The viewFinder scenario

By the Wezitat team: Jane, Gary, and Adrian

Contact: jane@, gary@, adrian@wezitat.com

Date: June 2015

# Assumptions:

* User has downloaded the app on their phone
* User opens the app at current location
* User opens the viewFinder to see what wits are here

# Definitions:

# **viewFinder =**

# Screen Shot 2015-06-17 at 4.57.38 PM.png

# The **viewFinder** is the app’s main screen, used to discover **witMarkers** (the yellow and green indicators) and **witImages** (the flag).

**cameraView =**

Live camera feed from device as a background. Focus, shoot, and other controls are disabled.

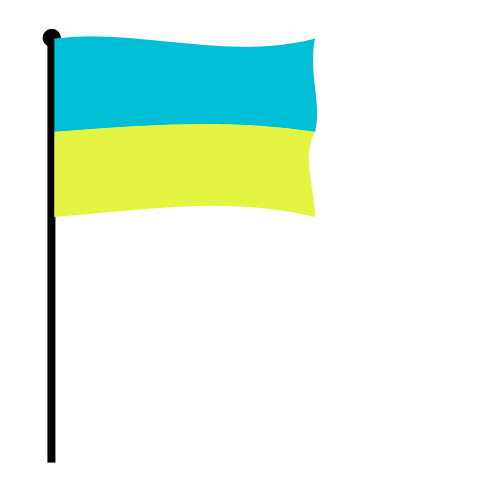
# **wit =**

# A **wit** is a container for digital media such as an image, music file, video, or voice recording. A **wit** isplaced at a geo-location so a user can discover it with the **viewFinder**. The **wit** also has attributes such as author, geo-location, date, media type, status etc. A wit appears on the screen as either a **witImage** or a **witMarker**.

A **wit** is represented by a **witImage** when it is **here** and **visible**, meaning that it is within a given geo-fence and is on-screen in the **viewFinder**. Otherwise it is represented by a **witMarker**.

One important status for a **wit** is whether it has been claimed, meaning the user discovered the **witImage** and tapped it.

**witImage =**

 or w回 (video) or w♫ (music)

# Image component of the **wit** that is shown on screen. Most wits are going to have images as their main media attribute, and this image is the **witImage**.

# 

# **wits** that have video/music as their main media attribute are shown on the screen by a standard, pre-defined **witImage**.

**visibility** =

This is an important concept. When a **wit** is **near** and can be seen on the cameraView it becomes visible. A visible **wit** is shown as its **witImage** and not as a **witMarker**. The goal of the app is to make wits visible, and users will need to move, pan, tilt, travel to get their wits visible. The software needs to calculate visibility by using the geo-location, altitude, heading, gyroscope and perhaps focal length.

# **witMarker =**

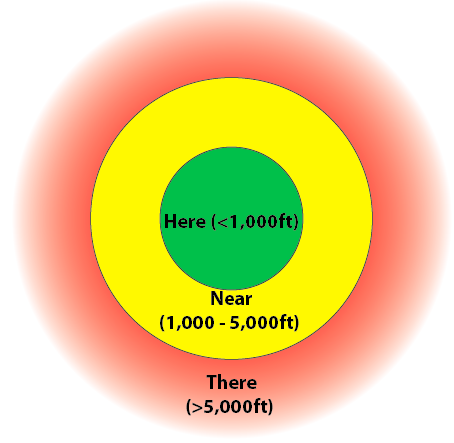
# 

# like pins on a map, these are used as visual cues on the **viewFinder** to point out the location of **wits** that the user has discovered or is looking for. **witMarkers** indicate direction and distance info when they are on the **viewFinder**. **witMarkers** can be:

# **here** = in an inner geofence and colored green (probably < 1000ft; controlled by a parameter). When a here witMarker is within the cameraView, it is visibleand is shown as a **witImage**

# **near** = outside the inner geofence but in the outer geofence and colored yellow (between 1000ft and 5000ft; paramerized)

# **there** = outside outer geofence and colored red

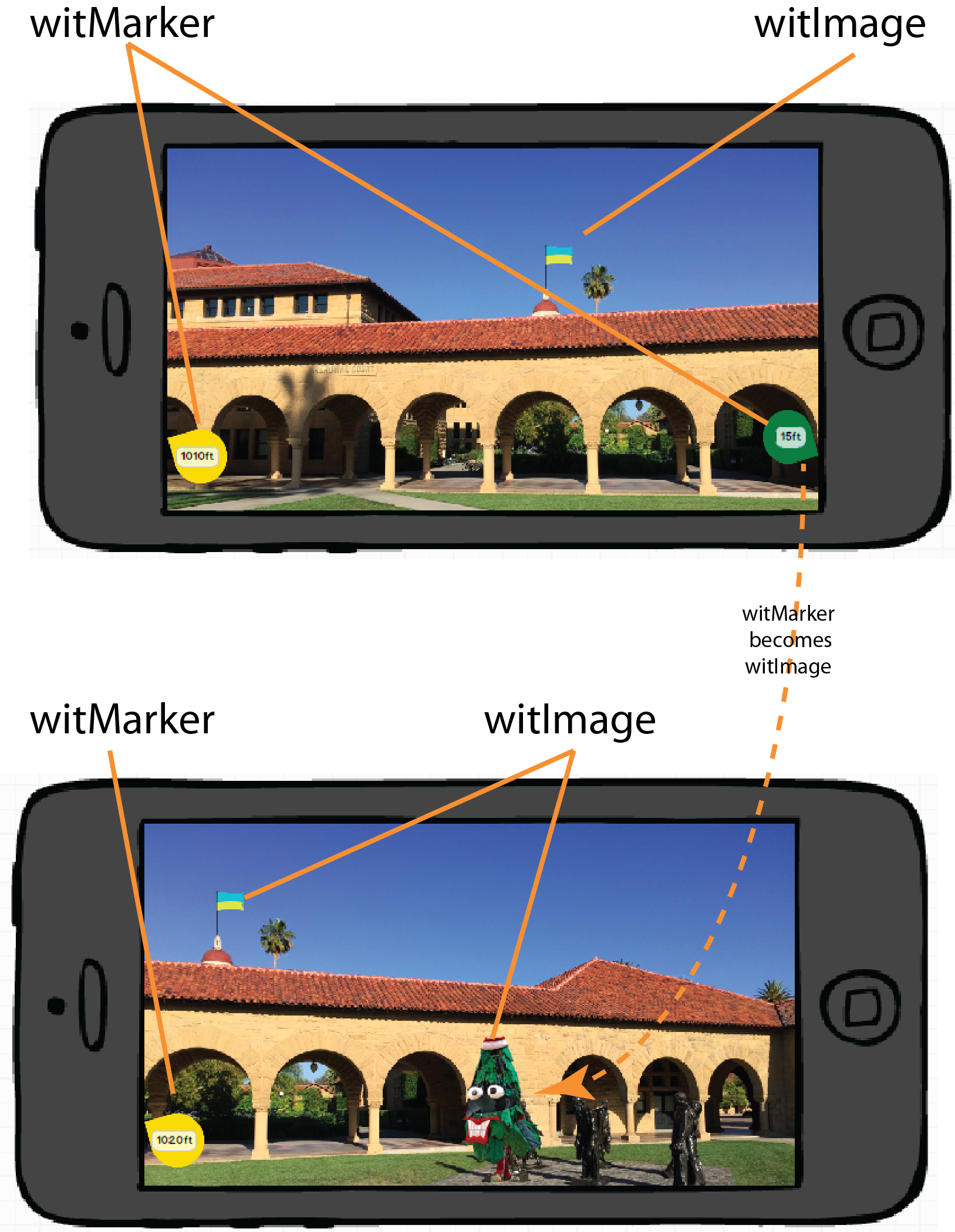
(not to scale)

# Scenario:

The app gets the user’s geolocation, and calculates all the **wits** within his/her here/near/there geofence. The app opens to the **viewFinder** with **here** and **near wits** overlaid on the **cameraView** as **witImages** or **witMarkers**.

**ViewFinder** opens with:

1. Camera view as background.
2. **Wits** that are in your **here**/**near** geofence are overlayed on the camera view. **Here** **wits** have a green **witMarker**; **near** **wits** have a yellow **witMarker**. If a **wit** is **visible** (i.e. it is **here**, close enough and the camera is aimed at it) then it’s **witImage** pops up automatically.
   1. Claimed **wits** that are **visible** will just be there virtually in full view. If you walk away from a claimed **wit** it will usually turn into a **here witMarker**
   2. Unclaimed wits that are **here** will be represented by either a green **witMarker** or a **witImage.**  If they are not in your current view then the green **witMarker** has a direction pointer and you have to pan/tilt/walk to get them into the view. When they become visible then the **witImage** pops right up on the **cameraView**. If multiple wits are very close it is possible to have multiple **witImages** pop up/down as you move the camera.
   3. Unclaimed wits that are **near** will be represented by a yellow **witMarker** with direction/distance pointer
3. Actions available on **witMarkers** and **witImages** in **cameraView**
   1. selection (tap) on a yellow or green **witMarker**
      1. show popup overlay - **wit** name, owner, date placed, status, standard wezitat logo
   2. selection (tap) on a visible **witImage**
      1. already claimed: show popup with **wit** name, date claimed
      2. previously unclaimed:
         1. show popup with message - “congrats! you claimed me” and
         2. change the status of the **wit** to “claimed”
         3. show a button to notify “[test@wezitat.com](mailto:test@wezitat.com)” by sending an email to that address



**Top mockup:** user is facing South and sees green witMarker, yellow witMarker, and Ukrainian Flag witImage.

**Bottom mockup:** when user pans to the right (or West), the green whitMarker becomes a Stanford Tree witImage. Ukrainian Flag witImage and yellow witMarker both remain.

# Example videos:

<https://www.youtube.com/watch?v=AX7UTKGvRE0>

<https://www.youtube.com/watch?v=BrI3Z-BTR5Y>