30-Day PyQt6 GUI Application Build Checklist

Day	Focus Area	What You Practice Today	Key Task (■ Check when done)	Notes
1	Environment Setup	Create virtual environment, install PyQt6, verify imports	☐ App launches with empty QMainWindow	
2	Layouts & Widgets	Understand QLabel, QLineEdit, QPushButton, and layout resizing	☐ Responsive layout using QHBoxLayout/QVBoxLayout	
3	Qt Designer	Design UI visually using drag-and-drop widgets	☐ Functional form saved as .ui file	
4	UI Loading	Load Qt Designer UI into Python using uic	☐ UI loads without runtime errors	
5	Signals & Slots	Connect UI events to Python logic	☐ Button click triggers slot	
6	Main Window	Design application main interface	☐ Search + results + action buttons wired	
7	Dialogs	Create modal dialogs for user input	☐ Create/Edit dialog opens and closes correctly	
8	Model/View	Separate data from UI using Qt models	☐ QTableView displays model data	
9	Create User	Pass dialog data back to model	☐ New user row added dynamically	
10	Edit User	Detect selected row and update data	☐ Existing user edited successfully	
11	Mock Backend	Abstract data logic from UI	☐ UI calls backend methods only	
12	SQLite Backend	Persist application data locally	☐ Users persist after restart	
13	Search	Implement filtering/search logic	☐ Search bar filters results	
14	Menus & Toolbars	Add menu bar and toolbar actions	☐ Menu/toolbar actions functional	
15	LDAP Concepts	Understand AD/LDAP read operations	☐ Test LDAP search script runs	
16	LDAP Integration	Integrate live AD backend (read-only)	☐ Table populated from AD	
17	AD Write Ops	Create and modify AD users safely	☐ Write ops tested in lab only	
18	Threading	Move long tasks off UI thread	☐ UI remains responsive during LDAP calls	
19	User Feedback	Provide progress and status updates	☐ Status bar messages displayed	
20	Error Handling	Handle failures gracefully	☐ Errors logged and user alerted	
21	Settings	Persist window and app settings	☐ Window restores previous state	
22	Styling	Customize look with QSS	☐ Theme applied consistently	
23	Icons & UX	Improve usability and clarity	☐ Icons added to actions	
24	Custom Widgets	Build reusable UI components	☐ Custom widget reused in app	
25	Testing	Validate behavior with pytest-qt	☐ Automated tests pass	
26	Packaging	Bundle app using PyInstaller	☐ Standalone executable created	
27	Validation	Test app on clean system	☐ Runs without Python installed	
28	Installer	Create professional installer	☐ Installer completes successfully	
29	Cleanup	Refactor and document code	☐ README and docstrings complete	
30	Review	Reflect and plan next project	☐ Next project defined	