

Excellent question! Let me give you a clear guide on **which file to edit for what purpose**:

📍 **Visual/UI Changes - Where to Edit:**

📍 **Page Layout & Design:**

Want to change how a specific page looks?

- **Dashboard appearance** → Edit `pages/1_dashboard.py`
- **Analytics page design** → Edit `pages/2_analytics.py`
- **Export page layout** → Edit `pages/3_export.py`
- **Settings page UI** → Edit `pages/4_settings.py`

Example: Want to change dashboard metrics from 5 columns to 3 columns?

```
# Edit: pages/1_dashboard.py
# Find this line:
col1, col2, col3, col4, col5 = st.columns(5)

# Change to:
col1, col2, col3 = st.columns(3)
```

📍 **Colors, Themes & Styling:**

Want to change colors throughout the entire app? → Edit `config.py`

```
# config.py - Change all colors here
STATUS_COLORS = {
    STATUS_STRANDED: "#FF4444", # Change this red
    STATUS_EN_ROUTE: "#FFA500", # Change this orange
    STATUS_RESCUED: "#00CC66" # Change this green
}

SIGNAL_COLORS = {
    'strong': "#00CC66", # Change strong signal color
    'medium': "#FFCC00", # Change medium signal color
    'weak': "#FF4444" # Change weak signal color
}
```

These changes will automatically apply everywhere!

📍 **Map Appearance:**

Want to change map style, markers, popups? → Edit `modules/map_manager.py`

```
# modules/map_manager.py

# Change marker icons:
def _get_marker_properties(self, victim):
```

```
# Change 'exclamation-circle' to 'user' or any Font Awesome icon
return config.MARKER_COLORS[status], 'user', 1.0

# Change popup design:
def _create_popup_html(self, victim):
    # Edit the HTML here to change popup appearance
    html = f"""
<div style="font-family: Comic Sans MS;"> # Change font
    ...
"""


---


```

Chart Styles:

Want to change chart colors, types, or layout? → Edit the page file where the chart appears

```
# pages/2_analytics.py

# Change pie chart colors:
fig = go.Figure(data=[go.Pie(
    labels=['Stranded', 'En-Route', 'Rescued'],
    marker=dict(colors=['#FF0000', '#00FF00', '#0000FF'])]) # Change these
])

# Change from pie chart to bar chart:
# Replace go.Pie with go.Bar


---


```

Sidebar (Navigation):

Want to change sidebar appearance or navigation? → Edit app.py

```
# app.py - in render_sidebar() function

# Change navigation labels:
page = st.radio(
    "Select Page",
    ["Home", "Stats", "Download", "Config"], # Change these names
    label_visibility="collapsed"
)

# Add new metrics or remove sections


---


```

Functionality Changes - Where to Edit:

Serial Port Settings:

Want to change default COM port or baud rate? → Edit config.py

```
# config.py
DEFAULT_SERIAL_PORT = "COM30" # Change from COM20
DEFAULT_BAUD_RATE = 9600 # Change from 115200
```

Priority Rules:

Want to change when victims are marked as high priority? → Edit config.py

```
# config.py
RSSI_WEAK_THRESHOLD = -90          # Change from -85
TIME_CRITICAL_THRESHOLD = 20       # Change from 15 minutes
```

Data Handling:

Want to change how data is saved or processed? → Edit modules/data_manager.py

```
# modules/data_manager.py

# Change auto-save interval:
# Find _auto_save() function
# Currently saves every 30 seconds (config.BACKUP_INTERVAL_SECONDS)
```

Analytics Calculations:

Want to change how statistics are calculated? → Edit modules/analytics.py

```
# modules/analytics.py

# Change efficiency grading:
def calculate_rescue_efficiency(self):
    if efficiency_score >= 90: # Change from 80
        grade = "Excellent"
    # ... modify grading logic
```

Files You Should RARELY/NEVER Edit:

app.py (Main Entry Point)

- **Only edit if:** Adding new pages, changing sidebar, or modifying navigation
- **Don't edit for:** Colors, charts, page layouts (edit page files instead)

** ~~utils/helpers.py & utils/validators.py~~

- **Only edit if:** Adding new utility functions or validation rules
- **Don't edit for:** Changing how pages look

** ~~modules/serial_reader.py~~

- **Only edit if:** Changing how serial data is read or parsed

- **Don't touch unless:** You have serial communication issues
-



Quick Reference Table:

What You Want to Change	Edit This File
Dashboard layout/design	pages/1_dashboard.py
Analytics page charts	pages/2_analytics.py
Export page options	pages/3_export.py
Settings page controls	pages/4_settings.py
All colors	config.py
All thresholds	config.py
Map style/markers	modules/map_manager.py
Sidebar/navigation	app.py
Data storage logic	modules/data_manager.py
Statistics formulas	modules/analytics.py
Serial port handling	modules/serial_reader.py



Pro Tips:

1. Always Start with config.py

- Change colors, thresholds, defaults here first
- These changes apply everywhere automatically

2. Page Files = Visual Changes

- Want to change what user sees? → Edit page files
- Want to change how data works? → Edit module files

3. Test Changes Locally

```
# After editing, just refresh the browser
# Streamlit auto-reloads on file changes
```

4. Common Beginner Mistakes:

- ❌ Editing app.py for color changes → Edit config.py instead
 - ❌ Editing modules/ for layout → Edit pages/ instead
 - ❌ Editing utility files for UI → Edit page files instead
-



How to Find What to Edit:

Method 1: Search by Feature

```
# Want to change "Total Detected" metric?  
# Search in all files:  
grep -r "Total Detected" .  
  
# Result shows: pages/1_dashboard.py
```

Method 2: Follow the Logic

1. **User sees something** → It's in a `pages/` file
 2. **Color/number is wrong** → Check `config.py` first
 3. **Data is wrong** → Check `modules/data_manager.py`
 4. **Map looks wrong** → Check `modules/map_manager.py`
-

Example Scenarios:

Scenario 1: "I want the dashboard to show 6 metrics instead of 5"

```
# Edit: pages/1_dashboard.py  
# Find: render_metrics_bar() function  
# Change: col1, col2, col3, col4, col5 = st.columns(5)  
# To: col1, col2, col3, col4, col5, col6 = st.columns(6)  
# Add: with col6: st.metric("New Metric", value)
```

Scenario 2: "I want red to be darker"

```
# Edit: config.py  
# Find: STATUS_COLORS  
# Change: STATUS_STRANDED: "#FF4444"  
# To: STATUS_STRANDED: "#CC0000"
```

Scenario 3: "I want to add a new page"

```
# 1. Create: pages/5_my_new_page.py  
# 2. Edit: app.py → Add "My New Page" to navigation  
# 3. Edit: modules/__init__.py if you need new functions
```

Does this make sense? Any specific change you want to make right now?

🚫 Files You Should RARELY/NEVER Edit:

✗ `app.py` (Main Entry Point)

- Only edit if: Adding new pages, changing sidebar, or modifying navigation
- Don't edit for: Colors, charts, page layouts (edit page files instead)

**✗ `utils/helpers.py` & `utils/validators.py`

- Only edit if: Adding new utility functions or validation rules
- Don't edit for: Changing how pages look

**✗ `modules/serial_reader.py`

- Only edit if: Changing how serial data is read or parsed
- Don't touch unless: You have serial communication issues

📋 Quick Reference Table:

What You Want to Change	Edit This File
Dashboard layout/design	<code>pages/1_dashboard.py</code>
Analytics page charts	<code>pages/2_analytics.py</code>
Export page options	<code>pages/3_export.py</code>
Settings page controls	<code>pages/4_settings.py</code>
All colors	<code>config.py</code>
All thresholds	<code>config.py</code>
Map style/markers	<code>modules/map_manager.py</code>
Sidebar/navigation	<code>app.py</code>
Data storage logic	<code>modules/data_manager.py</code>
Statistics formulas	<code>modules/analytics.py</code>
Serial port handling	<code>modules/serial_reader.py</code>

💡 Pro Tips:

1. Always Start with `config.py`

- Change colors, thresholds, defaults here first
- These changes apply everywhere automatically

2. Page Files = Visual Changes

- Want to change what user sees? → Edit page files

- Want to change how data works? → Edit module files