```
from pydantic import BaseModel, Field

from typing import Dict, Optional, List

import time

import uuid



class SwarmAPISchema(BaseModel):


    id: str = Field(default_factory=lambda: uuid.uuid4().hex)

    swarm_name: Optional[str] = Field(default="Swarm API")

    swarm_description: Optional[str] = Field(default="Swarm API description")

    created_at: Optional[int] = Field(default_factory=lambda: int(time.time()))

    owned_by: Optional[str] = Field(

        default="TGSC",

        description="The owner of the API.",

        examples="TGSC",

    )

    tags: Optional[list] = Field(

        default=...,

        description="The tags for the API.",

        examples=["tag_1", "agent"],

    )

    use_cases: Optional[Dict[str, str]] = Field(

        default=...,

        description="The use cases for the API.",

        examples={
            "use_case_1": "Use case 1 description",
            "use_case_2": "Use case 2 description",
        },
    )


class AllSwarmsSchema(BaseModel):
    swarms: Optional[List[SwarmAPISchema]] = Field(
        default=...,
        description="The list of all swarms.",
        examples=[],
    )


# example = {
#     "swarm_name": "Swarm API",
#     "swarm_description": "Swarm API description",
#     "created_at": 1628584185,
#     "owned_by": "TGSC",
#     "tags": ["tag_1", "agent"],
#     "use_cases": {
#         "use_case_1": "Use case 1 description",
#         "use_case_2": "Use case 2 description",
#     },
# }
# # Define the input model using Pydantic

# out = SwarmAPISchema(**example)

# print(out)
```