Drone Controller

ALBERTI Alessio
ANDRIAMANGA Niry
PACELLI Giuseppe
SVENDSEN Heidi
TATARANNI Giovanni

EURECOM
Sophia Antipolis

http://gtataranni.github.io/remote-access/

Main Idea

This is a client-server application that allows the user to send remote commands between Android devices.

It was designed with the idea of having one or more devices onboard a flying drone.

The Android application can send and receive messages. Received messages will be interpreted to perform actions described in it.

Opportunity Space

- More and more devices support Android.
- Drones are getting increasingly cheaper.
- Drones can be used for many different purposes.
- Combining a drone with a smart device (on board computer/smartphone/…) gives us a powerful device.

Business Model

FOSS - Free Open Source Software

Open source Application for research purposes: we develop the base framework and the communication protocol, then other researchers and developers can use it to extend it and add further custom functionalities.

We will release the software using the Apache Software Licence 2.0, this will let us collect credit for the original work (“Attribution”).

Target users: Researchers and developers that want to use the application for research and professional usage.

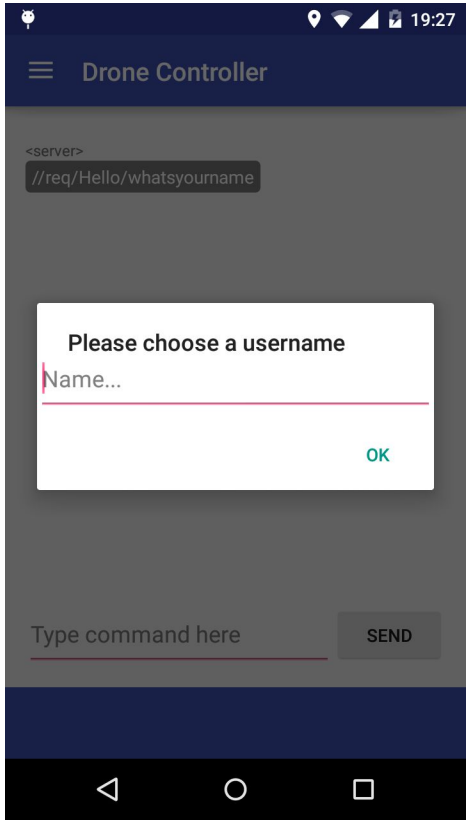
Generic protocol scheme

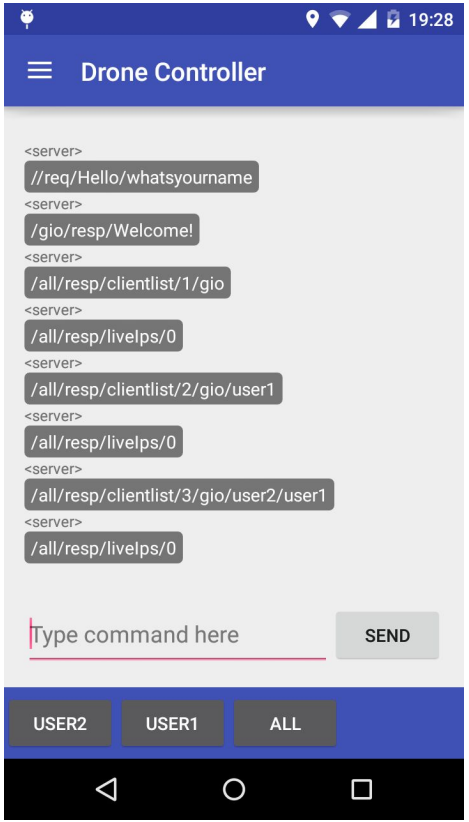
/recipient_user/command/action/params/moreparams

- Recipient_user -> User to send the command to (can be All)
- Command -> Req, Resp, Exec.
- Action -> Gps, Photo, Live, Network, ...
- Params -> specific for every different action.

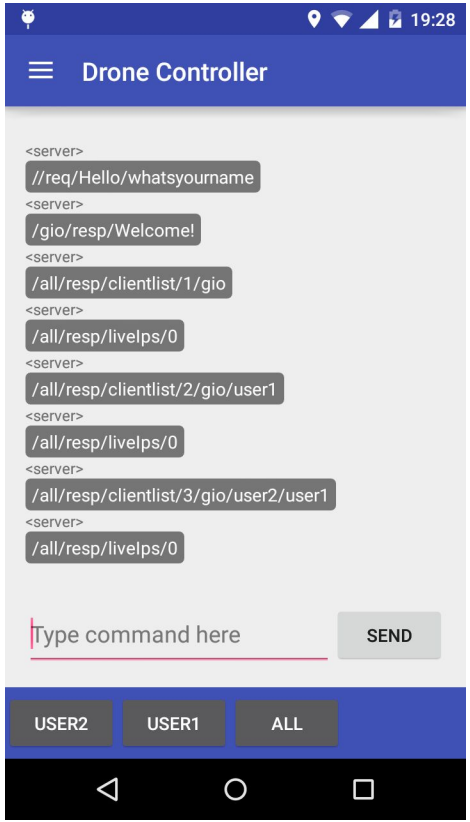
Find out more on http://gtataranni.github.io/remote-access/

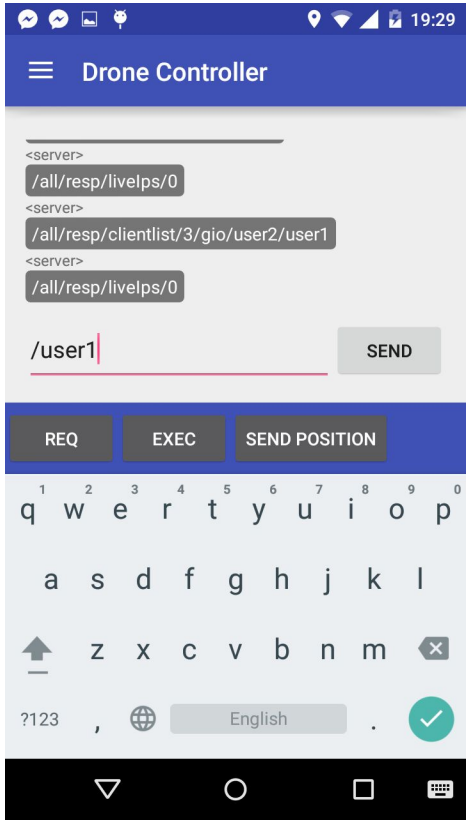
GUI and Clickstream

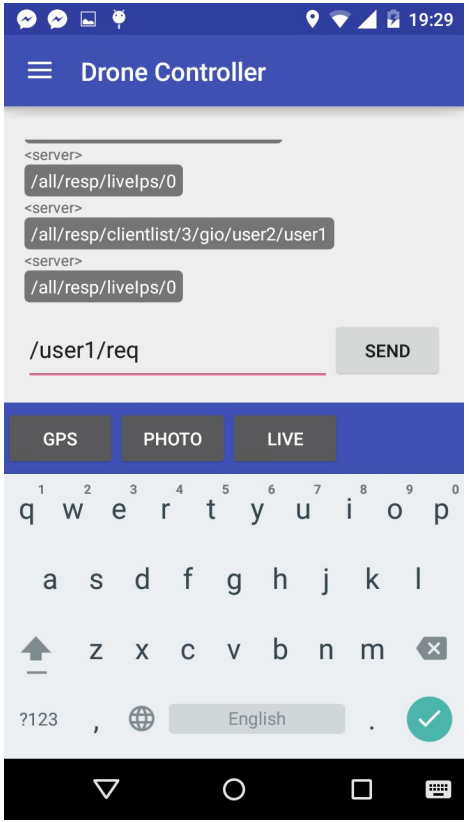




GUI and Clickstream



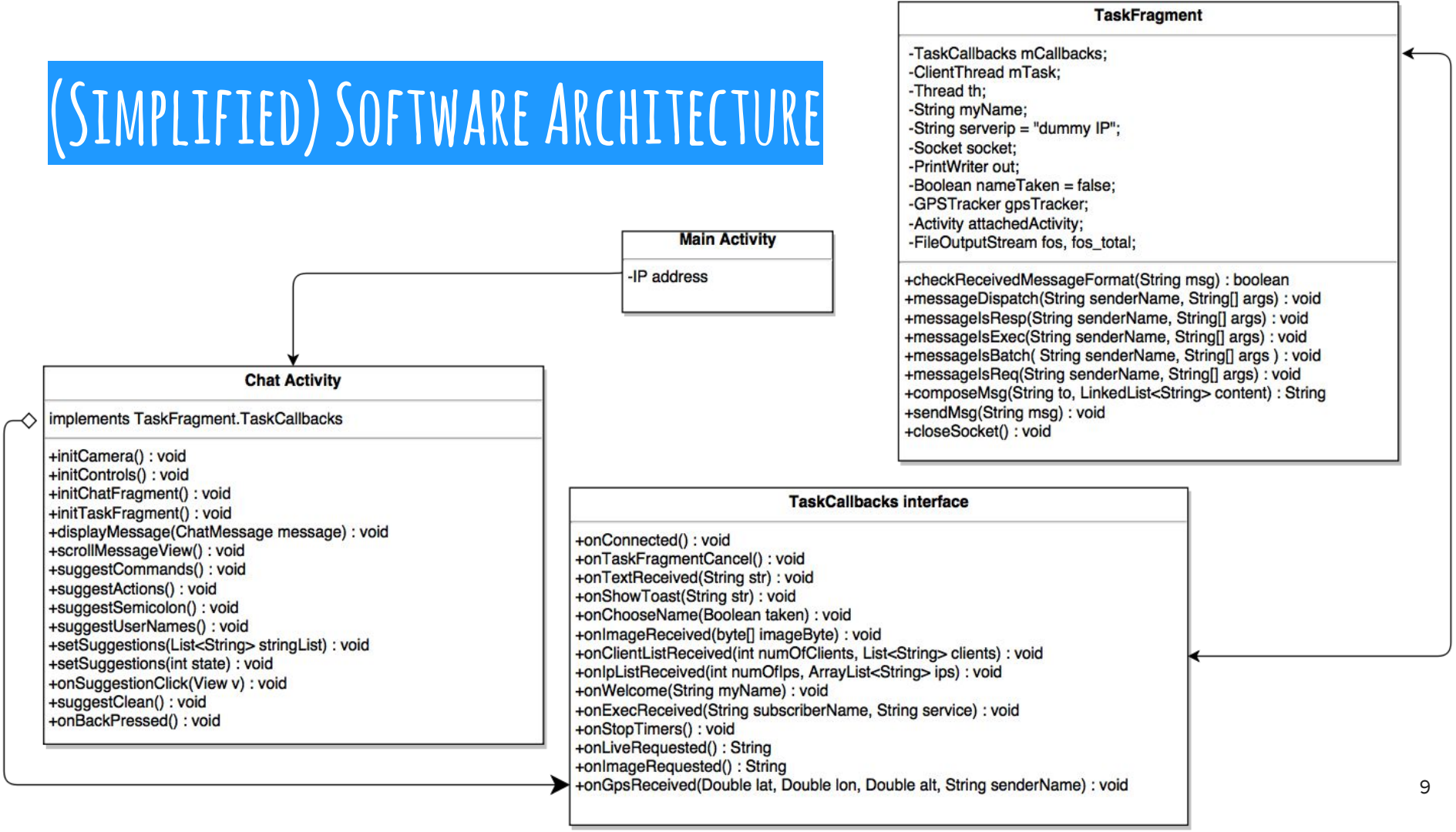




Use Case

- Python server runs in LAN
- User opens the application, chooses the ip address of the server
- Once connected, user will see the list of actually connected clients
- User can ask another client for a picture
- User will then receive the picture, it will displayed in a new view and saved to an application-internal gallery
- User ask another client for its GPS-coordinates
- User will receive a map, centered with the coordinates received

(Simplified) Software Architecture



High level Software architecture

Connection used

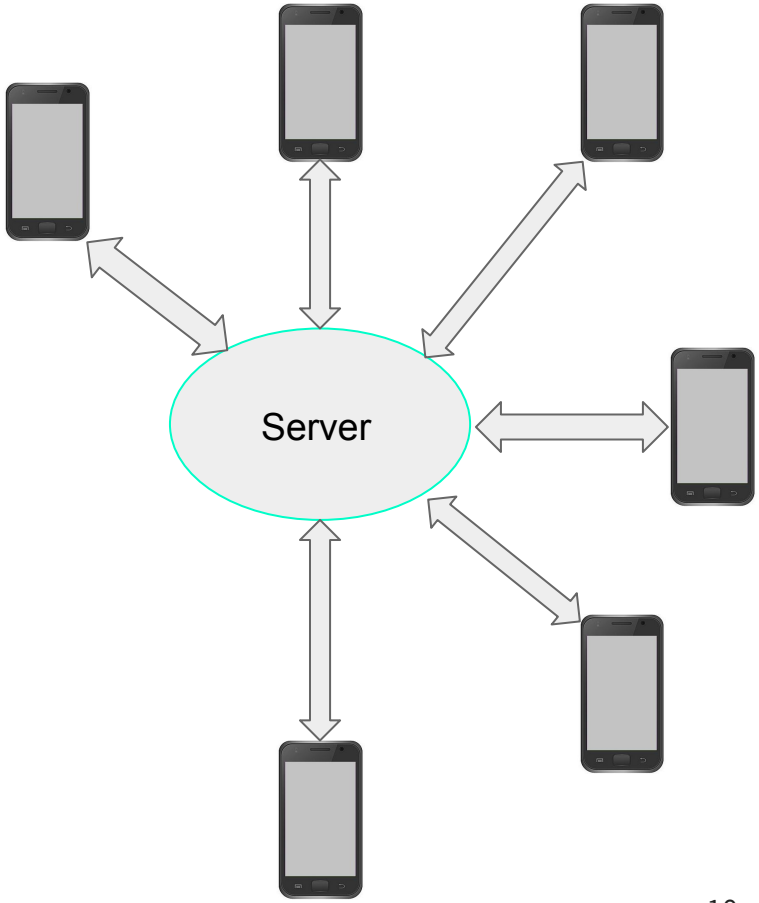
Socket (TCP/IP)

Server

Uses Python's library “twisted”. It will forward information and manage connections. Will direct the messages to the right receiver.

Clients

Will send and receive messages from other devices. Messages are interpreted and corresponding action executed.



Framework

Client Side:

- Plaintext communication protocol
- Callback interface
  - ↳ Re-implementable in new Activities
  - ↳ Easy to extend

Server Side:

- Python Twisted script, basic functionalities

Project Status (pre-release)

Testing: On Android phones and Emulator

Completed:

- Properly connect/disconnect from the server
- Proper interpretation of server messages
- Easy message composability/suggestions
- List of connected users
- Ask/send GPS coordinates in plaintext
- Show received GPS position on a map
- Send GPS position through clicking on a map
- Ask/Send pictures to/from other clients
- Save received pictures to memory
- Save received pictures in app gallery
- Stream video
- Improved application connectivity by keeping a thread in background
- Improved GUI
- Log activity to file

Future Developments

- Event-based subscription for batch commands
- Extend the range of data that can be requested from sensors
- Interact with the drone API and allow remote piloting of a drone
- Extend time-based subscription to more features
- Allow custom time period for subscription

Task Distribution

Alessio
    Software development

Niry
    Marketing

Giuseppe
    Software development

Heidi
    Management, Documentation

Giovanni
    Software development

CREDITS FOR PREVIOUS WORK (SERVER): Guillaume Dardelet, Chuan Wang, Dian Yu - Eurecom Mobserv class, fall 2014

Drone Controller

View project on GitHub

University project for MobServ class at Eurecom 2015

Drone Controller

University project for MobServ class at Eurecom.

The purpose of the project is to build a client-server application that allows to send remote commands between Android devices, having one (or more) of them onboard on drones. The server is a simple chat server in Python/Twisted, and the client is an Android app that sends/receives messages and performs actions described there.

Supported features so far are requests of gps position, photo shooting and live streaming.

Potential new features include a remote pilot for the drones.

Download .zip file

Download .tar.gz file

is maintained by gtataranni.

This page was generated by GitHub Pages using the Architect theme by Jason Long.

FIND OUT MORE ON http://gtataranni.github.io/remote-access/

Demo

Thank you for your attention!

Insights

Similar Applications

Existing applications(non exhaustive):

CommandFusion: software that allows the user to remotely control any IP-enabled devices from his/her Android or iOS device.

Tablet Remote: Android application that lets you control an Android tablet or phone with another Android device.

CAMERA

Twisted server is able to receive only plain text

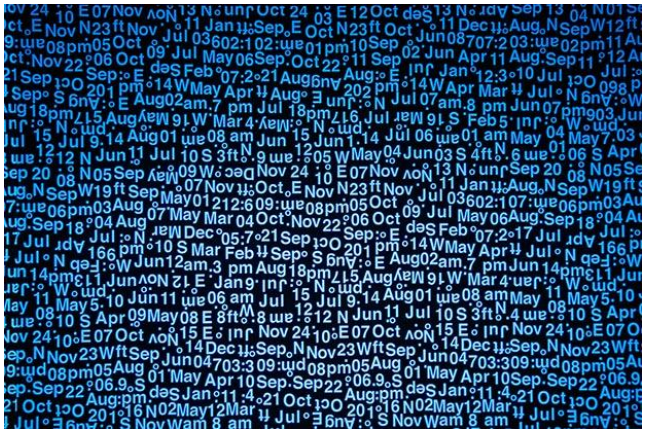
- Server side:

Create another modality -> RAW DATA

Header : /recever/write/photo -> enables the raw data mode

Server changes state -> Retrieve and Forward data without caring about the format

Ending tag = _end_



- Client side

Using a **surfaceView** to show the preview of the camera

Set the *back camera* to be used, the l*owest possible resolution*, the *Auto flash* and *Auto Focus*

TAKE A PICTURE

start the preview

wait for the focus

get the picture

SEND A PICTURE

convert the picture in a string by using the base64 encoding.

send header, picture and ending tag