**Step 1: Add an "Edit Tree" Mode**

Before adding full user-driven functionality, let's create the infrastructure for tree editing. This means adding buttons or menu options in the app that allow users to interact with the tree (e.g., adding or editing nodes).

**Task:**

* Add a button labeled "Edit Tree" to toggle between "view mode" and "edit mode."
* When in "edit mode," allow users to select a node and perform basic actions like adding or renaming a child.

**Step 2: Allow Adding New Nodes**

Once "Edit Tree" mode is active:

* Add a user interface component (like an input box and button) for the user to add a child to the selected node.

**Task:**

* Implement the ability to select a node and add a new child to it dynamically.
* Example: If the user selects "Amniota," they should be able to add children like "Synapsida" or "Sauropsida."

**Step 3: Allow Editing Existing Nodes**

Allow the user to rename nodes or update their data.

**Task:**

* Create a simple "Rename" feature that enables users to update the name of a node.

**Step 4: Allow Deleting Nodes**

Users should be able to delete nodes, but this will require careful handling:

* Deleting a node should also delete all of its descendants.
* You must ensure that users cannot delete the root node.

**Task:**

* Add a "Delete Node" feature with a confirmation dialog.

**Step 5: Add Ability to Insert Nodes Between Parent and Children**

This is more advanced. Users should be able to insert a node between a parent and its current children.

**Task:**

* Implement a way to select a parent, then add a child that becomes the parent of the current children.

**Step 6: Save and Load Trees**

Provide functionality for users to save their trees to a file (e.g., JSON) and load saved trees for editing.

**Task:**

* Create a "Save" button to save the tree structure.
* Create a "Load" button to load a tree into the app.

**Step 7: Finalize and Polish**

After implementing the above functionality:

* Refine the user interface to make it intuitive.
* Add error handling (e.g., ensure nodes cannot have duplicate children).
* Allow users to switch between multiple saved trees.

**Where to Start?**

**Start with Step 1: Adding "Edit Tree" Mode.**

* Add a button labeled "Edit Tree" in the app.
* When clicked, show a label or placeholder that says, "Click a node to start editing."
* In this mode, users can select a node from the tree, and we can build additional editing options in subsequent steps.

Would you like to begin with Step 1? If so, I can help you implement the "Edit Tree" mode as the foundation for further functionality.