**Customization of MSc Workflow and Documentation System**

The MSc Workflow and Documentation System is designed with flexibility in mind, allowing you to easily tailor its components to your specific requirements. Here's how you can customize various aspects of the project:

**Customizing the Django Admin Dashboard:**

The Django admin dashboard can be customized to match your institution's branding and user experience. To achieve this:

* **Static Files:** We Customized the look and feel by adding our own CSS and JavaScript files. These was done by freezing the static files:

python manage.py collectstatic

* **HTML Templates:** Create new HTML templates or modify existing ones as needed. These templates are stored in the **templates** directory and can be adjusted to match your desired layout and design.

**Modifying Existing Workflows or Adding New Ones:**

To modify existing workflows or add new ones, follow these steps:

1. **Views (views.py):** Update or create views that define the logic and behavior of your workflows. These views handle data processing, interactions, and render templates.
2. **URLs (urls.py):** Define URL patterns that map to your views. These patterns determine how users access the various workflows in your system.
3. **Models (models.py):** Modify the database models to accommodate any changes or additions required for your workflows. Ensure that the data structures align with your new features.
4. **Admin (admin.py):** This is a crucial step when modifying workflows. Customize the admin interface to provide an intuitive and user-friendly experience for managing workflows and associated data.

**Customizing Database Configurations:**

To customize the database configuration, you'll work with the **settings.py** file. Django makes it straightforward to adapt the database to your specific needs:

1. **Database Engine:** In the **settings.py** file, you can configure the database engine you wish to use. By default, your project uses SQLite, but you can switch to other database systems like PostgreSQL or MySQL by adjusting the **DATABASES** setting.
2. **Database Name and Credentials:** Modify the database name, username, password, and other credentials as required by your chosen database system.
3. **Database Migrations:** After making changes to your models or database configuration, create and apply database migrations using:

python manage.py makemigrations

python manage.py migrate

Customizing the MSc Workflow and Documentation System allows you to tailor it to the specific needs of your institution, enabling seamless integration with your existing workflows and processes.

Remember, while customization provides immense flexibility, it's important to maintain consistency and follow best practices to ensure the integrity of the system.