

# App

## Example Usage of MSAApp Copyright © 2022 - U2D.ai / S.Welcker

```
Example Usage of MSAApp
Copyright (c) 2022 - U2D.ai / S.Welcker
__version__ = "0.0.1"

from typing import Optional

from sqlmodel import SQLModel

from msaSDK.admin.utils.fields import Field
from msaSDK.models.service import get_msa_app_settings
from msaSDK.service import MSAApp
from msaSDK.utils.scheduler import MSATimers, MSATimerEnum

async def test_timer_min():
    app.logger.info("msaSDK Test Timer Async Every Minute")

def test_timer_five_sec():
    app.logger.info("msaSDK Test Timer Sync 5 Second")

class TestArticle(SQLModel, table=True):
    id: Optional[int] = Field(default=None, primary_key=True, nullable=False)
    title: str = Field(title='ArticleTitle', max_length=200)
    description: Optional[str] = Field(default='', title='ArticleDescription',
max_length=400)
    status: bool = Field(None, title='status')
    content: str = Field(title='ArticleContent')

class TestCategory(SQLModel, table=True):
    id: Optional[int] = Field(default=None, primary_key=True, nullable=False)
    title: str = Field(title='ArticleTitle', max_length=200)
    description: Optional[str] = Field(default='', title='ArticleDescription',
max_length=400)
    status: bool = Field(None, title='status')
    content: str = Field(title='ArticleContent')

get_msa_app_settings.cache_clear()
settings = get_msa_app_settings()
settings.title = "SPK.ai - MSA/SDK MVP"
settings.version = "SPK.0.0.1"
settings.debug = True

my_timers: MSATimers = MSATimers()
my_timers.create_timer(MSATimerEnum.every_minute, test_timer_min)
```

```
my_timers.create_timer(MSATimerEnum.on_the_5_second, test_timer_five_sec)

app = MSAApp(settings=settings, timers=my_timers, auto_mount_site=True,
             sql_models=[TestArticle, TestCategory],
             contact={"name": "msaSDK", "url": "http://u2d.ai", "email":
"stefan@u2d.ai"},
             license_info={"name": "MIT", "url": "https://opensource.org/licenses/MIT",
})

app.logger.info("Initialized " + settings.title + " " + settings.version)

@app.on_event("startup")
async def startup():
    app.logger.info("msaSDK Own Startup MSAUIEvent")
    #app.mount_site()

@app.on_event("shutdown")
async def shutdown():
    app.logger.info("msaSDK Own Shutdown MSAUIEvent")

if __name__ == '__main__':
    pass
```

---

Last update: September 13, 2022

Created: August 24, 2022