**AA-UAV****Functional Specification V0.001**

Shmulik Melamed & Lital Motola  
Last Updated: January 21, 2013

**- C O N F I D E N T I A L -**

All Rights Reserved.

# 1. Overview

**AA-UAV** is an add-on to the AR.Drone Android Application that transforms the AR.Drone to a mini UAV meaning we can flay it further and higher, we can tell him to flay alone and we can get more information as (place on map, high, distance, time to flay) Apart of the Application Add-On we provide a web service that allows everyone to see all the AE.Drone that are flaying in Real Time on the map and get their details all over the web. Explain what the service is about and what it will do.

**This spec is not, by any stretch of the imagination, complete.**All of the wording will need to be revised several times before it is finalized. The graphics and layout of the screens is shown here merely to illustrate the underlying functionality. The actual look and feel will be developed over time with the input of graphics designers and iterative user feedback.

This spec does not discuss the Network and Website algorithms used by the AR.Drone and the user, which will be discussed elsewhere. It simply discusses what the user sees when they interact with **the Ar.Drone Application and Website.**

Overview 1

Scenarios 2-3

Non Goals 4

AA-UAV Plane User & Server Application Flowchart 4

Server Application by Screen Specification 5-6

Splash Screen

Main Page

Plane Application by Screen Specification 7

Splash Screen

Main Page

User Application by Screen Specification 8-9

Splash Screen

Piloting Page

Map Page

Open Issues 10

2. **Scenarios**

In designing products, it helps to imagine a few real life stories of how actual (stereotypical) people would use them. We'll look at two scenarios.

**Scenario 1: Mike.**

Mike works for the national geographic channel, and they always gives him new challenges, in the past it was easy you had only a camera that can take one picture at a time, but now it can take many picture at one time, and the picture came in High Definition, and so because that technology is developing fast then Mike can take pictures that he couldn't take before, there for when he got his know challenge to picture animals and views above the sea or inside of a tunnel that could not fit in then he needed to find something that can help do this stuff.

Miki heard from a friend that there is a new application to the AR.Drone that can control him from a great distance and that’s exactly what he needs to shoot his new pictures, Miki decided to try this new app plane and so he downloaded it.

After a few test flights, Miki got how to use the new app and started to shoot very hard pictures that he could not shoot without it, such as shooting from a cliff or in the open sea or even shooting a chase after a train or a car, and all from a very long distance.

Last week Miki deliver his shooting to his manager, the manager was so proud that he gave Miki a big promotion and a very fat salary (After that Miki lived happily ever after).

**Scenario 2: Bobby.**

Bobby is 40 years old and he is the manager of a big electronic factory, in the last 5 years his factory is getting attack by thief's, so in order to handle that problem 3 years ago he installed outside the factory surveillance cameras that film almost every place for 24 hours hitch day, and it is also start an alarm that calls his cellular.

The Thieves are smart, they come at night wearing black masks and there for every time that the camera catch them and bobby goes with those films to the police station, thy tell him that they can't do anything because their faces is covered, and the only way is if the police catch's them in action.

Bobby gets that he needs to do the police work because they won't do it, so he starts to look for applications and websites that can help him catch those thief's, and so he came By chance to an application that lets him control a plane from any distance and that can bypass every obstacles in real time automatically, something that can help bobby while flaying that plane at night.

Bobby bought that application and placed the plane on the roof of his factory and want home (30 Kilometer from his factory) for a good night sleep, suddenly the alarm goes on in his factory and the alarm calls his cellular, bobby wakes up immediately and opens the AA-UAV application and start controlling the plane, so now he sees and recording the Thieves in action from the plane and after they finished stealing he starts following them until they stop in a Warehouse. Then bobby called the police and they finally got them. Bobby was so excited that he doesn't even get out of bad to catch does Thieves. From this day on bobby Caught every thieve using that AA-UAV Application.

# 

# 3. Non Goals

This version will *not* support the following features:

* Transforming picture from the AR.Drone to the application over the 3G communication.
* Web Service displaying the AR.Drone details.
* Evasion from risks on flight such as buildings and weather.

# 4. Key Flows

We'll have time later to go into mind-numbing detail, but for now, let's look at a quick flowchart of the service so you get the big picture. This flowchart is not complete, but it does give you the right idea for the "storyboard" of using **AR.Drone Plane User & Server Application.**

# J:\פרויקט גמר\מסמכים מוכנים\Drawing1.jpg

# 5. Server Application

***Ar.Drone Server Application*** purpose is to establish a connection between the user and plane application in order to send and receive messages.

According to the data/commands that the server gets from the user or from the plane, it knows how to handle with it by the algorithm written in the server background application. The algorithm will be explained in the future documents.

The server networking is built with PHP SQL and http request massages over the 3G Cellular Network.

# 6. Plane Application by Screen Specification

***Ar.Drone Plane Application***consists of few different screens. Most screens will follow a standard format, with a look and feel to be designed in the future by a graphic designer. This document is more concerned with the functionality and the interaction design, not the exact look and layout.

All screens are created in Java Android.

## Splash Screen

## A picture of the AR.Drone that will disappear in 5 seconds.

## Main Page

Displays when the Splash screen is complete. The Main Page is the only page in the plane application and it has three purposes: 1) Display us red screen with a massage "can't connect to ……" that tells us witch component is not responding or connected to the phone (server arduino or plane). 2) Display us green screen with a massage "establish connection to all devices" if all of the components are responding and connected. 3) Running code behind that receives massages from the client through the server and handles those massages.

After we started the server on the computer we can start the plane application on the android device and see if the server or any other component is on or off.

IF the main page display us a green screen with the massage "establish connection to all devices" then he knows that the connection is ok and then continue to generate key to insert in the user application for identification.

IF the main page displays us a red screen with the massage "can't connect to ……." then we have a problem, we need to check the connection to all components and then press on the button "check connection".



**Check connection**

Can't connect to arduino\plane\server

Establish Connection to all devices

The plane application has a code behind that can get initial commands from the user application through the server application via 3G and then interact with the plane via Wi-Fi and then fly this plane on its own.

# 7. User Application by Screen Specification

***Ar.Drone Plane Application***consists of few different screens. Most screens will follow a standard format, with a look and feel to be designed in the future by a graphic designer. This document is more concerned with the functionality and the interaction design, not the exact look and layout.

All screens are created in Java Android.

## Splash Screen

## A picture of the AR.Drone that will disappear in 5 seconds.

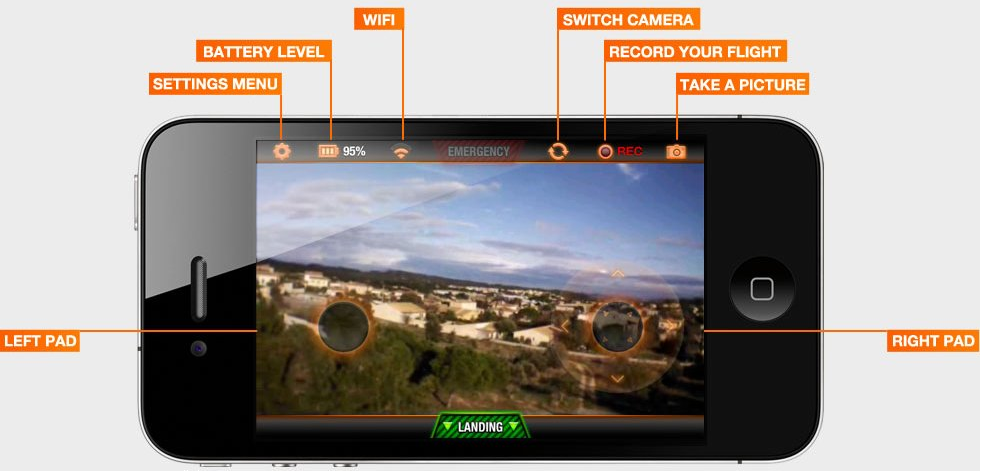
## Main Page (Piloting)

Displayed when the Splash screen is complete. The Main Page is a screen that asks us to enter the key that the plane application generated for us for identification, if we enter a different key the server will know and the input box will turn blank for entering the key again, if we enter the correct key we will transfer to the flight screen.

The flight screen let us control the AR.Drone.

This page also gives us options and information about the AR.Drone such as:

1. Settings Menu – direct us to a settings page where we can change out preferences about how the AR.Drone flays.
2. Battery Level – show us the battery level of the AR.Drone.
3. 3G – show us if we using the 3g cellular network (if not automatically it will work with Wi-Fi).
4. Switch Camera – let us switch the camera of the AR.Drone from camera one to camera two.
5. Record You're Flight – let us recording our flight on the USB connected to the AR.Drone.
6. Take A Picture – let us take a single picture from the camera.
7. Emergency – will light on if something goes wrong and display massages in the middle of the screen such as battery low, connection lost.
8. Left Pad & Right Pad – Pads that let us control the flight located in the center of the screen.
9. Landing – we can press this to land the AR.Drone in any situation.
10. Map – direct us to the map page where we can choose a track and let the AR.Drone to fly by itself.
11. Information – display in the top left screen information about the flight such as (altitude, longitude, battery level of the phone located on the AR.Drone, flight time, weather etc.).

C:\Users\שמוליק\Downloads\1354617228_Pin.pngC:\Users\שמוליק\Downloads\1354616593_Information.png 

**MAP**

**INFORMATION**

**This is**

**what we**

**see when pressing the informatin button.**

Flight Information:

Fly Time: 50min

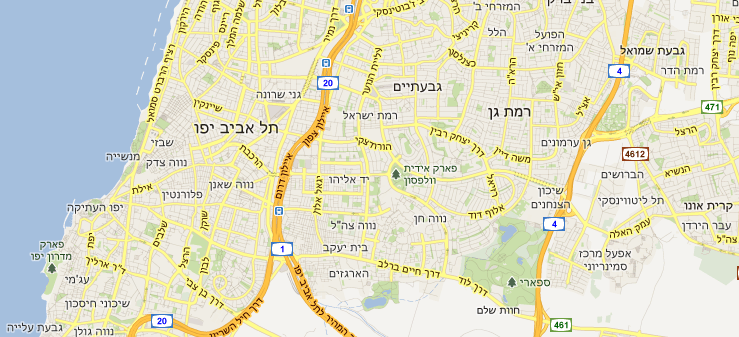
Latitude: 30.6568

Longitude: 36.5048

Battery: 20%

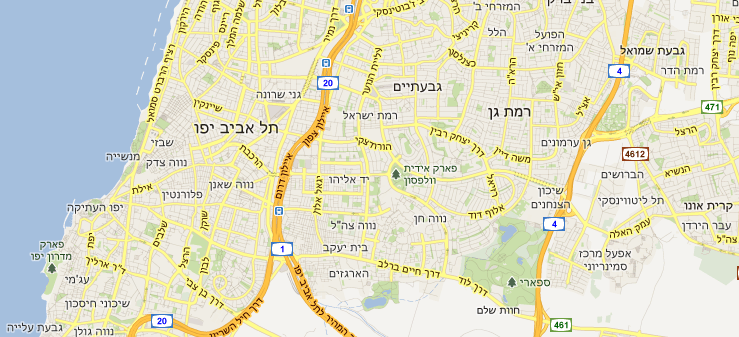
## Map Page.

In this page we can see where the AR.Drone is located on the map in that exact second and choose where we want that he will flays on its own by clicking in any place on the map, after that click there will be a massage that ask us if to continue or cancel the action, if we cancel the action we return to the piloting page, if we continue the action we will return to the piloting page but without the pads that let us control the AR.Drone, In this situation if we want to take back the control we gest need to click anywhere in the middle of the screen and the pads will come back.



We can see that in the first time that we see the map page he is marking us the exact place of the AR.Drone, now another press in any place on the map will mark us the destination of the flight.

**My Current Position**



Are You Sure?

**Cancel**

**Continue**

After we choose the destination point, it will also be marked on the map and we will get a confirmation massage.

**Destination**

**My Current Position**

Pressing on the continue button direct us back to the piloting page but also sends a command to the Plane Application that activates the algorithm that is responsible to flay the AR.Drone without any control of the user.

When the AR.Drone is finished the track on its own, he will give us the flight information and ask us if want to land, return or take control.



**Return**

**Take Control**

**Land**

Flight Information:

Fly Time : 50min

Latitude ;: 30.6568

Longitude: 36.5048

Battery : 20%

What do you want to do?

# Open Issues

1. Understand how the original AR.Drone application code is working.
2. Equation that can let us land the AR.Drone according to the battery level and the altitude.
3. Integrating the weather and flay the plane according to it.