**ספריות ומודולים ששומשו בפרויקט:**

|  |  |  |
| --- | --- | --- |
| שם המודול / ספרייה | סיבת השימוש | סיבת הבחירה |
| socket | ניהול תקשורת | ספרייה שמובנת בתוך Python, נוחה לשימוש, אני מכיר אותה היטב וידועה לניהול תקשורת. |
| os | אינטראקציה עם מערכת ההפעלה | מאפשר גישה לשירותי מערכת ההפעלה שמשומשים בפרויקט. |
| logging | ניהול ורישום לוגים | כלי מובנה לניהול לוגים בצורה מסודרת, גמיש וניתן להתאמה אישית בקלות. |
| time | ניהול זמן ושעון (fps) | מספק פונקציות לעבודה עם זמן, כולל מדידת ביצועים, השהיות, וקבלת תאריך נוכחי. |
| select | ניהול לקוחות מרובים  ב-RC | מאפשר ניהול לקוחות מרובים בשרתים בצורה פשוטה ונוחה |
| re | עיבוד ביטויים רגולריים | משמש לחיפוש, התאמה ומניפולציה של טקסטים עם ביטויים רגולריים, תומך בגמישות מירבית. |
| struct | עבודה עם מבני נתונים בינאריים | מאפשר המרה של נתונים בין פורמט בינארי לפורמט ש-Python מבין, חיוני בתקשורת והמרת נתונים. |
| zlib | דחיסת נתונים של וידאו ואודיו ב-TC | נוחה ופשוטה לשימוש ובנויה בתוך python |
| numpy | עיבוד וידאו | ספרייה נוחה ומוכרת שנותנת תמיכה  ב-opencv. |
| cv2 | משומשת לקליטת וידאו מה-TC ולקריאת הוידאו ב-Server | מוכרת בעולם, נוחה לשימוש ויש לי ניסיון רב איתה מפרויקטים קודמים |

|  |  |  |
| --- | --- | --- |
| שם המודול / ספרייה | סיבת השימוש | סיבת הבחירה |
| pyaudio | קליטת אודיו מה-TC | מאפשר ניהול והקלטה של אודיו, מתאים לעיבוד קובצי קול ולזיהוי דיבור. |
| math | פונקציות מתמטיות מתקדמות | מובנה בתוך python ונוח לשימוש. |
| random | יצירת מספרים רנדומליים ssrc (מספר מזהה בפרוטוקול RTP) | מובנה בתוך python, נוח וקל לשימוש. |
| thread | ניהול Threads ב-Server | מאפשר יצירה ועובדה של תהליכים בתוך התוכנה. |
| multiprocess | ניהול תהליכים ב-TC שליחת אודיו ושליחת וידאו | מספק תמיכה לריבוי תהליכים, משפר ביצועים במערכות עם עיבוד כבד ע"י הפעלת מספר ליבות של המעבד. |
| ssl | http and websocket encryption | ספרייה שהדפדפן יודע לעבוד עם המידע שהיא שולחת אליו ומטפלת בכל העניין אוטומטית. |
| dtls | ספרייה משומשת להצפנת זרמי UDP | לפי מחקר אינטרנטי הספרייה הכי מומלצת להצפנת UDP |
| sqlite3 | ניהול DB | ספרייה שאני משתמש כבר הרבה שנים ומכילה פרמטריזציה מובנת |
| vosk | תמלול | נותן אפשרות לשימוש אופליין של מודלים לוקליים לתמלול טקסט |

**מודולים ומחלקות שאני פיתחתי:**

**AudioReceiveHandler**

שימוש: Handles real-time UDP audio reception with delta time synchronization.

תכונות: אין

|  |  |
| --- | --- |
| Docs | Method name |
| Receives RTP audio packets and plays audio in real-time using delta time for better timing. :return: None | recv\_audio |

**VideoTransmission**

שימוש: Bundles video handlers and handles transmission

תכונות:

* video\_capture\_source – camera port
* port – port of udp connection
* payload\_type – rtp payload type of video
* rtp\_handle – instance of RTPHandler
* udp\_handle – instance of UDPClientHandler
* logger – class logger

|  |  |
| --- | --- |
| Docs | Method name |
| transmit video using rtp :return: None | transmit\_video |

**VideoCapture**

שימוש: This class handles the video capture and handling.

תכונות:

* logger – class logger
* source – camera source port
* capture – cv2 capture instance

|  |  |
| --- | --- |
| Docs | Method name |
| Retrieves a single frame from the video source.  :return tuple: (success, frame), where `success` is a boolean indicating if the frame was read successfully, and `frame` is the captured frame. | get\_frame |
| Releases the video capture resource.  :return: None | release |
| Ensures resources are released when the instance is deleted.  :return: None | \_\_del\_\_ |

**UDPClientHandler**

שימוש:

This class represents the video transmitting client using UDP.  
It initializes an RTPHandler instance to handle RTP packet creation  
and uses a UDP socket for sending packets.

תכונות:

* server\_address – server host
* server\_port – udp binding port
* sock – udp socket
* logger – class logger

|  |  |
| --- | --- |
| Docs | Method name |
| Captures video frames, creates RTP packets, and transmits them to the server via UDP. :param packets: (list[bytes]) packets to send :return bool: whether operation was successful | send\_packets |

**RTPHandler**

שימוש:

This class handles the RTP protocol, including creating and parsing RTP packets.

תכונות:

* payload\_type – rtp payload type
* ssrc – rtp ssrc
* sesquence\_number – rtp sequence number (init 0)
* logger – class logger

|  |  |
| --- | --- |
| Docs | Method name |
| Constructs the RTP header.  :param extension\_data: :param marker: (int) The marker bit. :param csrcs: (list[int]) List of contributing source identifiers. :return: bytes: The RTP header as a byte string. | build\_header |
| Creates an RTP packet by combining the header and payload.  :param payload: (bytes) payload to put in the rtp packet :param csrcs: (list[int]) List of contributing source identifiers. :return: bytes: The complete RTP packet. | create\_packets |
| Returns the SSRC for the RTP stream.  :return: int: The SSRC value. | get\_ssrc |
| updates timestamp :return: None | \_update\_timestamp |

**AudioTransmissionHandler**

שימוש:

Handles real-time audio transmission via RTP over UDP with delta time synchronization.

תכונות:

* logger – class logger
* audio\_capture – pyaudio audio capture instance
* rtp\_handler – rtpHandler instance
* udp\_handler – udpClientHandler instance

|  |  |
| --- | --- |
| Docs | Method name |
| Begins live audio transmission with delta time control for better synchronization. :return: None | start\_streaming |
| Cleans up resources after transmission stops. :return: None | stop\_streaming |

**AudioCapture**

שימוש:

Handles real-time audio capture and streaming

תכונות:

* rate – sample rate (const)
* channels – channels (const)
* chunk\_size – hunk size (const)
* logger -class logger
* audio – PyAudio instance
* stream – opening input stream in pyaudio

|  |  |
| --- | --- |
| Docs | Method name |
| Captures and returns a single chunk of audio data.  :return bytes: Raw audio data. | get\_audio\_chunk |
| Cleans up resources. :return: None | release |

**VideoReceiveHandler**

שימוש:

Handles real-time UDP video reception

תכונות: אין

|  |  |
| --- | --- |
| Docs | Method name |
| Receives video and presents it :return: None | recv\_video |

**UDPServerHandler**

שימוש:

This class represents the video receiving server using UDP.  
It initializes a UDP socket to listen for incoming packets and  
uses RTPacketDecoder to decode the received RTP packets.

תכונות:

* bind\_address – (const) the server address
* bind\_port – port to listen on
* sock – udp socket
* \_uncompleted\_frame\_packets – dictionary used to assemble messages
* logger – class logger

|  |  |
| --- | --- |
| Docs | Method name |
| Will receive a packet using a UDP connection :param buffer: (int) the buffer to receive :return bytes | None: The received packet if one was received | \_receive\_packet |
| Assembles / Discards frames :param seq\_start: (int) fragment sequence start :param seq\_end: (int) fragment sequence end :return bytes | None: bytes if a frame was assembled, None if frame was dropped | assemble\_rtp\_message |
| Receives RTP packets from the client & decodes them. :return bytes or None: If a message/packet is received then the payload will be returned. | receive\_rtp\_message |

**RTPPacketDecoder**

שימוש:

Decodes RTP packets into header fields and payload.

תכונות:

* packet – the raw packet
* version – rtp version
* padding – rtp padding bit
* extension – rtp extension bit
* csrc\_count – rtp csrc count
* marker – rtp marker bit
* payload\_type – rtp payload type
* sequence\_number – rtp sequence number
* timestamp – rtp timestamp
* ssrc – rtp ssrc
* csrcs – rtp csrcs list
* extension\_data – holds the data of the extension if there is one
* payload – rtp payload
* logger – class logger

|  |  |
| --- | --- |
| Docs | Method name |
| Decodes the RTP packet, extracting its header and payload. :return: None | decode\_packet |
| Returns the RTP header fields as a dictionary.  :return: dict: Header fields with their values. | get\_header\_info |
| Returns the payload from the RTP packet.  :return: bytes: The raw payload data. | get\_payload |

**Transcriptor**

מימוש:

Handles live speech recognition and captioning using AudioCapture.

תכונות:

* model – vosk model instance
* recognizer – vosk voice recognizer
* logger – class logger
* audio\_capture – audioCapture instance

|  |  |
| --- | --- |
| Method | Docs |
| transcribe\_audio | Continuously captures and transcribes live audio.  :return: None |

**HttpParser**

מימוש:

Class to make http requests usable in code more easily

תכונות:

* METHOD – http method (get / post)
* URI – uri of the request
* HTTP\_VERSION – version from the request
* QUERY\_PARAMS – query parameters given in the request
* HEADERS – headers given in the request
* BODY – body of the request
* COOKIES – cookies given in the request

|  |  |
| --- | --- |
| Method | Docs |
| \_\_header\_parser | Extracts the headers from the request :return dict[bytes]: {header: value} |
| \_\_body\_parser | Extracts the body from the requests :return bytes: the body of the request |
| \_\_section\_one\_parser | Splits the first section of the http\_ophir request (before the headers) :return tuple: (method, uri, http\_version, query\_params) |
| \_\_str\_\_ | Str dunder function for the HttpMsg class :return str: The http\_ophir message in full |

**HttpMsg**

מימוש:

Create easy to use http messages including responses and requests

תכונות:

* error\_codes – error codes in the http message
* body – body of the http message
* headers – dict of the headers

|  |  |
| --- | --- |
| Method | Docs |
| \_\_error\_code\_finder | Returns the error code and message using the error code. :param error\_code: The error code to be found :return bytes: The error code |
| prettify | Prints the message in a readable detailed format :return: None |
| \_\_build\_headers\_bytes | Formats the headers into bytes :return bytes: Byte string with the headers formatte |
| build\_message\_bytes | Builds the http\_ophir message :return bytes: The http\_ophir message |
| \_\_str\_\_ | Str dunder function for the HttpMsg class :return str: The http\_ophir message in full |

**App**

מימוש:

The Base for the web app

תכונות:

* routes – dict holding all the routes
* four\_o\_four – holds the path to the default 404 page
* \_\_closed – bool is connection closed

|  |  |
| --- | --- |
| Method | Docs |
| route | Route decorator - adds the route to the routes' dictionary. :param permission\_cookie: The cookie name that allows permission to page. :param route: the route for the uri :return: what the original function needs to return |
| \_\_receive\_message | Receives a message from a client :param client\_socket: :return bool or HttpParser: False if the message is invalid, HttpParser with the message |
| \_\_handle\_client | Handles the client input. :param request: The request that the server got. :param client\_socket: The client socket. :return: None |
| set\_four\_o\_four | Sets the route to the 404 page. :param route: The route to the 404 page html. :return: None |
| close\_app | Shuts down the server and closes the server :return: None |
| run | Starts the http server. :return: None |

**מסמך בדיקות מלא:**

|  |  |  |  |
| --- | --- | --- | --- |
| דרך התמודדות | דרך הבדיקה והאם צלחה | מה אמורה לבדוק | שם הבדיקה (שיעיד על התוכן) |
| בדיקה של פרוטוקול ה-RTP ותיקון מימושו | בדיקה של חיבור למצלמה המוגדרת. דיווח על שגיאות חיבור (אם קיימות).  נכשל | חיבור מהיר ויציב למצלמה המוגדרת | בדיקת חיבוריות - חיבור למצלמות אלחוטיות |
|  | בדיקה של חיבור למיקרופון המוגדר. דיווח על שגיאות חיבור (אם קיימות).  עבר בהצלחה | זיהוי וחיבור תקין למיקרופון המוגדר | בדיקת חיבוריות - חיבור למיקרופון |
|  | בדיקה של חיבור לשרת. דיווח על שגיאות חיבור (אם קיימות).  עבר בהצלחה | חיבור מהיר ויציב לשרת התמלול | בדיקת חיבוריות - חיבור לשרת |
|  | בדיקה של לכידת וידאו מכל המצלמות. וידוא איכות וידאו משתנה (רזולוציה, קצב פריימים).  עבר בהצלחה | לכידה תקינה של וידאו מכל המצלמה המחוברת | בדיקה פונקציונלית - לכידת וידאו |
|  | בדיקה של קליטת אודיו מהמיקרופון. וידוא איכות שמע משתנה (עוצמה, רעשי רקע).  עבר בהצלחה | קליטה תקינה של אודיו מהמיקרופון המחובר | בדיקה פונקציונלית - קליטת אודיו |
| בדיקה על הליך התמלול  ובדיקת אלטרנטיבות | שימוש בקובץ אודיו מקומי לבדיקה בדיקה אוטומטית של התמלול המתקבל. וידוא תמיכה בשפות שונות, טיפול בהפסקות בדיבור.  נכשל | תמלול אודיו בזמן אמת בצורה מדויקת | בדיקה פונקציונלית - תמלול בזמן אמת |

|  |  |  |  |
| --- | --- | --- | --- |
| שכתוב פעולת הצגת הכתוביות בזמן אמת | בדיקה של הצגת כתוביות על המסך. וידוא סנכרון בין כתוביות לווידאו, אפשרויות עיצוב כתוביות.  נכשל | הצגה תקינה של כתוביות על המסך | בדיקה פונקציונלית - הצגת כתוביות |
|  | ניטור של צריכת משאבים. דיווח על חריגות.  עבר בהצלחה | צריכת משאבים (CPU, זיכרון) תקינה בעת הפעלת התוכנה | בדיקת משאבי מערכת |