

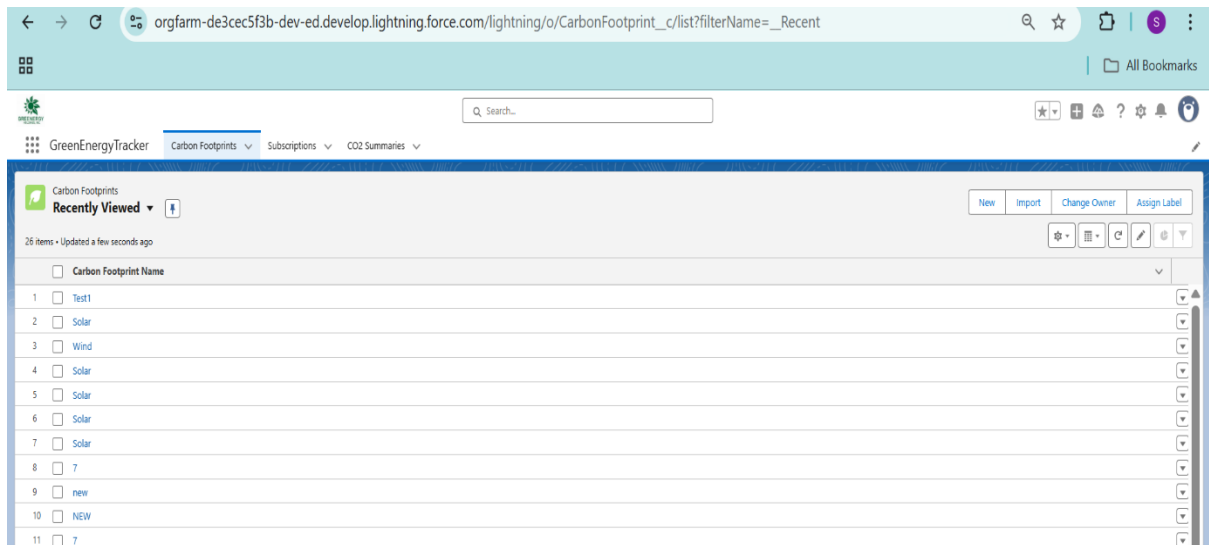
Phase 6: User Interface Development

Step 1: Lightning App Builder

- **Location:** Setup → Lightning App Builder → New.
- **Purpose:** Create **custom apps** combining pages, tabs, and components.
- **Steps:**
 - Click **New Lightning App**.
 - Enter **App Name:** GreenEnergyTracker.
 - Select **App Options:** Navigation style (Standard or Console), Branding (logo, colors), Utility Bar (optional).
 - Add **Tabs** (Subscriptions, Carbon Footprints, Reports, Dashboard).
 - Set **Visibility & Permissions**.

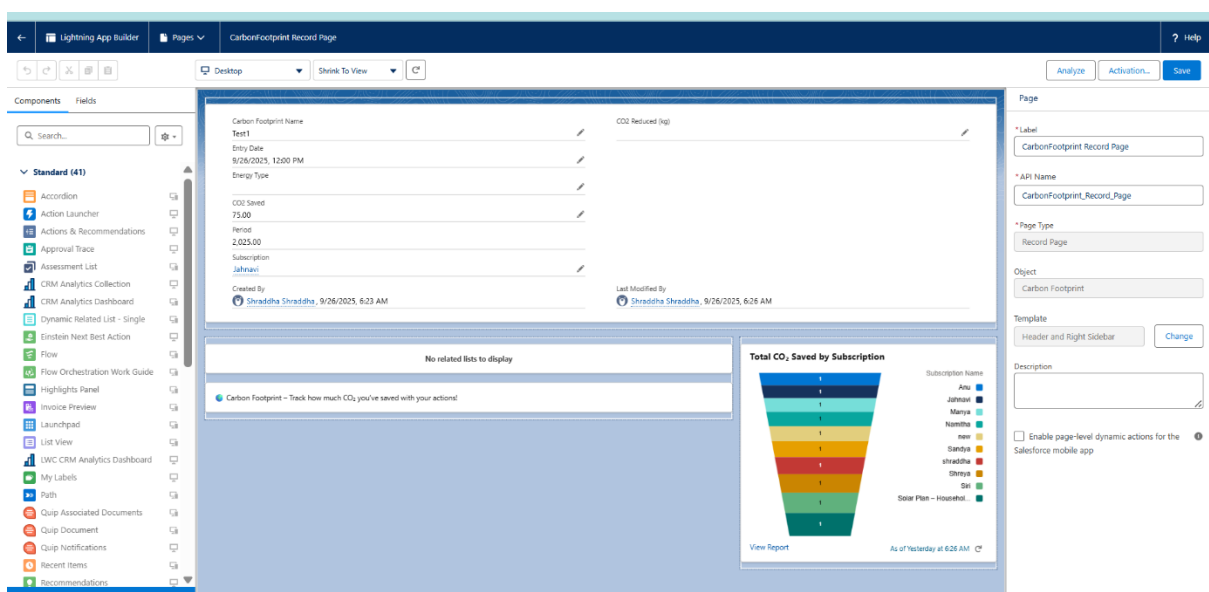
The screenshot shows the 'App Details & Branding' configuration screen in the Lightning App Builder. The left sidebar contains 'App Settings' with sub-items: 'App Details & Branding' (selected), 'App Options', 'Utility Items (Desktop Only)', 'Navigation Items', and 'User Profiles'. The main content area is divided into two sections: 'App Details' and 'App Branding'. In 'App Details', there are input fields for 'App Name' (filled with 'GreenEnergyTracker'), 'Developer Name' (filled with 'GreenEnergyTracker_3'), and a 'Description' field. In 'App Branding', there is an 'Image' field showing a logo, a 'Primary Color Hex Value' dropdown set to '#0070C2', and an 'Org Theme Options' checkbox which is unchecked. Below these is an 'App Launcher Preview' showing the app's icon and name.

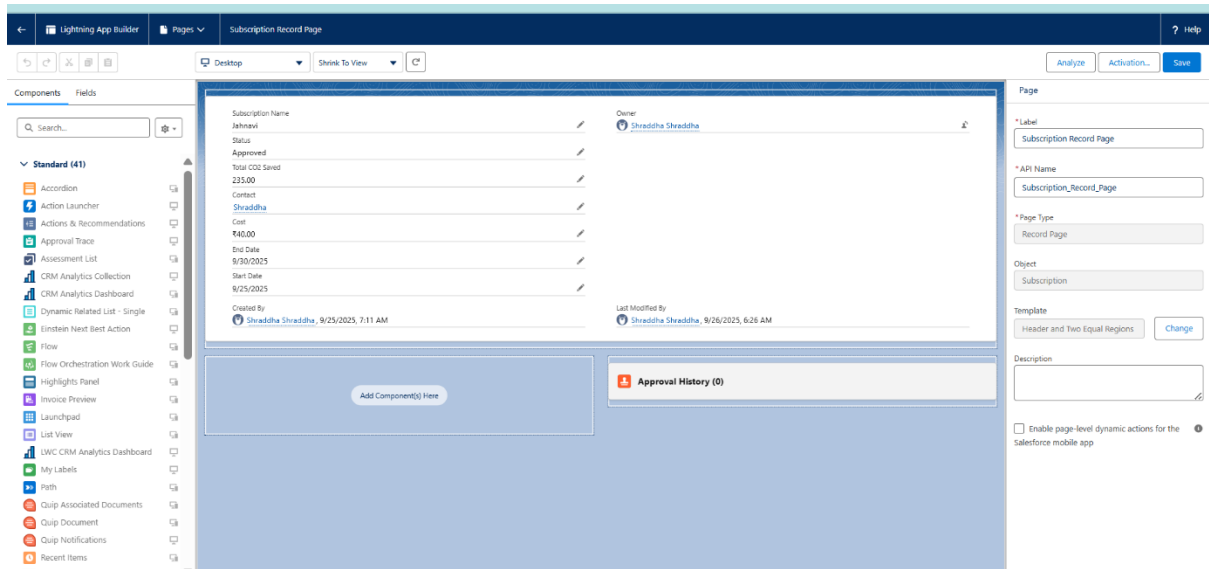
The screenshot shows the 'Navigation Items' configuration screen in the Lightning App Builder. The left sidebar is the same as the previous screen, with 'Navigation Items' selected. The main content area is titled 'Navigation Items' and includes a sub-header: 'Choose the items to include in the app, and arrange the order in which they appear. Users can personalize the navigation to add or move items, but users can't remove or rename the items that you add. Some navigation items are available only for phone or only for desktop. These items are dropped from the navigation bar when the app is viewed in a format that the item doesn't support.' Below this, there are two panels: 'Available Items' and 'Selected Items'. The 'Available Items' panel has a search bar and a list of items including Accounts, Activation Targets, Activations, All Sites, Alternative Payment Methods, Analytics, App Launcher, Appointment Categories, Appointment Invitations, Approval Requests, Approval Submission Details, Approval Submissions, Approval Work Items, Asset Action Sources, Asset Actions, and Asset State Periods. The 'Selected Items' panel shows three items: Carbon Footprints, Subscriptions, and CO2 Summaries, each with a green checkmark icon. Arrows between the panels allow for moving items back and forth.



Step 2: Record Pages

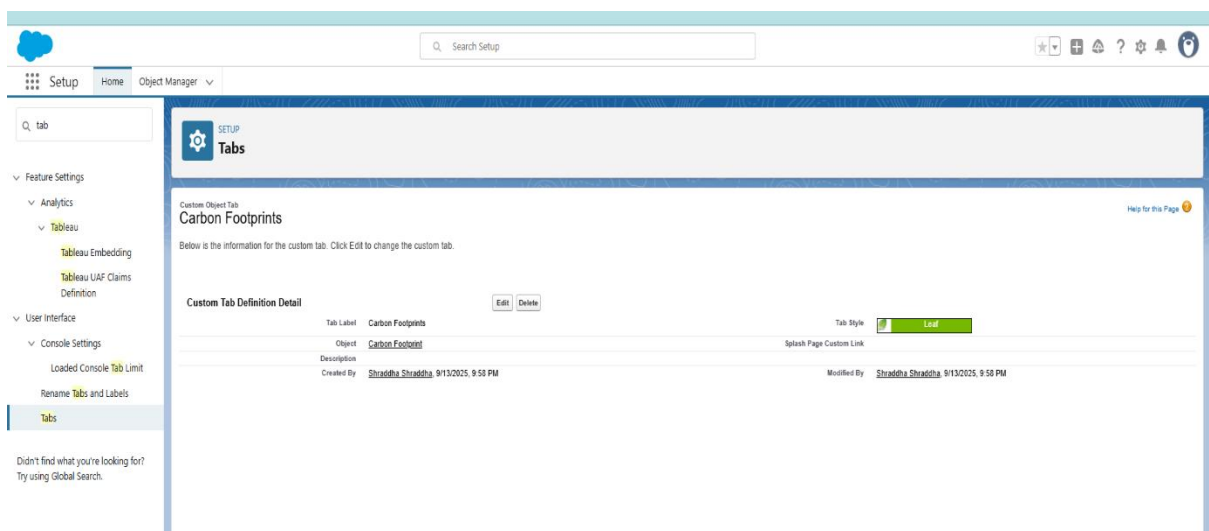
- **Purpose:** Customize how individual records are displayed in Salesforce Lightning.
- **Steps:**
 - Go to Object Manager → Subscription__c → Lightning Record Pages → New.
 - Choose Standard Page or clone existing.
 - Drag components like Tabs, Related Lists, Reports, or LWC onto the page.
 - Set component visibility based on profiles or permissions.
 - Save & activate as default for Org, App, or Profile.

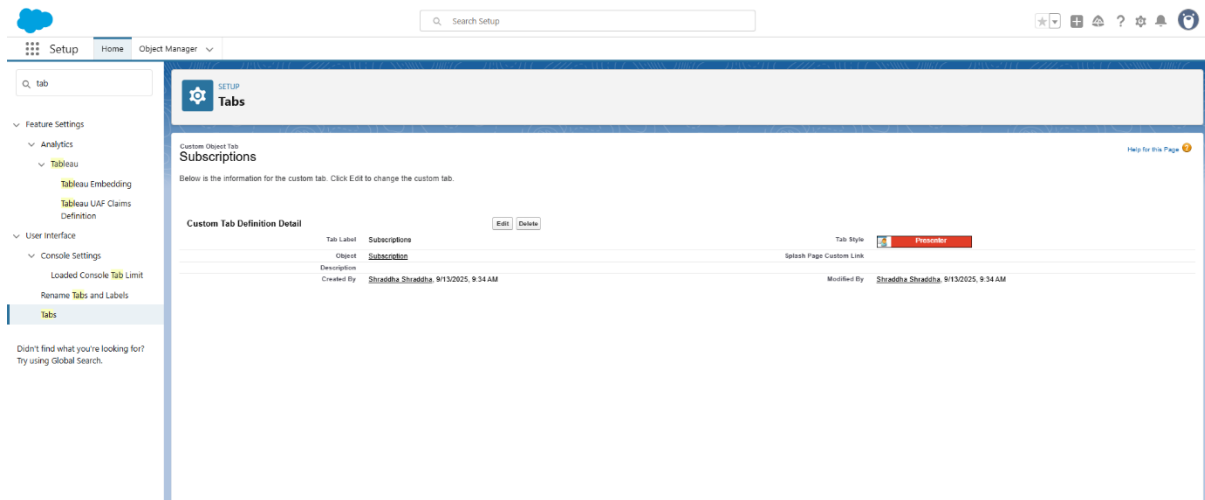




Step 3: Tabs

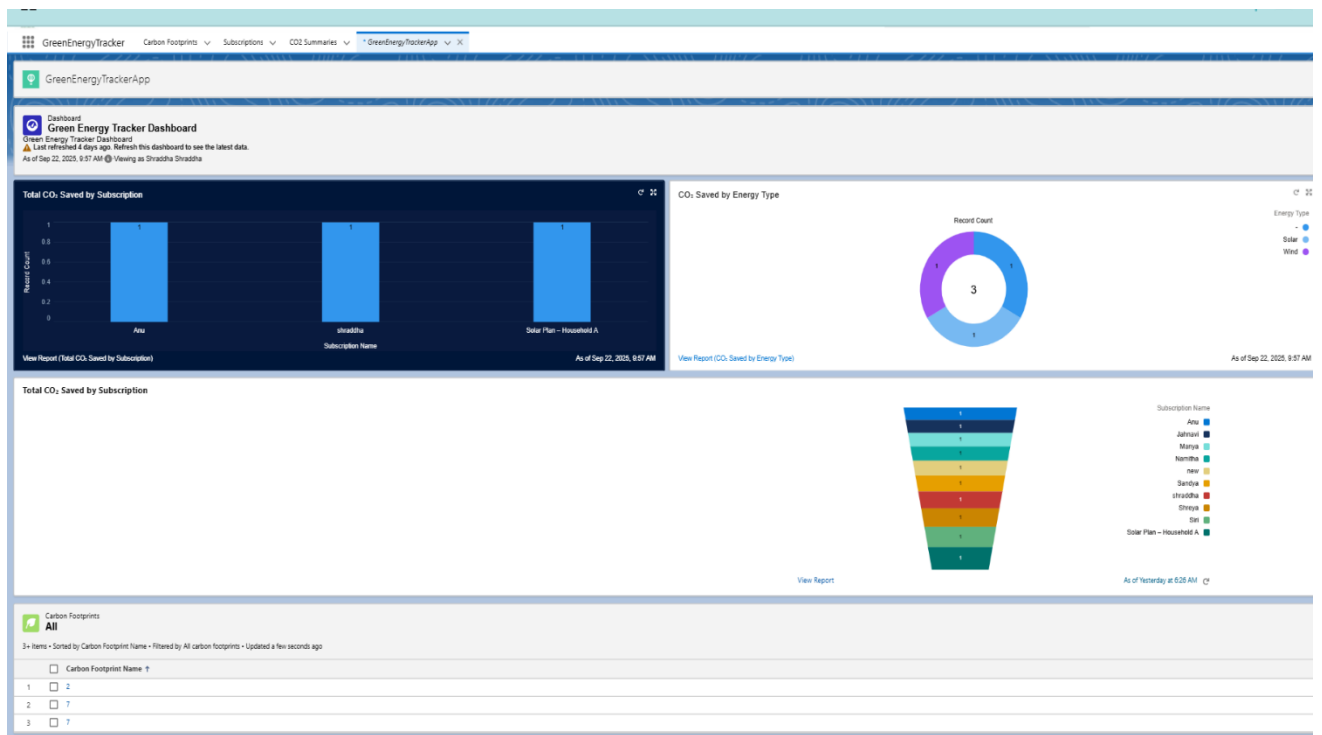
- **Purpose:** Organize the navigation within your app.
- **Steps:**
 1. Go to **Setup** → **Tabs** → **New Tab**.
 2. Select object (e.g., Subscription__c) or Visualforce/LWC tab.
 3. Name tab, choose icon, save.
 4. Add tab to **GreenEnergyTracker app** in App Manager.





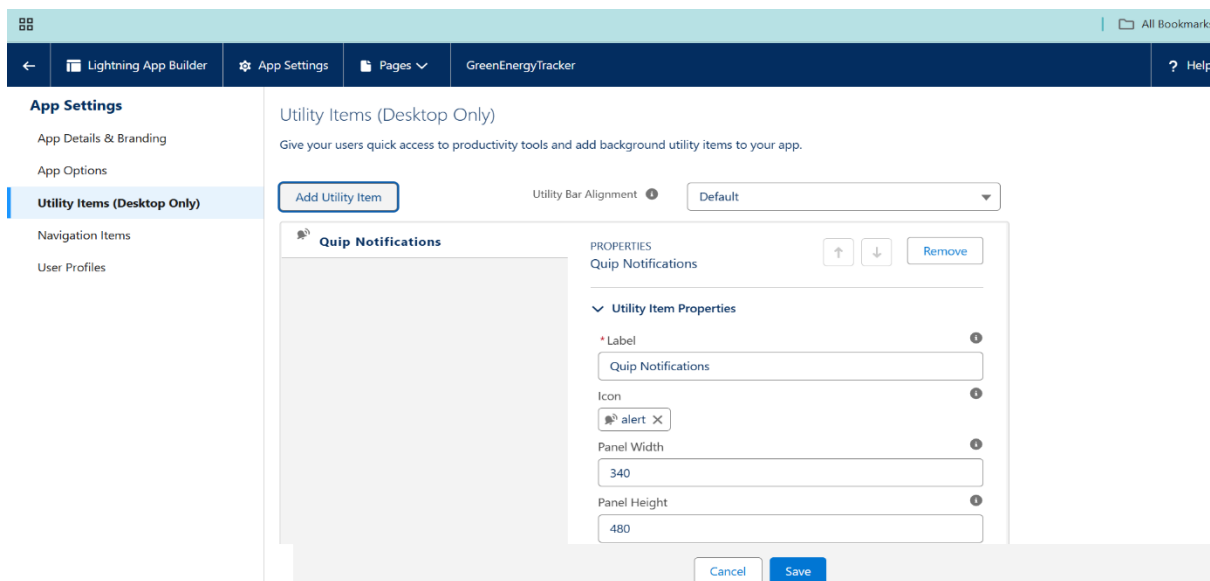
Step 4: App Page Layouts

- **Purpose:** Customize what users see on login.
- **Steps:**
 1. Go to **Setup** → **Lightning App Builder** → **App Page** → **New**.
 2. Add components like **Dashboards, Reports, Charts, or LWC components**.
 3. Assign the page to **profiles** (Admin, User).
 4. Activate.



Step 5: Utility Bar

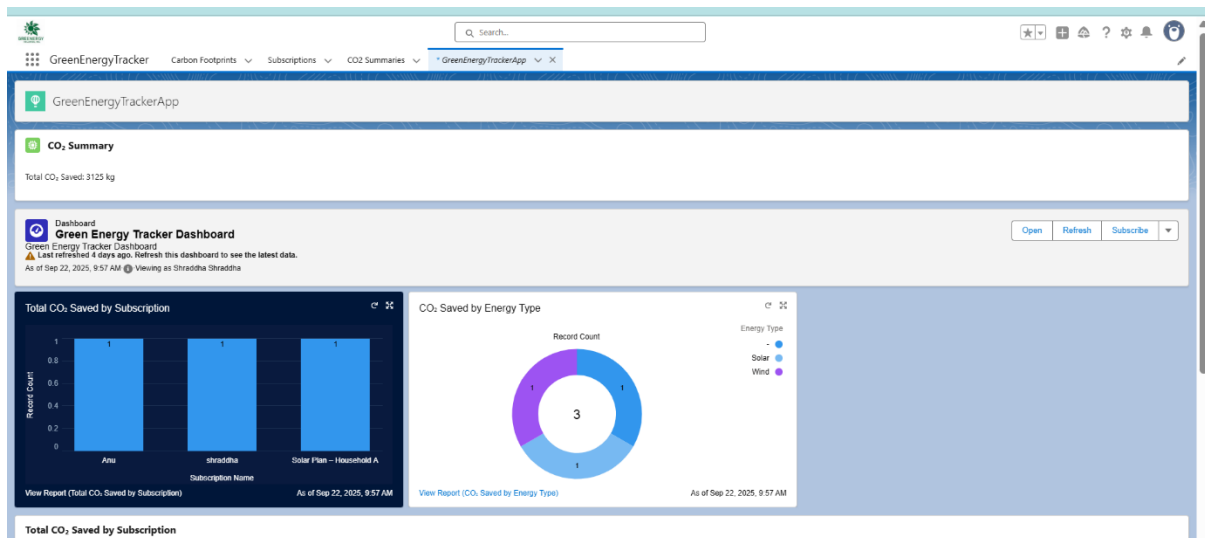
- **Purpose:** Quick-access tools at the bottom of your app.
- **Steps:**
 1. Open **App Manager** → **Edit GreenEnergyTracker** → **Utility Bar** → **Add Utility Items**.
 2. Examples: **Notifications**, **CO₂ Summary**, **Help**, **Chatbot**.
 3. Configure icons, labels, and actions.



Step 6: Lightning Web Components (LWC)

- **Purpose:** Create dynamic, reusable UI components.
- **Steps:**
 1. Use VS Code with Salesforce Extensions.
 2. Create new LWC:

```
sfdx force:lightning:component:create --type lwc --componentname CO2Summary --outputdir force-app/main/default/lwc
```
 3. Structure:
 - HTML: Template for UI.
 - JS: Component logic.
 - Meta.xml: Visibility, targets, and API version.
 - CSS: Optional styling.



Step 7: Apex Integration with LWC

- **Purpose:** Fetch or update Salesforce data dynamically.
- **Steps:**
 1. Create **Apex class** with `@AuraEnabled` methods.

```
public with sharing class SubscriptionController {
    @AuraEnabled(cacheable=true)
    public static List<Subscription__c> getAllSubscriptions(){
        return [SELECT Id, Name, Total_CO2__c FROM Subscription__c];
    }
}
```

Step 8: Events in LWC

- **Purpose:** Communicate between components.
- **Types:**
 1. **Custom Events:** Child → Parent communication.
 2. **Lightning Message Service:** Publish-subscribe across unrelated components.

Example:

```
const selectedEvent = new CustomEvent('subscriptionselected', { detail: subId });
this.dispatchEvent(selectedEvent);
```

Step 11: Navigation Service

- **Purpose:** Navigate programmatically between pages or records.
- **Steps:**

1.Import NavigationMixin:

```
import { NavigationMixin } from 'lightning/navigation';  
export default class NavigateExample extends NavigationMixin(LightningElement) {}
```

2.Navigate to a record page:

```
this[NavigationMixin.Navigate]({  
  type: 'standard__recordPage',  
  attributes: { recordId: sub.Id, objectApiName: 'Subscription__c', actionName: 'view' }  
});
```