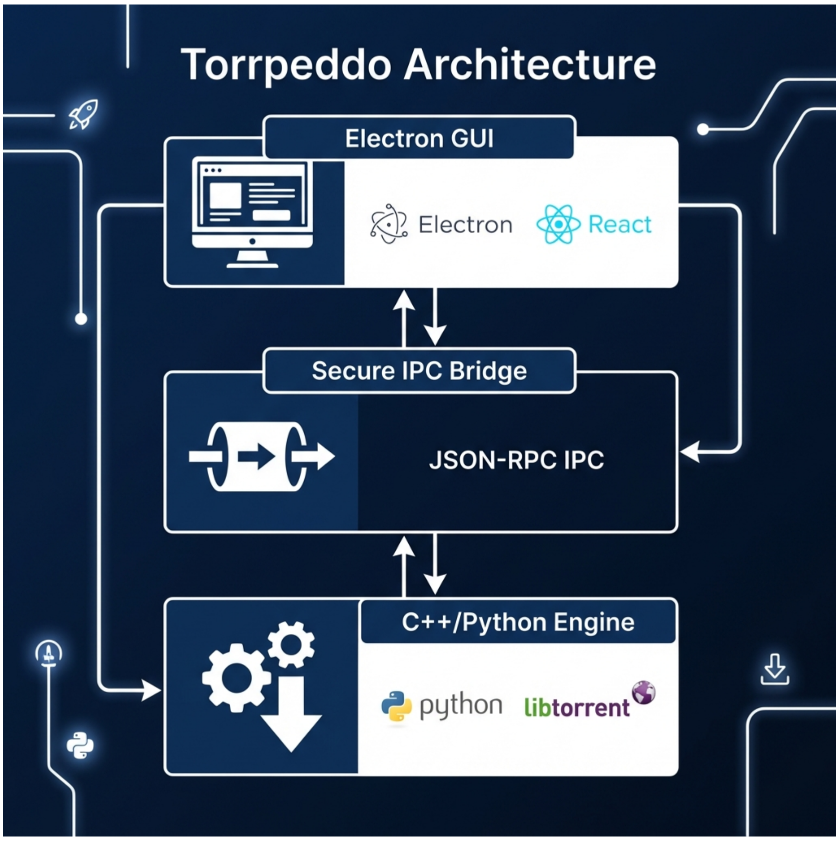
TORRPEDDO PROJECT BOOK



Executive Summary

Torrpeddo is an industrial-grade, premium torrent client designed for the modern desktop. Built primarily with Python and the Electron framework, Torrpeddo leverages the powerful `libtorrent` suite to offer a seamless, high-performance experience that bridges the gap between complex network protocols and professional user interfaces.

Architectural Deep Dive

Torrpeddo follows a decoupled architectural pattern, separating the presentation layer from the core logic and network engine. This is achieved through three primary layers:

1. Frontend: Electron Framework

What is Electron?

Electron is an open-source fram
allows developers to build cross
web technologies: HTML, CSS,
Chromium rendering engine (fo
system-level access).

Benefits for Torrpeddo:

- Visual Excellence: Leveragin
web components to create a "W

2. The Bridge: IPC (Inter-Process Communication)

What is IPC?

IPC, or Inter-Process Communi
different processes to share dat
Torrpeddo, we use a custom IP
frontend with the Python backe

Implementation: Secure JSON-RPC

Communication is handled via a
stdin/stdout channels.

Why this approach?

- Decoupling: The engine can
replaced without touching the U

3. Backend Engine: Python & libtorrent

The Core: libtorrent with Python Bindings

At the heart of Torrpeddo is `libt implementation. While the unde high-performance C++, Torrpec bindings for rapid development bridge logic.

Multi-threaded Performance:

- Engine Level: The `libtorrent` thread pool for disk I/O, network allows for parallel processing of simultaneously.

Development Process & Methodology

The Torrpeddo project followed a "Platform-First" methodology:

1. Language Choice: Python w support and ease of integration
