

Module **p2p_interface**

Functions

```
def main()
```

Classes

```
class UserInterface (node)
```

Command-Line User Interface for P2P File Sharing System

Provides an interactive menu-driven interface for users to interact with the P2P node. Handles user input, displays results, and manages the interaction flow between the user and the P2P network.

Attributes

node : P2P

Reference to the P2P node instance

last_search_results : list

Cache of most recent search results for downloads

Initialize the User Interface

Args

node : P2P

P2P node instance to interface with

Methods

```
def connect_to_peer(self)
```

Handle manual peer connection interaction Prompts user for peer IP address and port, then attempts to connect. Provides default value for IP address and validates port number.

On success: Peer is added to node's peer list Triggers peer discovery (node learns about peer's known peers)

```
def download_file(self)
```

Handle file download interaction from cached search results

Guides user through downloading a file from previous search results: 1. Select file from search results 2. Select peer to download from 3. Execute download 4. Display success/failure message

Requirements

- Must run `search_files()` first to populate `self.last_search_results`
- Selected file must have at least one available peer

```
def run(self)
```

Main UI loop - runs until user exits or interrupts

Continuously displays menu, processes user input, and executes corresponding actions. Handles both normal exit (option 7) and keyboard interrupt (Ctrl+C).

- Runs while `self.node.running` is True
- Exits on option 7 or KeyboardInterrupt
- Gracefully shuts down node on exit

Exception Handling: - KeyboardInterrupt: Catches Ctrl+C and shuts down gracefully - Invalid input: Displays error message and continues

```
def search_files(self)
```

Handle file search interaction

Prompts user for search query, performs network-wide search, caches results for later download, and displays formatted results.

Updates `self.last_search_results` for use in `download_file()` Prints search results to console

```
def show_menu(self)
```

Display the main menu options to the user

Functions

`main`

Classes

ManagerInterface

connect_to_peer

download_file

run

search_files

show_menu