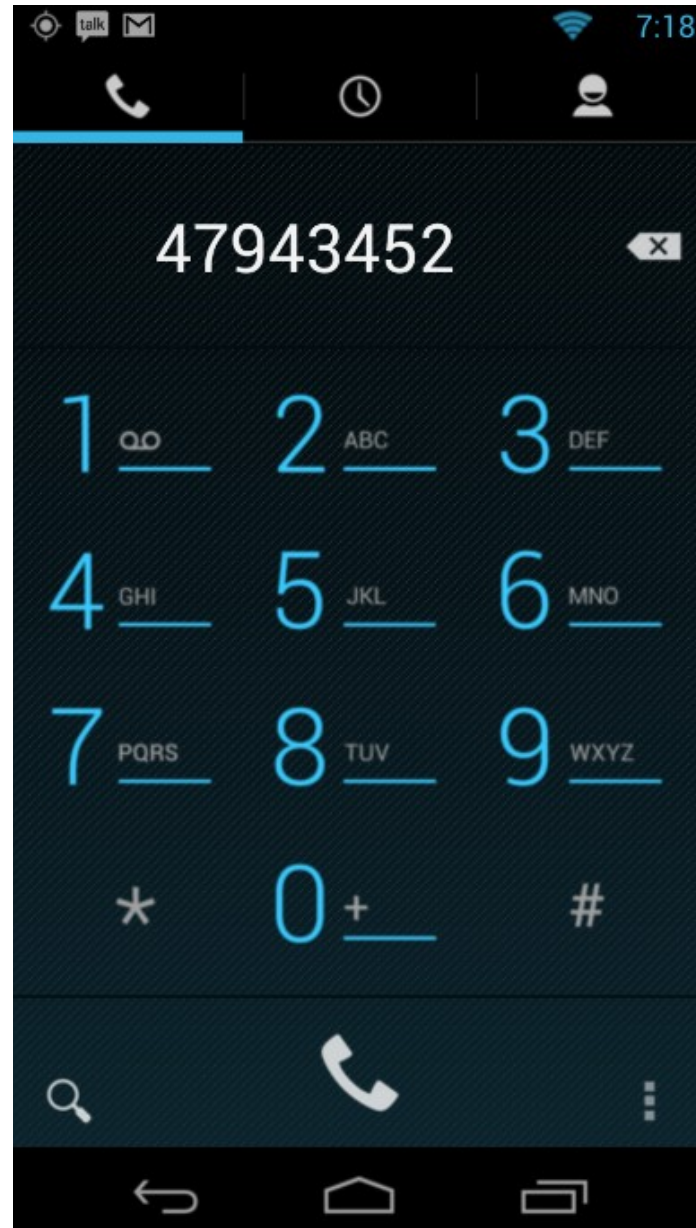
Caso de uso típico: selección de restaurant

- El usuario selecciona uno de los restaurants que aparecen en la lista.
- La aplicación muestra un detalle del restaurant, su ubicación en el mapa junto a la ubicación actual del usuario y una ruta posible que le indique como llegar en auto.

Vista restaurant en el mapa.



Vista de la llamada



Caso de uso típico: agenda y mail

- El usuario, una vez ha elegido y ubicado el restaurant puede crear una nueva cita en la agenda para ir al mismo.
- El usuario también puede enviar un mail con el nombre y la dirección del restaurant seleccionado.

Vista de agenda y mail

8:58

CANCEL DONE

MartinCalendar
martincammi@gmail.com

Miranda

Costa Rica 5602, Palermo

FROM
Sun, Oct 6, 2013 9:00PM

TO
Sun, Oct 6, 2013 10:00PM

☐ ALL DAY

(GMT-3:00) Argentina Time

GUESTS
Guests

Tel: 4794-3452

REPETITION
One-time event

9:03

Compose

martincammi@gmail.com

To

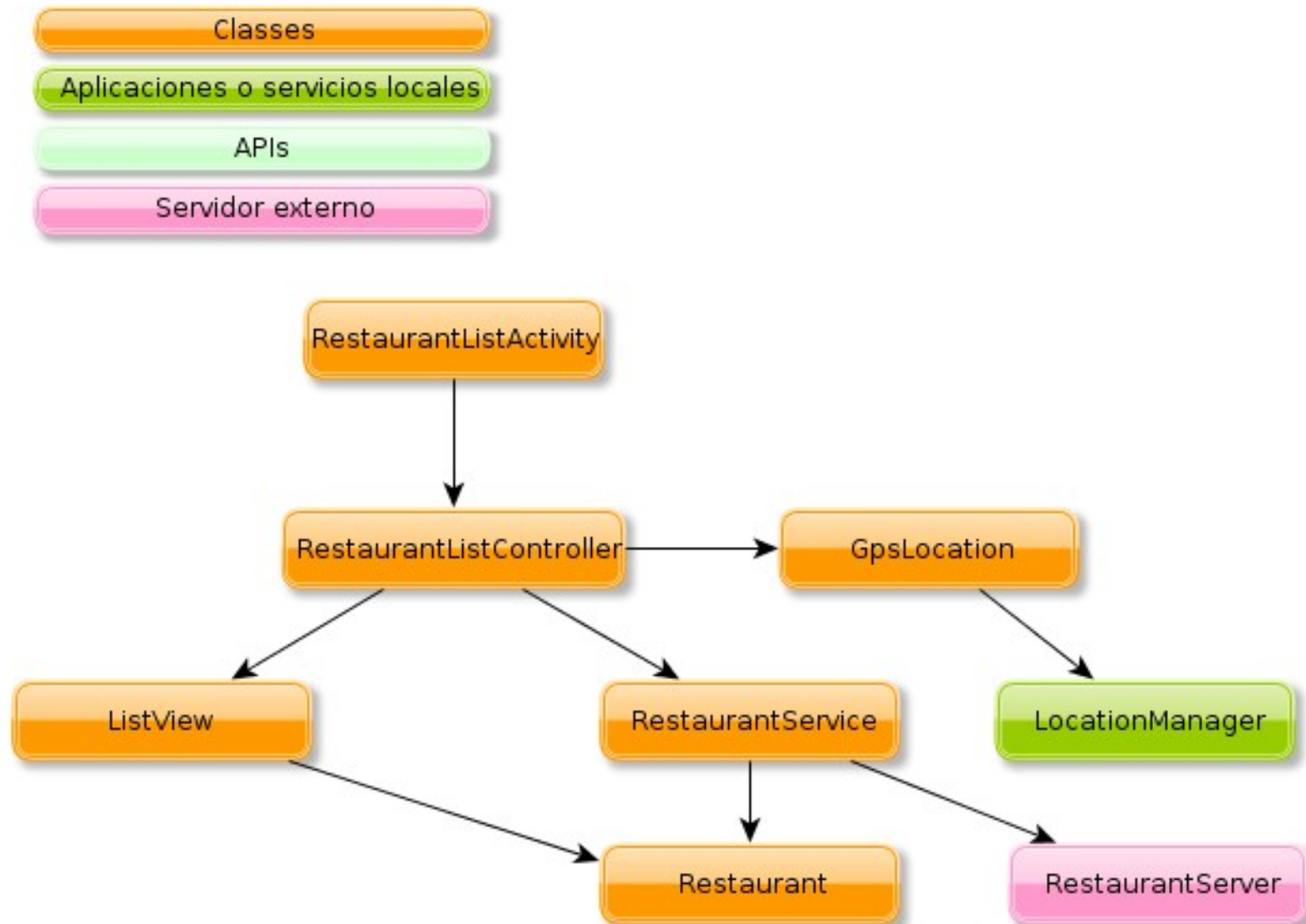
Reserva en Miranda

Reserve en el restaurant Miranda, que queda en Costa Rica 5602, Palermo

Arquitectura básica parte1

- La arquitectura sigue el modelo MVC para dispositivos Android donde existen Activities que se comunican con Controllers que finalmente interactúan con los elementos de la Vista y del Modelo de negocio.
- La entidad de negocio principal es el Restaurant que agrupa todos sus datos relevantes.
- Una entidad RestaurantService permite filtrar los Restaurants de interés, según diferentes criterios.
- El RestaurantService interactúa con el GpsLocation para conocer las distancias entre los Restaurantes y la posición actual del usuario.
- El RestaurantService también interactúa con un RestaurantServer para la obtención remota de los locales de comida.

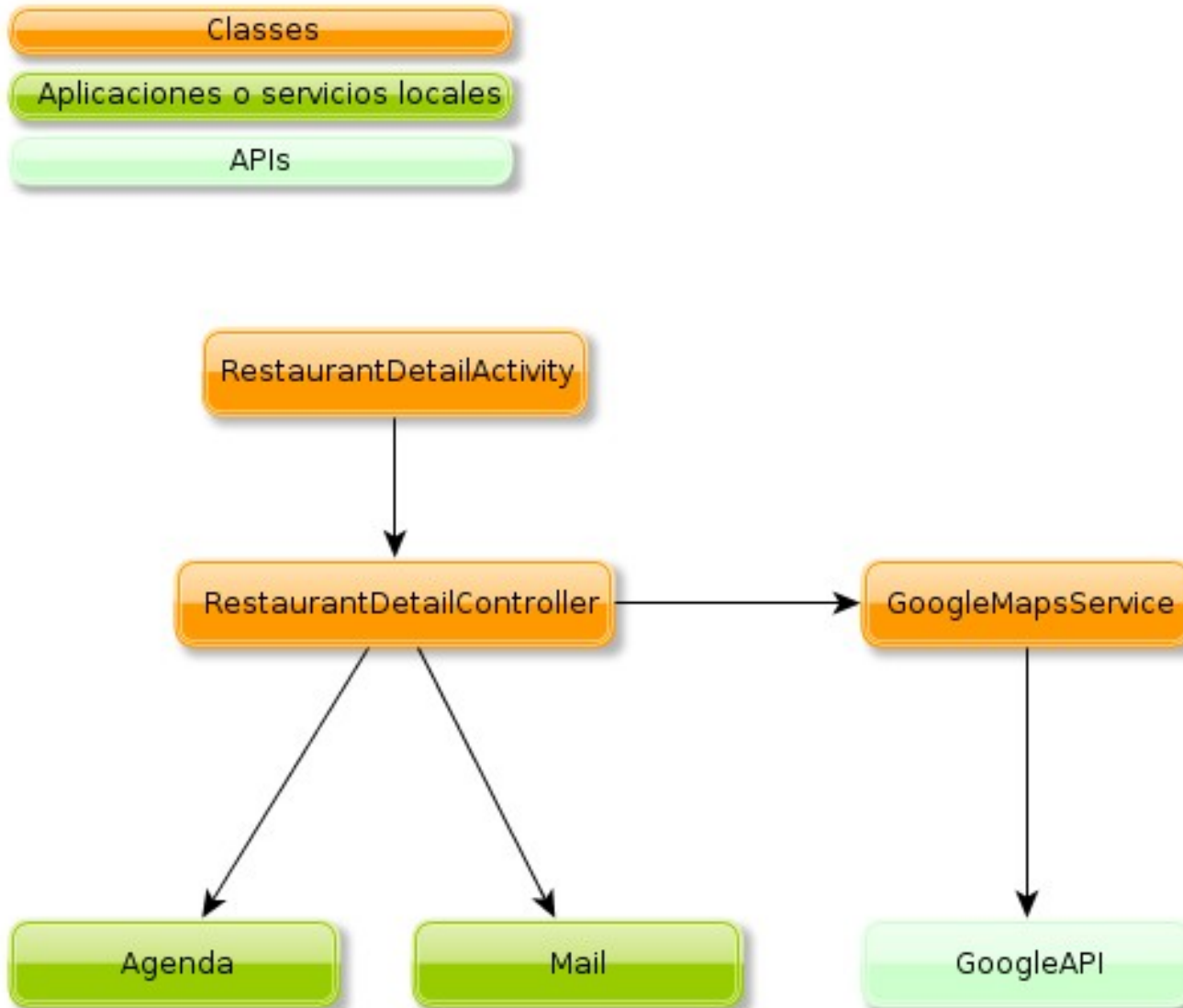
Arquitectura básica parte1



Arquitectura básica parte2

- Al momento de ubicar el restaurant en el mapa, se utiliza la entidad GoogleMapsService que interactúa con la API de Google.
- Alternativamente es posible programar una cita o enviar un mail con información del restaurante ubicado, esto se realiza interactuando con las aplicaciones locales de Agenda y Mail.

Arquitectura básica parte2



Funcionalidades adicionales

- **Geolocalización:** A través del LocationManager se agregó la funcionalidad de obtener la ubicación actual del usuario mediante el GPS.
- **Mapa de Google:**
 - **Integración el Mapa en la aplicación:** El mapa de google se muestra integrado en la aplicación utilizando un FragmentActivity.
 - **Ubicación en el Mapa:** Se cuenta con la ubicación de los restaurantes en el mapa.
 - **Distancia en el Mapa:** Se muestra la distancia de cada restaurant con respecto a la posición actual del usuario.
 - **Dibujo de la ruta:** Se dibuja la ruta desde la posición actual al restaurant ubicado en el mapa.
- **Obtención de ruta desde un Servidor**
 - Para el dibujo de la ruta se interactúa mediante HttpRequest con un servidor externo de Google (<http://maps.googleapis.com/maps/api/directions/json>) se parsea una respuesta de JSON y se obtiene la ruta.
- **Llamada**
 - Al momento de ver el detalle de un Restaurant se agregó la posibilidad de llamar por teléfono al Restaurant.