

# MensajerO

Taller de programación 2 - Grupo 3

# Metodología de trabajo

- Approach “ágil”
  - Iteraciones definidas por los checkpoints (3 en total)
  - Armado de backlog
  - Priorización de tareas y definición de alcances de cada iteración
- Equipo de pares
- Fluída comunicación
- Reuniones de trabajo

# Herramientas utilizadas

- Versionado: Git
  - 2 repositorios:
    - [http://github.com/mcapristo/taller\\_de\\_prog\\_2](http://github.com/mcapristo/taller_de_prog_2)
    - [http://github.com/mcapristo/taller\\_de\\_prog\\_2\\_android](http://github.com/mcapristo/taller_de_prog_2_android)
- Rest API:
  - C++
  - Eclipse
  - Librerías indicadas por la cátedra

# Herramientas utilizadas (cont.)

- Aplicación móvil:
  - Java, android sdk
  - Android Studio
  - Librerías: Voley, Gson
- Herramientas de tracking de req. y bugs:
  - Arrancamos con Trello...
  - Luego migramos a Github

# Cronograma

- 1er Iteración
  - Investigación y armado de entornos
  - Definición e implementación de rest API.
  - Documentación de instalación del server y dependencias
  - Implementación del server.

# Cronograma

- 2da Iteración
  - Funcionalidad básica aplicación móvil (se exceptúan incluir check in y manejo de fotos de perfil)
  - Corrección de bugs en servidor y cliente

# Cronograma

- 3ra Iteración
  - Features de check in y manejo de foto de perfil
  - Corrección de bugs en servidor y cliente
  - Documentación en formato solicitado

# Pruebas

## Tests de código:

- Servidor:
  - Librerías: gtest, gcov.
  - Mayor importancia a estos tests automatizados, dado que no se realizan tests de usuario en este componente del sistema.



# Pruebas

- Aplicación móvil:
  - Librerías: JUnit, Android testing framework.

## Tests de usuario (de aplicación móvil):

- Instalación en distintos dispositivos
- Usuarios finales (sin conocimiento técnico y sin estar involucrados en el proyecto)

# Pruebas

## Tests de usuario (cont.):

- Funcionalidades testeadas a destacar:
  - Envío de emoticones.
  - Rotación y uso de la aplicación con el celular en horizontal.
  - Registración simultánea de usuarios.
  - Uso en simultáneo del envío de chats comunes como chats por difusión.

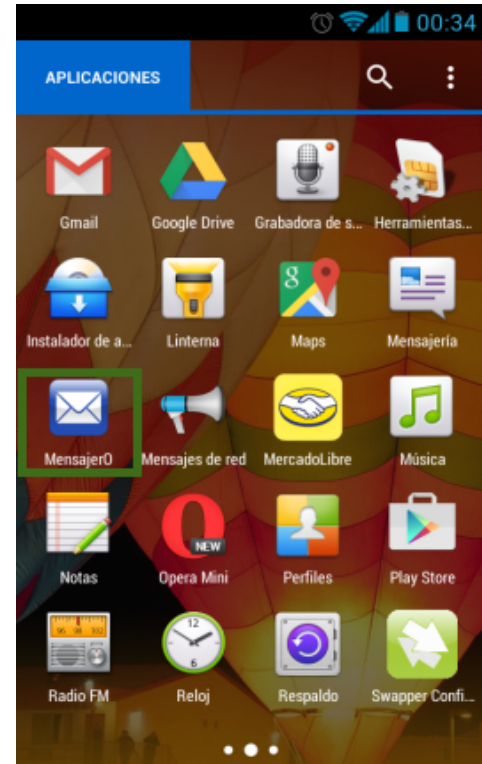
# Aplicación - usabilidad

Aplicación realizada de manera que sea:

- Fácil de comprender y utilizar
- Intuitiva para los usuarios
- Que facilite las acciones a los usuarios

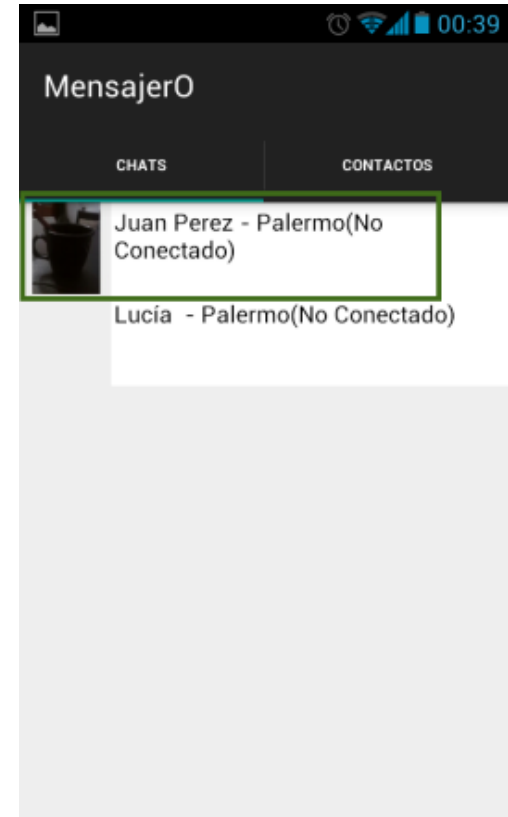
# Aplicación - usabilidad

Logo de la aplicación para evitar la imagen por default de android.



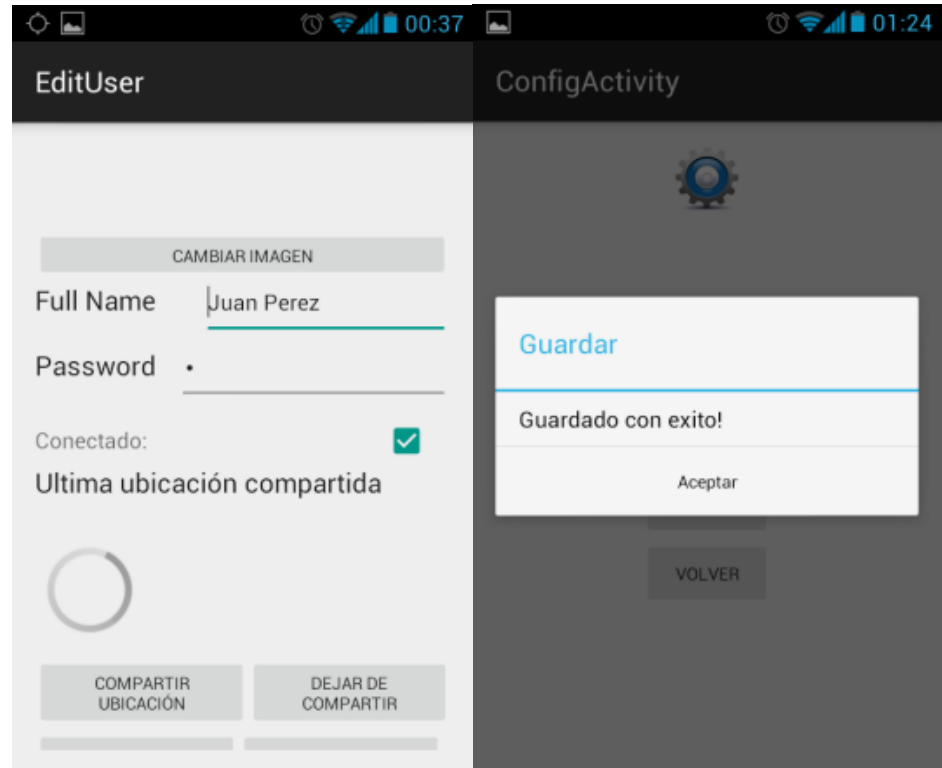
# Aplicación - usabilidad

- Pestaña de chats abiertos y de todos los contactos, deslizando el dedo.
- Lista de contacto con la información visible rápidamente



# Aplicación - usabilidad

Informar al usuario  
en que estado  
están las acciones  
que realiza.



# Demo

# Lecciones aprendidas

- No subestimar requerimientos técnicos ni nuevas tecnologías.
- Hacer tests a la par que el código, y no dejar mucho por testear al final.
- Poner más foco en la estética de la app.
- Estimar tiempo de reserva para imprevistos y atrasos en cada iteración.



# Preguntas

