Queuify

Turn everything into a queue!

Setup

This project uses poetry, a python dependency manager.

Once you have poetry, you can run

```
poetry shell
```

It will pull the project's dependencies and create a virtualenv-enabled shell.
Every other python command will need to run inside a poetry session.

If you don't want to install poetry, see The docker section

Dependencies

We use the following dependencies:

- `django-crispy-forms`: Makes forms look better
- `crispy-bootstrap5`: As crispy-forms does not have bootstrap5 support
- `Pillow`: Manages images
- `django-timezone-field`: Helps display user-friendly timezones.
- `tzdata`: Has a timezone database.
- `django-webpush`: Handles Web push notifications
- `six`: Is required by django-webpush

Usage

We provide an example database, create it using:

```
python3 manage.py resetdb
```

Use this script to start the software:

```
python3 manage.py runserver
```

Always remember to run it inside poetry.

Docker

Alternatively, you can use docker-compose, you won't even require a python
installation

```
docker-container build
docker-container up
```

This will create the Docker image, populate a test database and start the server

Test database

The test database creates users and queues on the fly, to test our site we suggest using an already present user, as they have already friends, queues and one month of previous (randomly generated) data.

We suggest using the (obviously fake) account named "molly0xFF" as it has already friends and is also a moderator.

User logins:

- `molly0xFF: wikipiki` (moderator)
- `torvalds: isecretlyusewindows` (moderator)
- `hamilton: hamilton123`
- `edsheeran: Cmaj9add4add6(b13)(b5)`
- `carolshaw: idontknowastrongpasswordmaybe`
- `Sora_Sakurai: nintendont`
- `admin: password` (amministratore sito)