Dokumentacja techniczna KairosAI

## GoogleCalendarAPI.py

* getCredentials
  + Argumenty:
    - user\_token\_dict – przyjmuje obiekt słowanika (dict) posiadający dane z danych .json od google calendar api
  + Zwraca:
    - Obiekt google. oauth2.credentials
  + Wyjątki:
    - Google.auth.exceptions.RefreshError
    - Google.oauth2.credentials.exceptions.DeafultCredentialsError
  + Co robi:
    - Konwertuje słownik (user\_token\_dict) na obiekt Credentials jeśli to możliwe. Jeśli można odświeżą token
* getEvents
  + Argumenty:
    - user\_token\_dict:dict,
    - calendar\_id: str = 'primary',
    - time\_min: str = datetime(1970, 1, 1, tzinfo=timezone.utc).isoformat(),
    - time\_max: str = datetime(2100, 1, 1, tzinfo=timezone.utc).isoformat(),
    - max\_results: int = 200,
    - order\_by: str = 'startTime',
    - show\_deleted: bool = False,
    - single\_events: bool = True,
    - show\_hidden\_invitations: bool = False,
    - updated\_min: Optional[datetime] = None,
    - always\_include\_email: bool = True,
    - query=''
  + Zwraca:
    - List[Event]
    - None
  + Co robi:
    - Wyszukuje eventy na podstawie argumentów i zwraca ich liste
* getEvents\_by\_id
  + Argumenty:
    - user\_token\_dict:dict
    - event\_id
    - calendar\_id: str = 'primary'
  + Zwraca:
    - Event
    - None
  + Co robi:
    - Wszukuje eventy na podstawie id i go zwraca lub zwraca None
* addEvent
  + Argumenty:
    - user\_token\_dict:dict
    - event: Event,
    - calendar\_id: str = 'primary',
  + Zwraca:
    - Event
    - None
  + Co robi:
    - Dodaje event i zwraca dodany event lub zwraca None
* updateEvent
  + Argumenty:
    - user\_token\_dict:dict,
    - event: Event,
    - calendar\_id: str = 'primary'
  + Wyjątki:
    - ValueError
  + Zwraca:
    - Event
    - None
  + Co robi:
    - Aktualizuje event i zwraca dodany event lub zwraca None
* updateEvent
  + Argumenty:
    - user\_token\_dict:dict,
    - event: Event,
    - calendar\_id: str = 'primary'
  + Wyjątki:
    - ValueError
  + Zwraca:
    - Event
    - None
  + Co robi:
    - usuwa event i zwraca dodany True lub zwraca False
* Klasa Reminder
  + Funkcje:
    - setUserToken:
      * argumenty:
        + user\_token\_dict
      * co robi:
        + przypisuje \_user\_token\_dict:=user\_token\_dict
    - update:
      * argumenty:
        + time\_max
      * co robi:
        + aktualizuje i dodaje przyszłe przypomnienia(od czasu teraźniejszego do time\_max) w postaci tuple(identyfikator eventu,data przypomnienia), i sortuje
    - get:
      * argumenty:
        + now=None
      * zwraca
        + list[Event]
      * co robi:
        + zwraca eventy których data <= czasu teraźniejszego

## Event.py

* Klasa Event
  + Funkcje:
    - \_\_init\_\_
      * Argumenty
        + Data:dict=None
      * Co robi
        + Tworzy obiekt Event na podstawie słownika
* Klasa EventBuilder
  + Funkcje:
    - \_\_init\_\_
      * Argumenty
        + Event=Event()
      * Co robi
        + Tworzy obiekt EventBuilder
    - as\_calendar\_event
      * Co robi:
        + Ustawia event jako calendar#event
    - with\_summary
      * argumenty:
        + summary:str
      * Co robi:
        + Ustawia tytuł eventu
    - with\_description
      * argumenty:
        + description:str
      * Co robi:
        + Ustawia opis eventu
    - with\_location
      * argumenty:
        + location:str
      * Co robi:
        + Ustawia lokalizacje eventu
    - with\_start\_date
      * argumenty:
        + dateTime:str (poprawny format: *2026-07-01T15:31:00+00:00 ISO)*
      * Co robi:
        + Ustawia czas rozpoczęcia eventu
    - with\_end\_date
      * argumenty:
        + dateTime:str (poprawny format: *2026-07-01T15:31:00+00:00 ISO)*
      * Co robi:
        + Ustawia czas zakończenia eventu
    - with\_attendees
      * argumenty:
        + attendees\_emails:list | str (jeśli str email@domena.pl,email2@domena.pl)
      * Co robi:
        + Dodaje uczestników do eventu
    - with\_color\_id
      * argumenty:
        + color\_id:str poprawne wartości (0,1,2,3,4,5,6,7,8,9)
      * Co robi:
        + Dodaje kolor do eventu
    - with\_reminders
      * argumenty:
        + useDefault:bool,
        + overrides:list[dict[str,str | int]] format: [{'method':'popup' | 'email', 'minutes':int>=0}]
      * Co robi:
        + Dodaje przypomnienia
    - add\_reminder
      * argumenty:
        + method {‘popup’ | ‘emai’}
        + minutes {int >=0}
      * Co robi:
        + Dodaje przypomnienie
    - with\_status
      * argumenty:
        + status {‘confirmed’,tentative’,’canceled’}
      * Co robi:
        + Ustawia status eventowi
    - with\_creator
      * argumenty:
        + email:str
        + displayName:str = None
      * Co robi:
        + Dodaje twórce do eventu
    - with\_anyoneCanAddSelf
      * argumenty:
        + b:bool
      * Co robi:
        + Ustawia atrybut anyoneCanAddSelf eventowi
    - with\_guestsCanInviteOthers
      * argumenty:
        + b:bool
      * Co robi:
        + Ustawia atrybut guestsCanInviteOthers eventowi
    - with\_guestsCanModify
      * argumenty:
        + b:bool
      * Co robi:
        + Ustawia atrybut guestsCanModify eventowi
    - with\_guestsCanSeeOtherGuests
      * argumenty:
        + b:bool
      * Co robi:
        + Ustawia atrybut guestsCanSeeOtherGuests eventowi
    - Build
      * Co robi:
        + Tworzy ostateczny event

## CredenctialsFunctions.py

* create\_authorization\_url
  + Argumenty:
    - user\_id
  + Zwraca:
    - str
  + Co robi:
    - Tworzy URL dla użytkownika do logowania
* create\_authorization\_url
  + Argumenty:
    - user\_id
  + Zwraca:
    - str
  + Co robi:
    - Tworzy URL dla użytkownika do logowania
* check\_user\_credentials
  + Argumenty:
    - user\_id
  + Zwraca:
    - bool
  + Co robi:
    - Sprawdza czy user posiada credentials w bazie danych
* get\_user\_credentials
  + Argumenty:
    - user\_id
  + Zwraca:
    - dict
  + Co robi:
    - Zwraca credentiale user zapisane w bazie
* delete\_user\_credentials
  + Argumenty:
    - user\_id
  + Zwraca:
    - dict
  + Co robi:
    - Usuwa credentiale user zapisane w bazie