

# Using ArcGIS Online and AppStudio to Create a Comprehensive Citizen Problem Reporting Platform

Levi Coldiron

GIS Coordinator

City of Salisbury, NC



# OVERVIEW OF PROCESS

1. Feature Class
2. Crowdsource Reporter Web Application
  - Create the web map
  - Configure web application
3. Internal Management Web Application
  - Create the web map with views for each department
  - Configure the web application
4. Custom Workflows
  - E-mail notifications using Python and HTML
  - Set up Windows Task Scheduler to run Python script hourly, or more frequently
5. Use AppStudio To Publish Mobile Applications To App Stores
  - Deep Customizations Using QML Editor (JavaScript)
6. Future Enhancements



# CREATE THE FEATURE CLASS

## **Required Fields**

- Request Type, Status, Address, User Info, Submission Details, Staff Notes, Open & Closed Dates
- Public View: Yes/No, Notification Status, User Tracking: UserID, Contact Info Fields

## **Symbology And Representation For Your Audience**

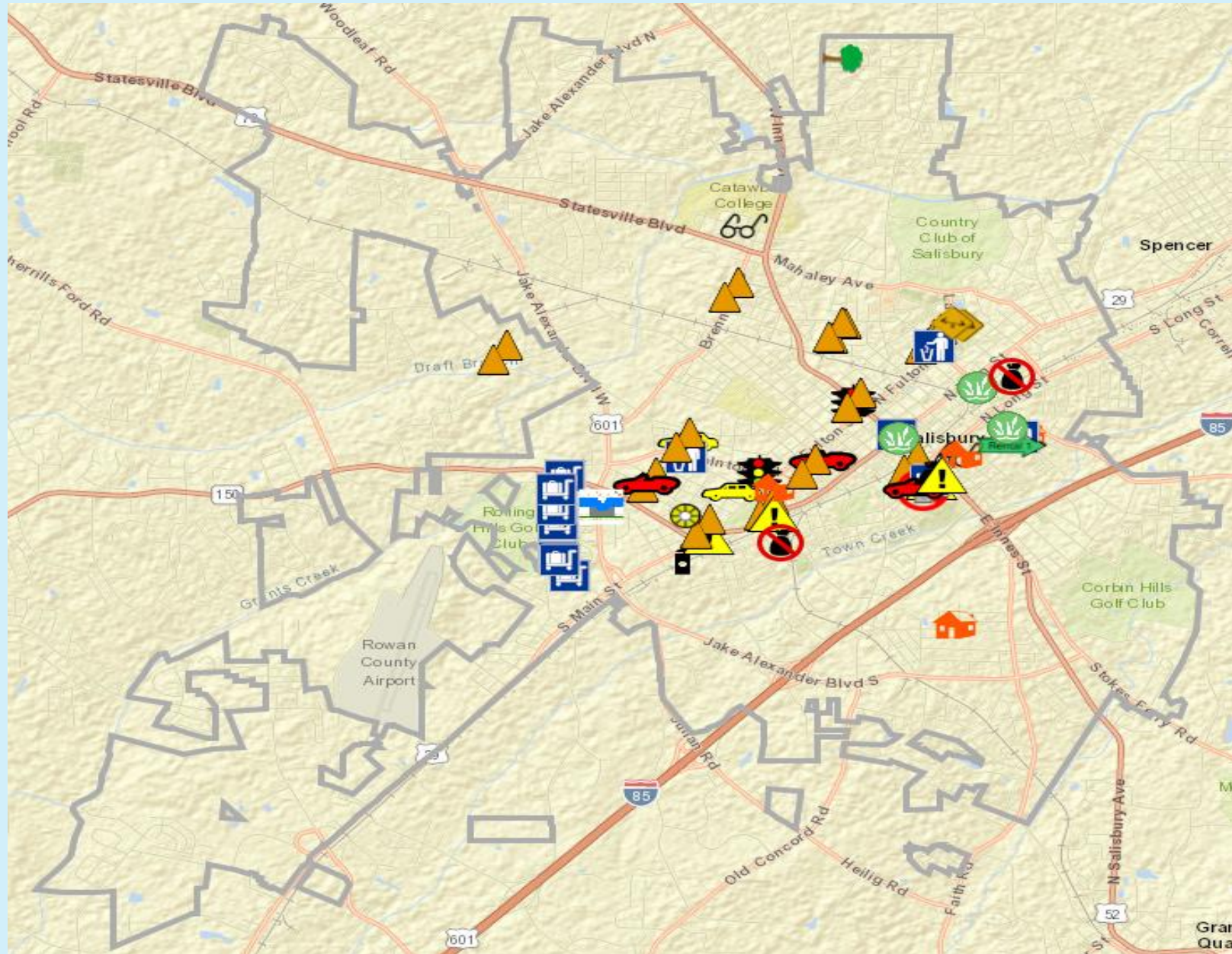
- Request Type or Status or something else

## **Host On AGOL/Portal Or On Local Server**

- Local ArcGIS Server does not allow for user tracking
- AGOL automatically adds fields for photo attachments, user tracking, editor tracking, etc. when you select those settings in AGOL.

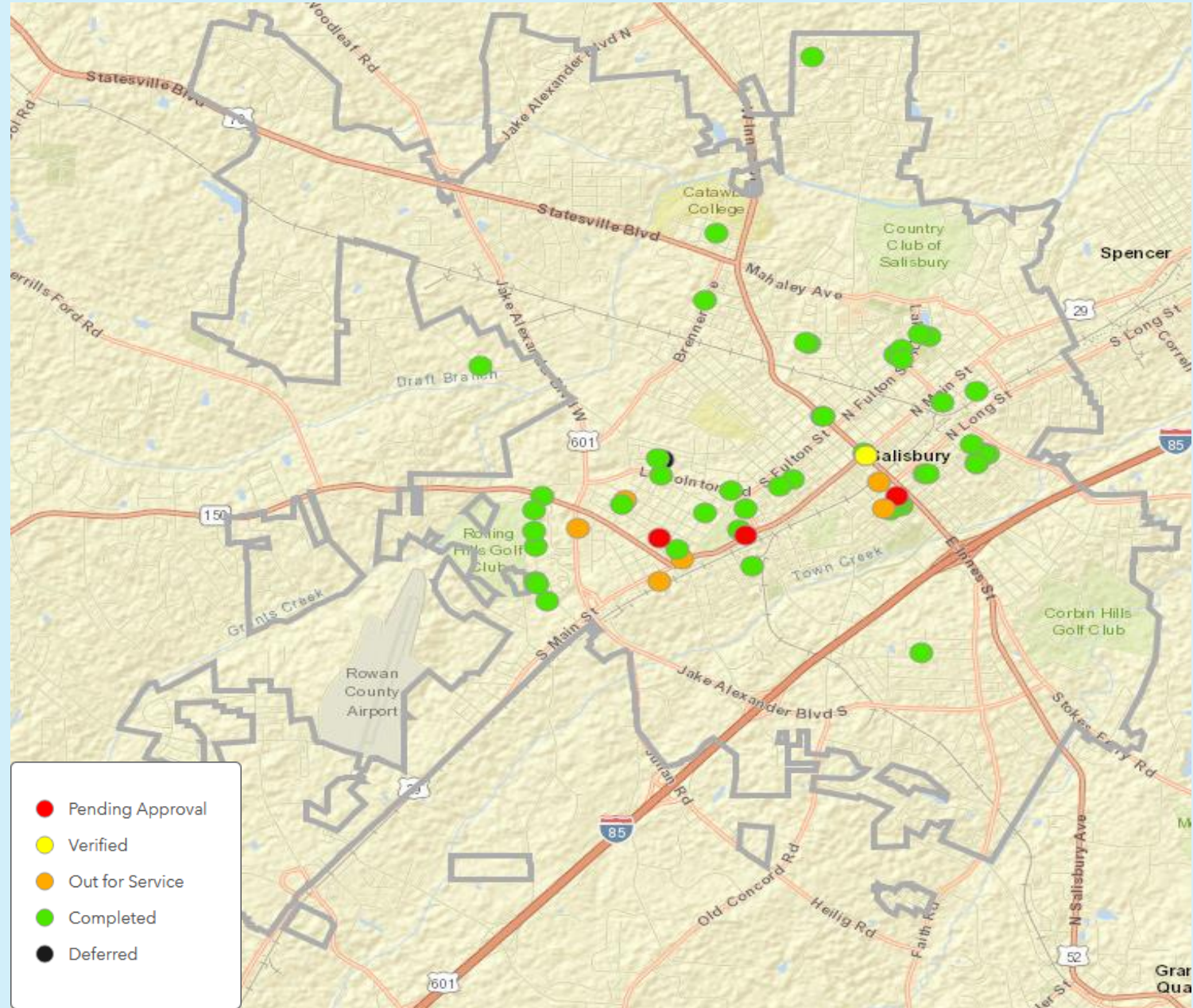


## Symbolized by Request Type





# Symbolized by Status



# CREATE WEB MAP TO DRIVE WEB APP

Add boundary feature class for jurisdiction(s)

- Note the Table Of Contents name to configure in the web app
- Multi-Jurisdictional apps should have a jurisdiction field in the feature class to determine notification e-mail routing

Create Filters

- Hide requests from public until verified (PUBLICVIEW=Yes)
- Show only requests that are open or have been closed in the past month (or any amount of time)

[CLOSED DATE] IS **BLANK** OR [CLOSED DATE] IS WITHIN THE LAST **MONTH**

Configure Pop-Ups to determine what the public sees when they click on an issue

- Withhold requestor name and contact information

# CREATE WEB APP FROM GROUP

Create AGOL Group for your public facing application to reference.

- The application references an AGOL group instead of a single map.

From the group page, click share button and click Create Web App button

- Select Crowdscore Reporter template and start configuring

Pick a unique splash page background image

Sign-in and User Tracking Options

## BUG ALERT!!!

ESRI is not allowing public accounts to submit reports unless the account has explicit permission.  
This isn't feasible for a public application of this nature.

- Anonymous Login (you can still ask for a user's name and contact info via fields)
- Social Media Login



Must register your app with Facebook and Google and provide IDs. Search Google for instructions.

# CREATE INTERNAL MANAGEMENT APP

## Create a management group in your AGOL organization

- Internal staff will need AGOL accounts and join this group to access management app
- Add Requests feature layer to this group

## Create an Admin Web Map

- Add Requests feature layer and jurisdiction boundary for reference
- Filters
  - Create Hosted Feature View Layers for filtering by department
  - Create departmental groups and add users
  - Create View Layers from the Requests Feature Layer Item Page
  - Set permissions
  - Add View Layers to the web map
  - Set filters for each View Layer to only display requests types that the department handles
    - Request Type = Overgrown Grass OR Request Type = Upholstered Furniture Outside of Home AND Closed Date is Blank OR Closed Date is within the last month





# Salisbury311 Management



Salisbury311 Admin



Salisbury311 All Departments



Code Services



Engineering



Public Services



Salisbury-Rowan Utilities



Stormwater



# CREATE INTERNAL MANAGEMENT APP (CONT.)

From the ADMIN group page, click share button and click Create Web App button

- Select Crowdsourcing Manager template and start configuring
- **Share to Admin Group Only**

**Salisbury311 Management**

Code Services 1 selected / 49 records

Request Type	Status	PUBLICVIEW	Address	Date Submitted	Details	City Staff Co
Abandoned Housing / Bad Housing Co	Out for Service	Yes	927 S Jackson	April 17, 2018	abandoned house not secure	THANK YOU
Upholstered Furniture or Appliances in	Completed	Yes	526 & 530 e bank st	April 13, 2018	Upholstered furniture and other crap on	THANK YOU
Grass Overgrown	Completed	Yes	514/516 e fisher st	April 10, 2018	Bushes grown into sidewalk, lot overgro	THANK YOU
Grass in Street	Completed	Yes	530 e fisher st	April 10, 2018	No curb, dirt in gutter, grass in street	THANK YOU
Abandoned Housing / Bad Housing Co	Completed	Yes	This is an abandoned home with an aba	April 07, 2018		THANK YOU
Abandoned Housing / Bad Housing Co	Completed	Yes	corner of E. Lafayette & N. MLK Jr. Ave	April 07, 2018	There is & has been a trailer park here. I	THANK YOU
Abandoned or Illegally Parked Vehicle c	Completed	Yes	1117 Crosby St	April 06, 2018	Cars parking on grass	THANK YOU
Grass Overgrown	Completed	Yes	200 W 14th St Salisbury, NC 28144	April 05, 2018	Extremely long grass between sidewalk	THANK YOU
Rollcart - Damaged, Missing or Left Out	Completed	No	Spankys	April 04, 2018	Alyssa	THANK YOU

**Filters:**

- Grass Overgrown
- PUBLICVIEW: Yes
- Status: Completed
- Address: 200 W 14th St Salisbury, NC 28144
- City Staff Comments: THANK YOU FOR YOUR SUBMITAL CODE SERVICES WILL INSPECT ON MONDAY.

**Map:** A map view showing the location of the selected request (200 W 14th St Salisbury, NC 28144) with a green tree icon. The map includes street names like W Franklin St, N Church St, N Main St, E Lafayette St, N Railroad St, E Lee St, N Long St, and W Main St. The map is powered by Esri.

# CUSTOM WORKFLOWS

## Python Script for E-mail Notifications

- URL is Requests Feature Service URL + “0/query” (0 is the LayerID in the map service)
- Outfields are the fields you wish to output in the query
- If CreateDate > Current time – 1 hour then we start appending the OBJECTID and RequestType to the oidList

```
import urllib2, json, urllib, datetime, time, smtplib
from datetime import timedelta, datetime
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText

oidList = []

URL = 'http://services.arcgis.com/FKrJWv8CWiT6Rsn/arcgis/rest/services/Salisbury311_Requests/FeatureServer/0/query'
params = {'f': 'pjson', 'where': "1=1", 'outfields': 'OBJECTID, CreationDate, RequestType', 'returnGeometry': 'false'}
req = urllib2.Request(URL, urllib.urlencode(params))
response = urllib2.urlopen(req)
data = json.load(response)
for feat in data['features']:
    createDate = feat['attributes']['CreationDate']
    createDate = int(str(createDate)[0:-3])
    t = datetime.now() - timedelta(hours=1)
    t = time.mktime(t.timetuple())
    if createDate > t:
        oidList.append(feat['attributes']['OBJECTID'])
        oidList.append(str(feat['attributes']['RequestType']))
```

# CUSTOM WORKFLOWS — EMAIL NOTIFICATIONS

## Use HTML to draft the body of the e-mail

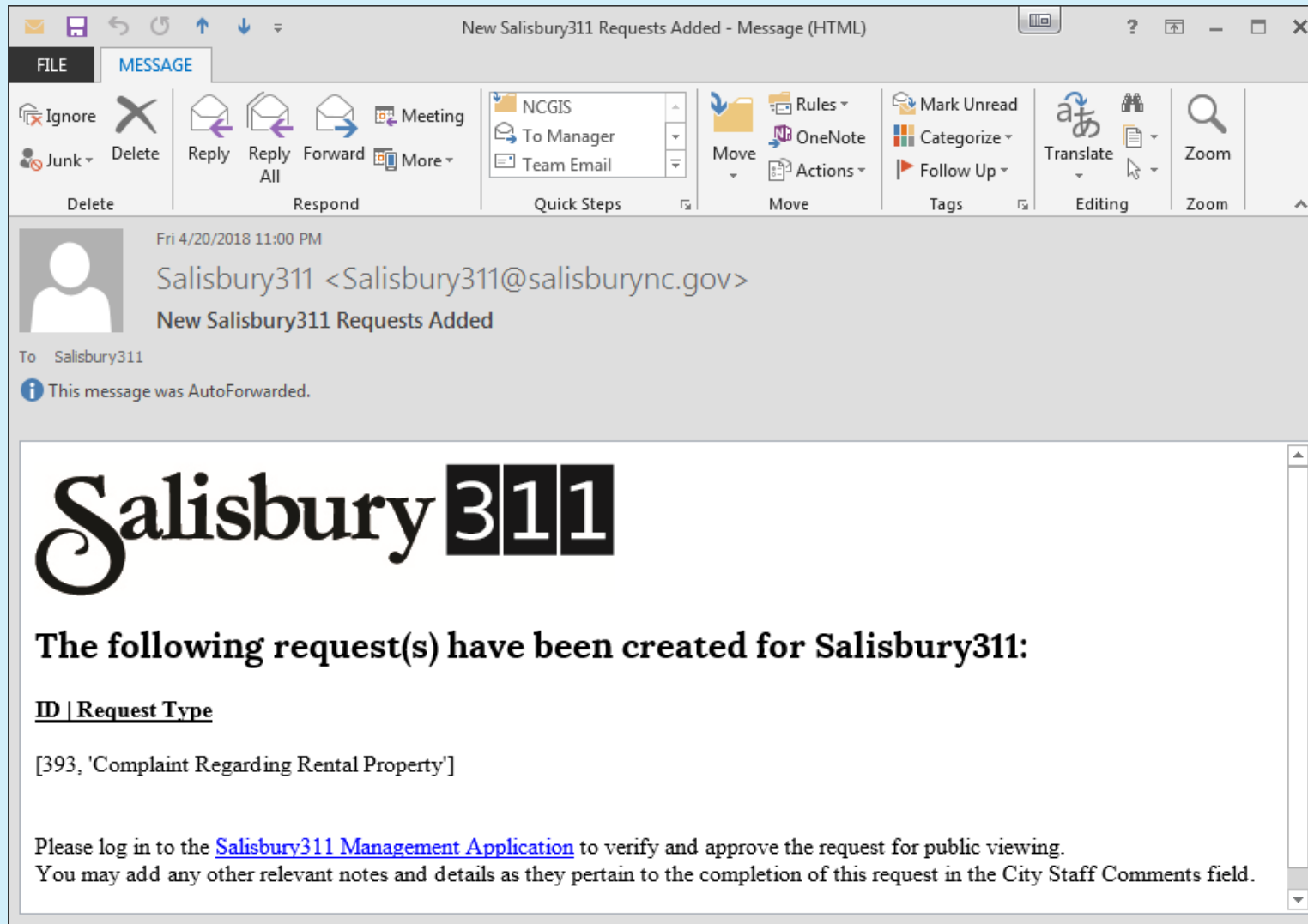
```
FROM = 'Salisbury311@salisburync.gov'
TO = ['Salisbury311@salisburync.gov']

# Create message container - the correct MIME type is multipart/alternative.
msg = MIMEMultipart('alternative')
msg['Subject'] = 'New Salisbury311 Requests Added'
msg['From'] = FROM
msg['To'] = ', '.join(TO)
```

```
text = "The following request(s) have been created for Salisbury311" + str(oidList)
html = """\
<html>
<head><a href="http://imgur.com/XtcUPpb"></a></head>
<body>
  <p><b><font size="5" color="black" face="Lora">The following request(s) have been created for Salisbury311:</font></b><br>
    <br>
    <u><b>ID | Request Type</b></u></b><br>
    <br>
    {oidList}<br>
    <br>
    <br>
    Please log in to the
    <a href="http://salisburyncgis.maps.arcgis.com/apps/CrowdsourcingManager/index.html?appid=8e96ff4f819043668577596cb256e04e">
    Salisbury311 Management Application</a> to verify and approve the request for public viewing.<br>
    You may add any other relevant notes and details as they pertain to the completion of this request in the City Staff Comments field.
  </p>
</body>
</html>
""".format(oidList=str(oidList))
```



# EMAIL OUTPUT



# CUSTOM WORKFLOWS...

Send the email if any feature meets the time requirements and if not, else: pass  
(if createdate < current time – 1 hour, then take no action)

```
part1 = MIMEText(text, 'plain')
part2 = MIMEText(html, 'html')

msg.attach(part1)
msg.attach(part2)

# Send the message via local SMTP server.
s = smtplib.SMTP('email.salisburync.gov', 25)
# sendmail function takes 3 arguments: sender's address, recipient's address
# and message to send - here it is sent as one string.
s.sendmail(FROM, TO, msg.as_string())
print "Successfully sent email"
s.quit()

else:
    pass
```

Set up Windows Task Scheduler to run this python script at the top of every hour

Set up Inbox rules to forward messages based on Request Types

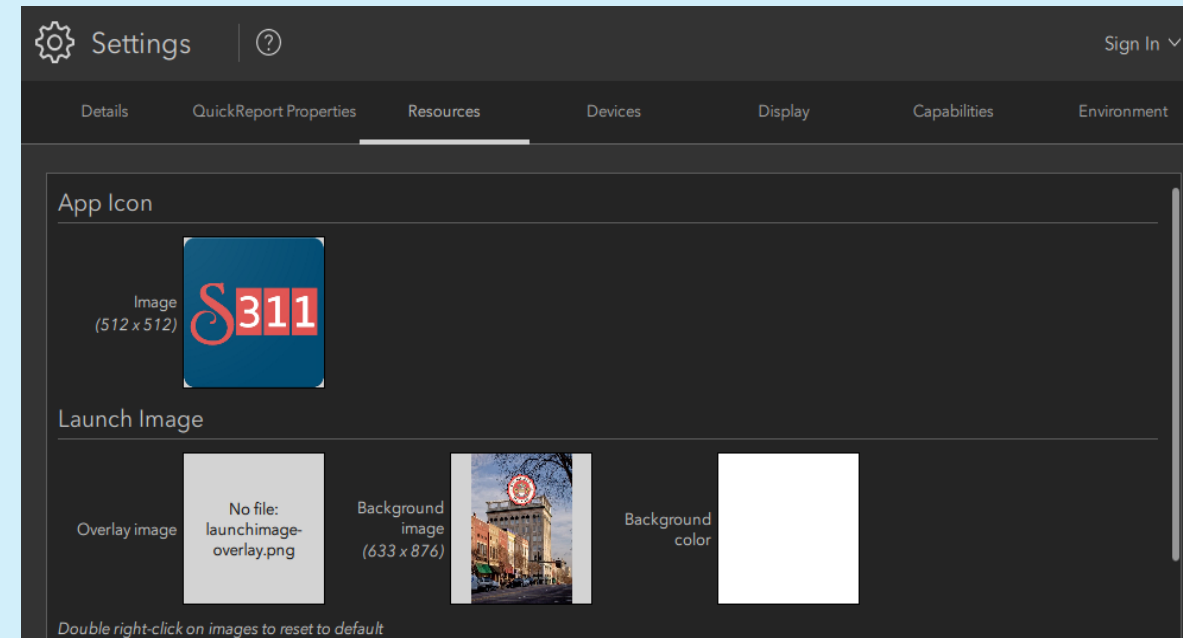
# USING APPSTUDIO TO CREATE NATIVE MOBILE APPS

## AppStudio Set Up

- Need ESRI Developer Subscription and AppStudio Standard License. ELA comes with 1 free license.
- You will create a separate developer organization in AGOL to store and register your app(s).
  - [developers.arcgis.com](https://developers.arcgis.com)

## Choose AppStudio Template

- “New App” in AppStudio
- Choose Quick Reporter template for crowdsource reporting
- Configure and customize inside AppStudio
  - Use same old “Requests” layer
  - Logos, fonts, front page background, page colors, etc.
  - Device, Display, and Capability settings  
(Hint: Do not include capabilities that aren’t necessary. Apple reviewers will reject the app.)



# CUSTOMIZATIONS IN QML EDITOR

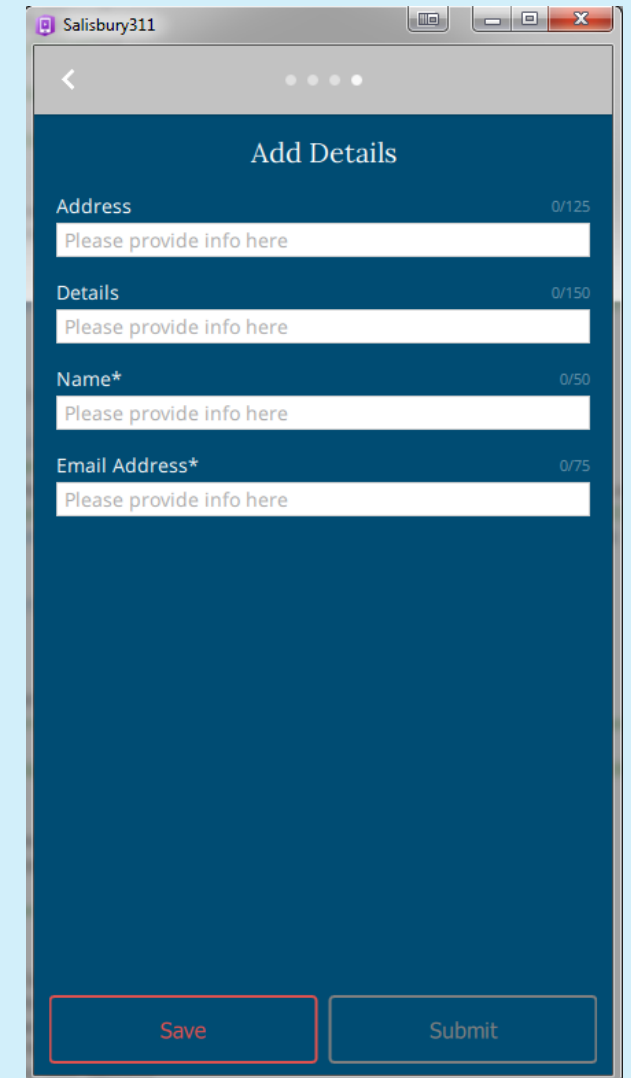
ALWAYS MAKE A COPY OF FILE BEFORE EDITING!

Deep Customizations to:

- Fonts and sizes for individual pages, page section background colors, text colors, button colors, button text colors, opacity, etc. with Dark Mode toggles

```
color: app.isDarkMode? "lightgray":"white"
```

- Feedback message texts
- Depending on your app's color scheme, you may need to make some changes to font and text input colors.
- Remove fields from Add Details Page <<<This is a crucial customization!



The screenshot shows a mobile application window titled 'Salisbury311'. The app's interface has a dark blue background. At the top, there's a header with a back arrow and four dots. Below the header, the title 'Add Details' is centered. There are four input fields, each with a label and a character count: 'Address' (0/125), 'Details' (0/150), 'Name\*' (0/50), and 'Email Address\*' (0/75). Each field contains the placeholder text 'Please provide info here'. At the bottom, there are two buttons: 'Save' and 'Submit'.



# REMOVING FIELDS FROM ADD DETAILS PAGE

```
676 ▲ if(capabilities.indexOf("Create")>-1){
677     var fields = root.fields
678     for(var i=0;i<fields.length;i++){
679         ///@@@f structure not right
680 ▲         if(fields[i].editable===true && fields[i].name!=root.typeIdField && fields[i].name!="CLOSE_DATE" && fields[i].name!="STAF_FEEDBK" && fields[i].name!="Lat" &&
fields[i].name!="Long" && fields[i].name!="Status" && fields[i].name!="GUID" && fields[i].name!="PUBLICVIEW" && fields[i].name!="Username" && fields[i].name!="CreationDate" && fields[i].name!
="EditDate" && fields[i].name!="Creator" && fields[i].name!="Editor" && fields[i].name!="AccountID" && fields[i].name!="SUBMIT_DAT" && fields[i].name!="GLOBALID" ) {
681             var f = fields[i];
682             app.fields.push(f);
683             fieldsMassaged.push(f);
684         }
685     }
```

Protip: Field names are case sensitive!!!

# PUBLISHING TO GOOGLE PLAY AND APP STORE

 [ESRI.COM/VIDEOS](https://esri.com/videos) Publishing Apps to Google Play  
Publishing Apps to Apple App Store 

## Developer Account Fees

- Google Play is \$25 one time fee. Apple is \$105 annually.

Publishing an iOS app requires use of a Mac computer.

# FUTURE ENHANCEMENTS

## 1. Geofencing

- Routes e-mail notifications to correct code services officer based on territory or zone
  - Will require code service zone layer and a field in the requests layer to be populated by code representing officers' territory.

## 2. More robust e-mail notification system

- Tiers that e-mail staff multiple times if a request hasn't been touched after a set number of days

## 3. AutoReply e-mails to citizens upon submission and status changes

# KEYS TO SUCCESS

- Set up the apps according to your organization's structure
- Find your super users, internal and external
- Commit to training for management and department staff
- **Outreach**



# THANK YOU!!!!

E-Mail: [LCOLD@salisburync.gov](mailto:LCOLD@salisburync.gov)

Web: [salisburync.gov/government/engineering/gis-maps](http://salisburync.gov/government/engineering/gis-maps)

Salisbury311 URL: [salisburync.gov/311](http://salisburync.gov/311)

Twitter: [@VPI\\_GeoHokie](https://twitter.com/VPI_GeoHokie)

