msaSDK Module

`.service`

Main Service Module for MSAApp.

Initialize with a MSAServiceDefintion Instance to control the features and functions of the MSAApp.

Attributes

password_helper `module-attribute`

```
password_helper = PasswordHelper(security_context)
```

> Password Helper Instance

security `module-attribute`

```
security = getMSASecurity()
```

> MSASecurity instance

security_context `module-attribute`

```
security_context = CryptContext(
    schemes=["bcrypt"], deprecated="auto"
)
```

> Security Context for Password Helper

Classes

MSAApp

> Bases: `MSAFastAPI`

Creates an application msaSDK instance.

> **Note**
>
> As with FastApi the MSAApp provides two events: `startup`: A list of callables to run on application startup. Startup handler callables do not take any arguments, and may be be either standard functions, or async functions. `shutdown`: A list of callables to run on application shutdown. Shutdown handler callables do not take any arguments, and may be be either standard functions, or async functions. Those are also used internally, which are triggered before the external events.
>
> Do not include the `self` parameter in the `Args` section.

PARAMETER	DESCRIPTION
`settings`	MSAServiceDefinition (Must be provided), instance of a service definition with all settings **TYPE:** `MSAServiceDefinition`
`sql_models`	List of SQLModel Default None, provide list of your SQLModel Classes and the instance can create CRUD API and if site is enabled also UI for CRUD **TYPE:** `List[SQLModel]` **DEFAULT:** `None`
`auto_mount_site`	Default True, if site is enabled in settings and this is true, mounts the site in internal startup event. **TYPE:** `bool` **DEFAULT:** `True`

ATTRIBUTE	DESCRIPTION
`logger`	loguru logger instance
`auto_mount_site`	bool auto_mount_site **TYPE:** `bool`
`settings`	MSAServiceDefinition settings instance.
`healthdefinition`	MSAHealthDefinition settings.healthdefinition **TYPE:** `MSAHealthDefinition`
`limiter`	Limiter = None **TYPE:** `Limiter`

ATTRIBUTE	DESCRIPTION
`db_engine`	AsyncEngine = Db Engine instance **TYPE:** `Limiter`
`sql_models`	List[SQLModel] = sql_models **TYPE:** `List[SQLModel]`
`sql_cruds`	List[MSASQLModelCrud] = [] **TYPE:** `List[MSASQLModelCrud]`
`scheduler`	MSAScheduler = None **TYPE:** `MSAScheduler`
`site`	AdminSite Admin/Auth Site instance.
`scheduler_task`	The Task instance that runs the Scheduler in the Background
`ROOTPATH`	str os.path.join(os.path.dirname(**file**))

Attributes

Base `instance-attribute`

```
Base: DeclarativeMeta = declarative_base()
```

ROOTPATH `instance-attribute`

```
ROOTPATH = os.path.join(os.path.dirname(__file__))
```

abstract_fs `instance-attribute`

```
abstract_fs: MSAFilesystem = None
```

auto_mount_site `instance-attribute`

```
auto_mount_site: bool = auto_mount_site
```

fs `instance-attribute`

```
fs: FS = None
```

graphql_app `instance-attribute`

```
graphql_app: GraphQLRouter = None
```

graphql_schema `instance-attribute`

```
graphql_schema: schema = None
```

healthcheck `instance-attribute`

```
healthcheck: health.MSAHealthCheck = None
```

healthdefinition `instance-attribute`

```
healthdefinition: MSAHealthDefinition = (
    self.settings.healthdefinition
)
```

json_db_engine `instance-attribute`

```
json_db_engine: TinyDB = None
```

limiter `instance-attribute`

```
limiter: Limiter = None
```

logger `instance-attribute`

```
logger = logger_gruru
```

scheduler `instance-attribute`

```
scheduler: MSAScheduler = None
```

settings `instance-attribute`

```
settings = settings
```

site instance-attribute

```
site = None
```

sql_cruds instance-attribute

```
sql_cruds: List[MSASQLModelCrud] = []
```

sql_models instance-attribute

```
sql_models: List[SQLModel] = sql_models
```

sqlite_db_engine instance-attribute

```
sqlite_db_engine: AsyncEngine = None
```

templates instance-attribute

```
templates = Jinja2Templates(
    directory=self.settings.templates_dir
)
```

Functions

__init__

```
__init__(
    settings: MSAServiceDefinition,
    sql_models: List[SQLModel] = None,
    auto_mount_site: bool = True,
    *args,
    **kwargs
) -> None
```

get_healthcheck async

```
get_healthcheck(request: Request) -> ORJSONResponse
```

> Get Healthcheck Status

get_scheduler_log async

```
get_scheduler_log(
    request: Request,
    optionClearLog: bool = False,
    optionFORCEClearLog: bool = False,
) -> MSASchedulerLog
```

Get Service Scheduler Log

PARAMETER	DESCRIPTION
`request`	The input http request object **TYPE:** `Request`
`optionClearLog`	If True the Log gets cleared after the response was build **TYPE:** `bool` **DEFAULT:** `False`
`optionFORCEClearLog`	Forcing the clearing of the log before the response gets created **TYPE:** `bool` **DEFAULT:** `False`

RETURNS	DESCRIPTION
`sst`	MSASchedulerLog Pydantic Response Model **TYPE:** `MSASchedulerLog`

get_scheduler_status async

```
get_scheduler_status(
    request: Request,
) -> MSASchedulerStatus
```

Get Service Scheduler Status, with the registered Task's

PARAMETER	DESCRIPTION
`request`	The input http request object **TYPE:** `Request`

RETURNS	DESCRIPTION

RETURNS	DESCRIPTION
`sst`	MSASchedulerStatus Pydantic Response Model **TYPE:** `MSASchedulerStatus`

get_services_definition

```
get_services_definition(
    request: Request,
) -> MSAServiceDefinition
```

Get Service Definition Info

PARAMETER	DESCRIPTION
`request`	The input http request object **TYPE:** `Request`

RETURNS	DESCRIPTION
`settings`	MSAServiceDefinition Pydantic Response Model **TYPE:** `MSAServiceDefinition`

get_services_openapi_info

```
get_services_openapi_info(
    request: Request,
) -> MSAOpenAPIInfo
```

Get Service OpenAPI Info

PARAMETER	DESCRIPTION
`request`	The input http request object **TYPE:** `Request`

RETURNS	DESCRIPTION

RETURNS	DESCRIPTION
`oai`	MSAOpenAPIInfo Paydantic Response Model **TYPE:** `MSAOpenAPIInfo`

get_services_openapi_schema

```
get_services_openapi_schema(
    request: Request,
) -> ORJSONResponse
```

Get Service OpenAPI Schema

PARAMETER	DESCRIPTION
`request`	The input http request object **TYPE:** `Request`

RETURNS	DESCRIPTION
`openapi`	ORJSONResponse openapi schema **TYPE:** `ORJSONResponse`

get_services_settings

```
get_services_settings(request: Request) -> ORJSONResponse
```

Get Service OpenAPI Schema

PARAMETER	DESCRIPTION
`request`	The input http request object **TYPE:** `Request`

RETURNS	DESCRIPTION
`settings`	ORJSONResponse **TYPE:** `ORJSONResponse`

get_services_status async

```
get_services_status(request: Request) -> MSAServiceStatus
```

Get Service Status Info

PARAMETER	DESCRIPTION
`request`	The input http request object **TYPE:** `Request`

RETURNS	DESCRIPTION
`sst`	MSAServiceStatus Pydantic Response Model **TYPE:** `MSAServiceStatus`

index_page

```
index_page(request: Request) -> _TemplateResponse
```

Get Service Index.html Page

PARAMETER	DESCRIPTION
`request`	The input http request object **TYPE:** `Request`

init_graphql async

```
init_graphql(strawberry_schema) -> None
```

Internal helper function to initialize the graphql router

monitor async

```
monitor(request: Request) -> _TemplateResponse
```

Simple Service Monitor Page. Only works if pages is enabled in MSAServiceDefinition

PARAMETER	DESCRIPTION
`request`	The input http request object **TYPE:** `Request`

monitor_inline async

```
monitor_inline(request: Request) -> _TemplateResponse
```

Simple Monitor Page as Inline without head and body tags. Only works if pages is enabled in MSAServiceDefinition

PARAMETER	DESCRIPTION
`request`	The input http request object **TYPE:** `Request`

mount_site

```
mount_site() -> None
```

msa_exception_handler async

```
msa_exception_handler(request: Request, exc: HTTPException)
```

Handles all HTTPExceptions if enabled with HTML Response or forward error if the code is in the exclude settings list.

PARAMETER	DESCRIPTION
`request`	The input http request object **TYPE:** `Request`
`exc`	The HTTPException instance

RETURNS	DESCRIPTION
	HTTPException or Template

msa_exception_handler_disabled `async`

```
msa_exception_handler_disabled(
    request: Request, exc: HTTPException
) -> JSONResponse
```

> Handles all HTTPExceptions if Disabled with JSON Response.
>
> | PARAMETER | DESCRIPTION |
> |---|---|
> | `request` | The input http request object
TYPE: `Request` |
>
> | RETURNS | DESCRIPTION |
> |---|---|
> | `HTTPException` | as JSONResponse
TYPE: `JSONResponse` |

profiler

```
profiler(request: Request) -> _TemplateResponse
```

> Simple Profiler Page. Only works if pages is enabled in MSAServiceDefinition
>
> | PARAMETER | DESCRIPTION |
> |---|---|
> | `request` | The input http request object
TYPE: `Request` |

shutdown_event `async`

```
shutdown_event() -> None
```

> Internal Shutdown event handler

startup_event `async`

```
startup_event() -> None
```

> Internal Startup Event Handler

testpage

```
testpage(request: Request) -> _TemplateResponse
```

> Simple Testpage to see if the Micro Service is up and running. Only works if pages is enabled in MSAServiceDefinition
>
> | PARAMETER | DESCRIPTION |
> |---|---|
> | `request` | The input http request object
TYPE: `Request` |

validation_exception_handler async

```
validation_exception_handler(
    request: Request, exc: RequestValidationError
) -> JSONResponse
```

Functions

getSecretKey

```
getSecretKey()
```

> Get Secret Key for Token creation from OS Environment Variable **SECRET_KEY_TOKEN**
>
> | RETURNS | DESCRIPTION |
> |---|---|
> | `key` | The SECRET_KEY_TOKEN. |

getSecretKeyCSRF

```
getSecretKeyCSRF() -> str
```

> Get Secret Key for CSRF Middleware from OS Environment Variable **SECRET_KEY_CSRF**
>
> | RETURNS | DESCRIPTION |
> |---|---|

RETURNS	DESCRIPTION
`key`	The SECRET_KEY_CSRF. **TYPE:** `str`

getSecretKeySessions

```
getSecretKeySessions()
```

Get Secret Key for Session Middleware from OS Environment Variable **SECRET_KEY_SESSIONS**

RETURNS	DESCRIPTION
`key`	The SECRET_KEY_SESSIONS.

Last update: September 16, 2022

Created: September 16, 2022