

Chat UI Dokumentation

Generated by Doxygen 1.14.0

1 Namespace Index	1
1.1 Namespace List	1
2 Class Index	3
2.1 Class List	3
3 Namespace Documentation	5
3.1 chat_ui Namespace Reference	5
3.1.1 Detailed Description	5
4 Class Documentation	7
4.1 chat_ui.ChatClientUI Class Reference	7
4.1.1 Detailed Description	7
4.1.2 Constructor & Destructor Documentation	7
4.1.2.1 __init__()	7
4.1.3 Member Function Documentation	8
4.1.3.1 run()	8
4.1.4 Member Data Documentation	8
4.1.4.1 DEFAULT_CONFIG	8
Index	9

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

chat_ui	5
-----------------------------------	---

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

chat_ui.ChatClientUI	Diese Klasse stellt die textbasierte Benutzeroberfläche und Netzwerklogik bereit	7
--------------------------------------	--	---

Chapter 3

Namespace Documentation

3.1 chat_ui Namespace Reference

Classes

- class [ChatClientUI](#)

Diese Klasse stellt die textbasierte Benutzeroberfläche und Netzwerklogik bereit.

Variables

- `ui = ChatClientUI()`

3.1.1 Detailed Description

```
@file chat_ui.py
@brief Kommandozeilen-Oberfläche und Logik für den Chat-Client.
```

Dieses Modul verwaltet die Benutzereingaben, Konfigurationen, Nachrichtenaustausch und Discovery-Kommunikation via Queue.

Chapter 4

Class Documentation

4.1 chat_ui.ChatClientUI Class Reference

Diese Klasse stellt die textbasierte Benutzeroberfläche und Netzwerklogik bereit.

Public Member Functions

- `__init__` (self, config_path="config.toml")
Initialisiert das UI und lädt Konfiguration.
- `load_config` (self)
- `save_config` (self, config)
- `change_config` (self)
Ändert die Konfiguration über Benutzereingabe (außer whoisport).
- `run` (self, Queue ui_to_net, Queue net_to_ui)
Startet das textbasierte UI: verarbeitet alle Eingaben und zeigt Netzwerknachrichten an.

Public Attributes

- `CONFIG_FILE` = config_path
- dict `DEFAULT_CONFIG`
- `config` = dict(self.DEFAULT_CONFIG)
- dict `peers` = {}

4.1.1 Detailed Description

Diese Klasse stellt die textbasierte Benutzeroberfläche und Netzwerklogik bereit.

Sie verarbeitet Eingaben, lädt und speichert Konfigurationen, und kommuniziert mit dem Netzwerkprozess über Queues.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 __init__()

```
chat_ui.ChatClientUI.__init__ (  
    self,  
    config_path = "config.toml")
```

Initialisiert das UI und lädt Konfiguration.

Parameters

<i>config_path</i>	Pfad zur TOML-Konfigurationsdatei.
--------------------	------------------------------------

4.1.3 Member Function Documentation

4.1.3.1 run()

```
chat_ui.ChatClientUI.run (
    self,
    Queue ui_to_net,
    Queue net_to_ui)
```

Startet das textbasierte UI: verarbeitet alle Eingaben und zeigt Netzwerknachrichten an.

Parameters

<i>ui_to_net</i>	Queue zum Senden von Nachrichten.
<i>net_to↔_ui</i>	Queue zum Empfangen von Netzwerkereignissen.

Startet die Chat-Schleife:

- Anzeige eingehender Nachrichten und Discovery-Events aus net_to_ui
- Befehle: /help, /who, /msg <Handle> <Text>, /img <Handle> <Bildpfad>, /config, /quit
- Nachrichten ohne '/' werden als Broadcast via ui_to_net gesendet

4.1.4 Member Data Documentation

4.1.4.1 DEFAULT_CONFIG

```
chat_ui.ChatClientUI.DEFAULT_CONFIG
```

Initial value:

```
= {
    "handle": None,
    "port": 5000,
    "whoisport": 4000,
    "autoreply": "Ich bin gerade nicht da.",
    "imagepath": "./images"
```

The documentation for this class was generated from the following file:

- chat_ui.py

Index

- `__init__`
 - `chat_ui.ChatClientUI`, [7](#)
- `chat_ui`, [5](#)
- `chat_ui.ChatClientUI`, [7](#)
 - `__init__`, [7](#)
 - `DEFAULT_CONFIG`, [8](#)
 - `run`, [8](#)
- `DEFAULT_CONFIG`
 - `chat_ui.ChatClientUI`, [8](#)
- `run`
 - `chat_ui.ChatClientUI`, [8](#)