```
import yaml



def update_mkdocs(
    class_names,
    base_path="docs/zeta/nn/modules",
    mkdocs_file="mkdocs.yml",
):
    """
    Update the mkdocs.yml file with new documentation links.

    Args:
    - class_names: A list of class names for which documentation is generated.
    - base_path: The base path where documentation Markdown files are stored.
    - mkdocs_file: The path to the mkdocs.yml file.
    """
    with open(mkdocs_file) as file:
        mkdocs_config = yaml.safe_load(file)

    # Find or create the 'zeta.nn.modules' section in 'nav'
    zeta_modules_section = None
    for section in mkdocs_config.get("nav", []):
        if "zeta.nn.modules" in section:
            zeta_modules_section = section["zeta.nn.modules"]
            break
```

```
    if zeta_modules_section is None:
        zeta_modules_section = {}
        mkdocs_config["nav"].append(
            {"zeta.nn.modules": zeta_modules_section}
        )

    # Add the documentation paths to the 'zeta.nn.modules' section
    for class_name in class_names:
        doc_path = f"{base_path}/{class_name.lower()}.md"
        zeta_modules_section[class_name] = doc_path

    # Write the updated content back to mkdocs.yml
    with open(mkdocs_file, "w") as file:
        yaml.safe_dump(mkdocs_config, file, sort_keys=False)


# Example usage
classes = [
    "DenseBlock",
    "HighwayLayer",
    "MultiScaleBlock",
    "FeedbackBlock",
    "DualPathBlock",
    "RecursiveBlock",
    "PytorchGELUTanh",
    "NewGELUActivation",
```

```
    "GELUActivation",
    "FastGELUActivation",
    "QuickGELUActivation",
    "ClippedGELUActivation",
    "AccurateGELUActivation",
    "MishActivation",
    "LinearActivation",
    "LaplaceActivation",
    "ReLUSquaredActivation",
]

update_mkdocs(classes)
```