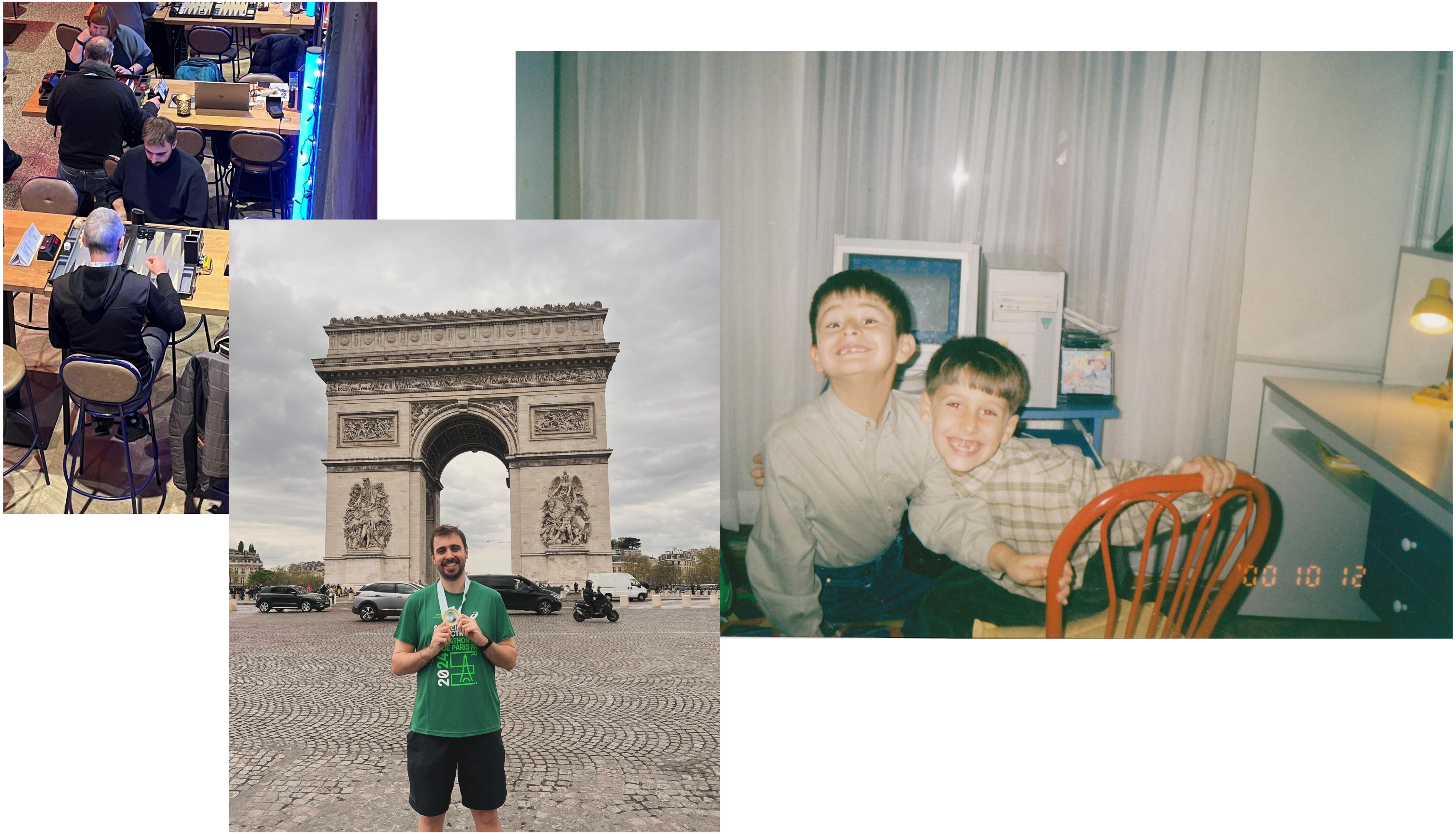
Multilingual Django Applications with a sprinkle of AI ✨

Efe Öge

Γεια σας παιδιά, είμαι ο Efe

- Originally from **Türkiye**, based in **The Netherlands**
- **Past:** Founding engineer in early phase startups
- **Now:** Senior engineer @ Sendcloud 📦
- Writes at **efe.me** 🔗
- DjangoNL & Mentor at Django Girls Amsterdam (previously PyCon Turkey, Python Istanbul)
- Django, Python, LLMs, Backend
- Boring Technology Advocate


2024
DE PARIS
'00 10 12

1) What is multilingual application?

2) Understanding user’s language

3) Multilingual app with Django

4) Speeding up translation with AI ✨ (yesglot)

5) Q&A

Multilingual application

A multilingual application is an app that can work in **more than one language**.

- The text (buttons, menus, messages) changes to the chosen language.
- Numbers, dates, money formats also follow the language rules. *(e.g. 13/12/1989 in EU, 12/13/1989 in US)*
- Users can switch the language in settings or the app can detect it automatically.
- It supports different writing styles like left-to-right (English) and right-to-left (Arabic).

Multilingual application

A multilingual application is an app that can work in **more than one language**.

- **The text (buttons, menus, messages) changes to the chosen language.**
- Numbers, dates, money formats also follow the language rules. *(e.g. 13/12/1989 in EU, 12/13/1989 in US)*
- **Users can switch the language in settings or the app can detect it automatically.**
- It supports different writing styles like left-to-right (English) and right-to-left (Arabic).

Multilingual application

- The application written in one language. (mostly in English)
- Translatable texts should be defined.
  - “Welcome to PyCon Greece” → translatable (user facing)
  - “DB backup is failed” → non-translatable (backend only, e.g. logging)

Döner vs Gyros

- Who Did It First
    - Doner fans claim Greeks just copied their kebab and rebranded it as gyros.
    - Greeks fire back saying they perfected it — doner is just the "rough draft."
- Meat Wars
    - Doner prides itself on beef or lamb, calling pork-based gyros "a downgrade."
    - Gyros lovers argue pork + tzatziki is superior, and doner's sauces are "greasy street food."
- Cultural Identity
    - Doner = Turkish pride, a symbol of migration and street food dominance in Europe.
    - Gyros = Greek pride, "authentic Mediterranean," claiming doner is "too Middle Eastern, not European enough."

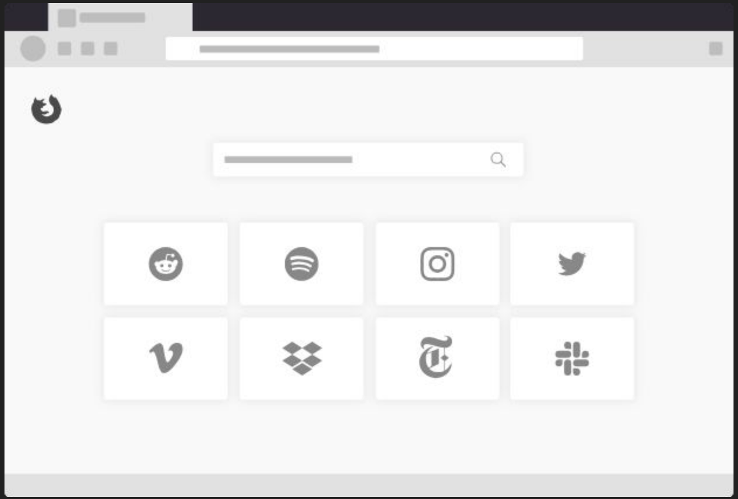
1) What is multilingual application?

2) Understanding user’s language

3) Multilingual app with Django

4) Speeding up translation with AI ✨ (yesglot)

5) Q&A

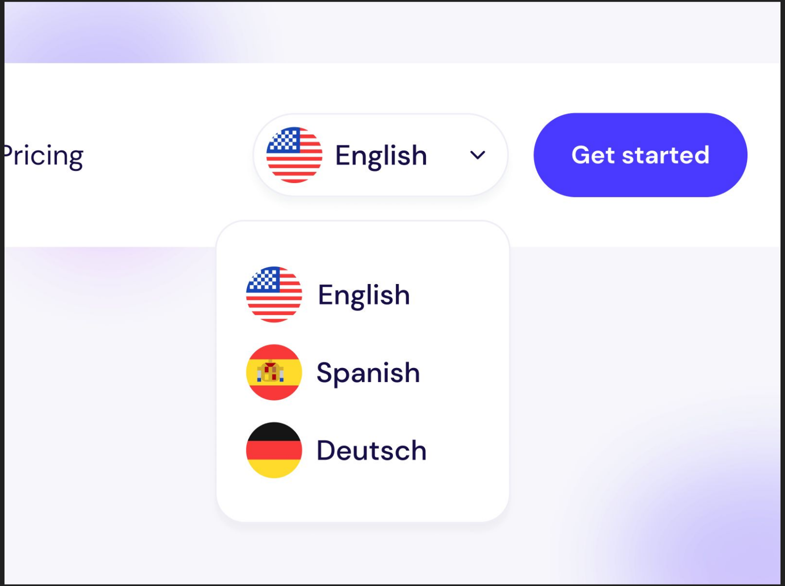


Content Negotiation in HTTP Protocol

Negotiation of what?

Entity	**HTTP Header**	**Example**
Media Type	Accept	text/html, image/png
Character Encoding	Accept-Charset	UTF-8, ISO-8859-1
Language	**Accept-Language**	**en-US, el-GR, en-GB**
Encoding (compression)	Accept-Encoding	gzip, deflate, brotli

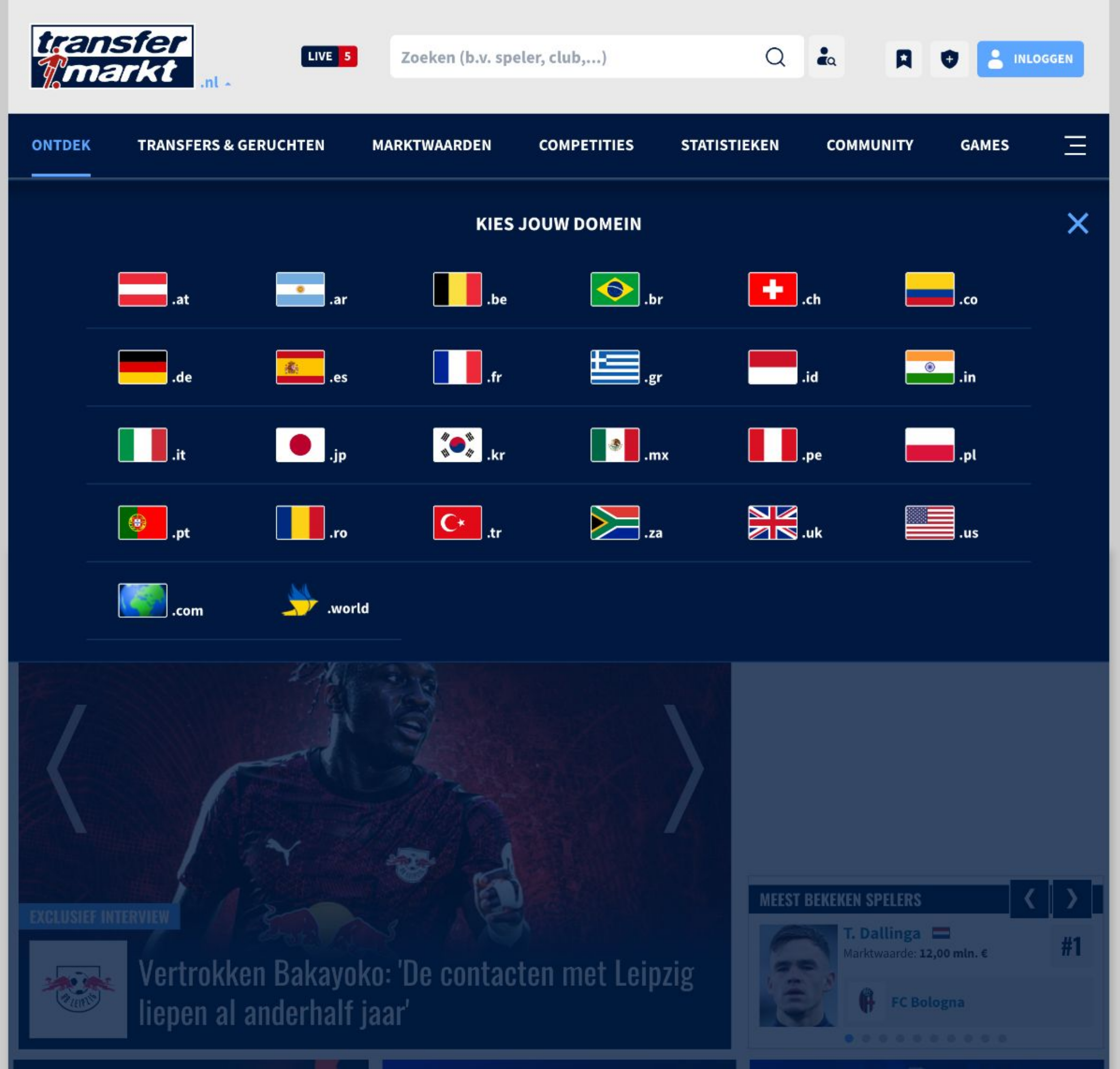
Default to
“Accept-Language”,
but don’t depend on it



Pricing
English
Get started
English
Spanish
Deutsch

session.language

user.language



Reclame

KIES JOUW DOMEIN

.at	.ar	.be	.br	.ch	.co
.de	.es	.fr	.gr	.id	.in
.it	.jp	.kr	.mx	.pe	.pl
.pt	.ro	.tr	.za	.uk	.us
.com	.world				

EXCLUSIEF INTERVIEW

Vertrokken Bakayoko: 'De contacten met Leipzig liepen al anderhalf jaar'

MEEST BEKEKEN SPELERS

T. Dallinga
Marktwaarde: 12,00 mln. €
#1
FC Bologna

Checklist for understanding user’s language

Unauthenticated Users:

1) Default: Accept-Language
2) Session.language

Authenticated Users:

- user.language

1) What is multilingual application?

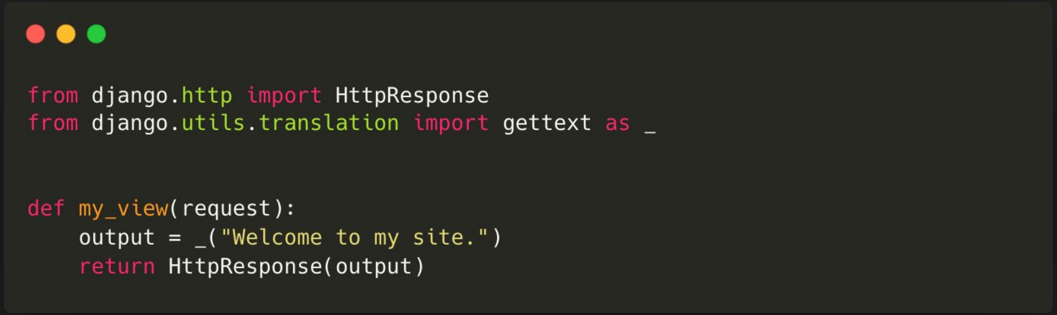
2) Understanding user’s language

3) Multilingual app with Django

4) Speeding up translation with AI ✨ (yesglot)

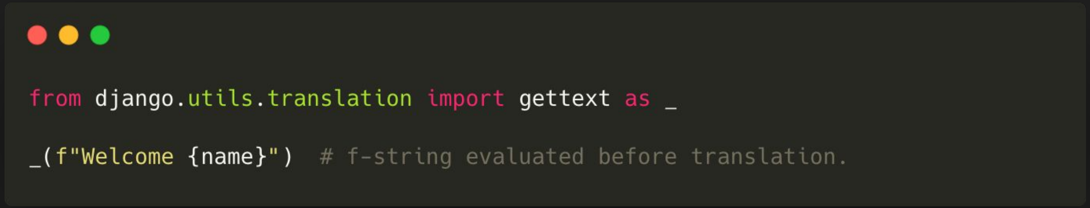
5) Q&A

1. gettext



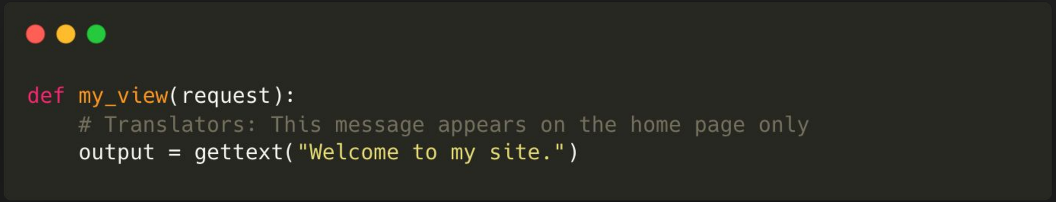
```python
from django.http import HttpResponse
from django.utils.translation import gettext as _


def my_view(request):
    output = _("Welcome to my site.")
    return HttpResponse(output)
```

1. gettext (no f-string)



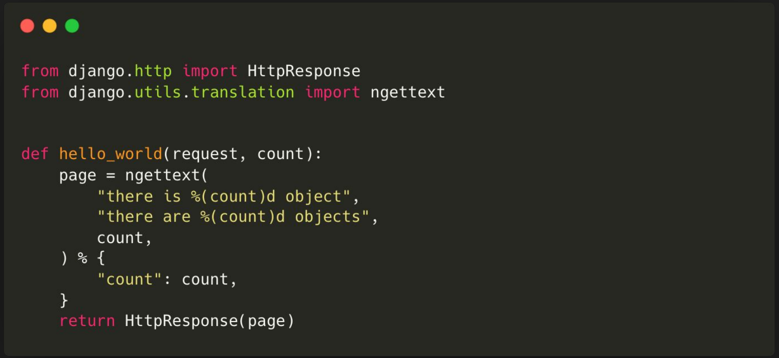
```python
from django.utils.translation import gettext as _

_(f"Welcome {name}")  # f-string evaluated before translation.
```

1. gettext (with comments)



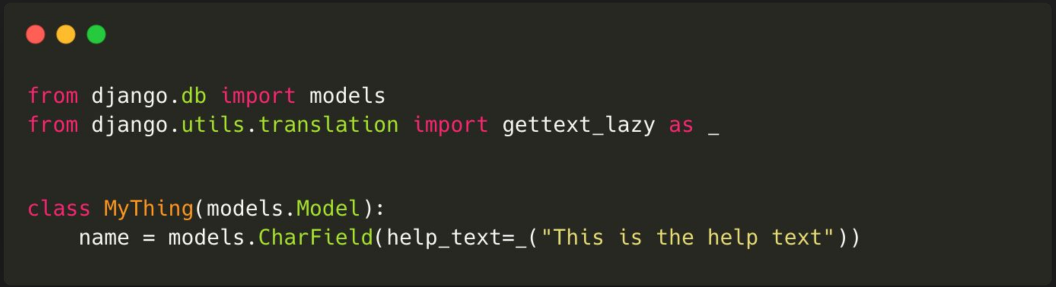
```python
def my_view(request):
    # Translators: This message appears on the home page only
    output = gettext("Welcome to my site.")
```

1. ngettext for plurals



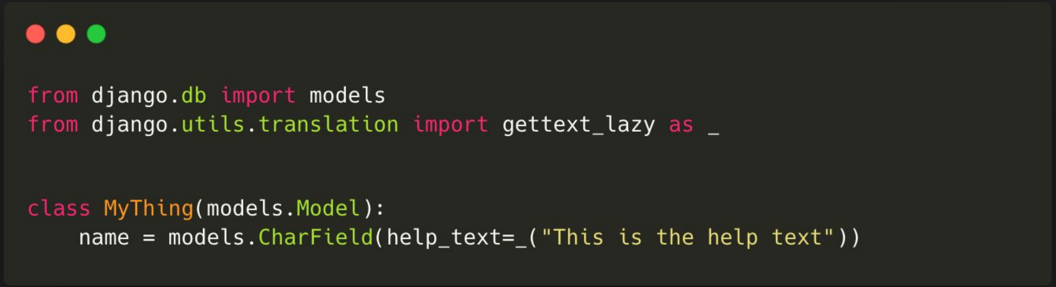
```python
from django.http import HttpResponse
from django.utils.translation import ngettext


def hello_world(request, count):
    page = ngettext(
        "there is %(count)d object",
        "there are %(count)d objects",
        count,
    ) % {
        "count": count,
    }
    return HttpResponse(page)
```

1. gettext but lazy



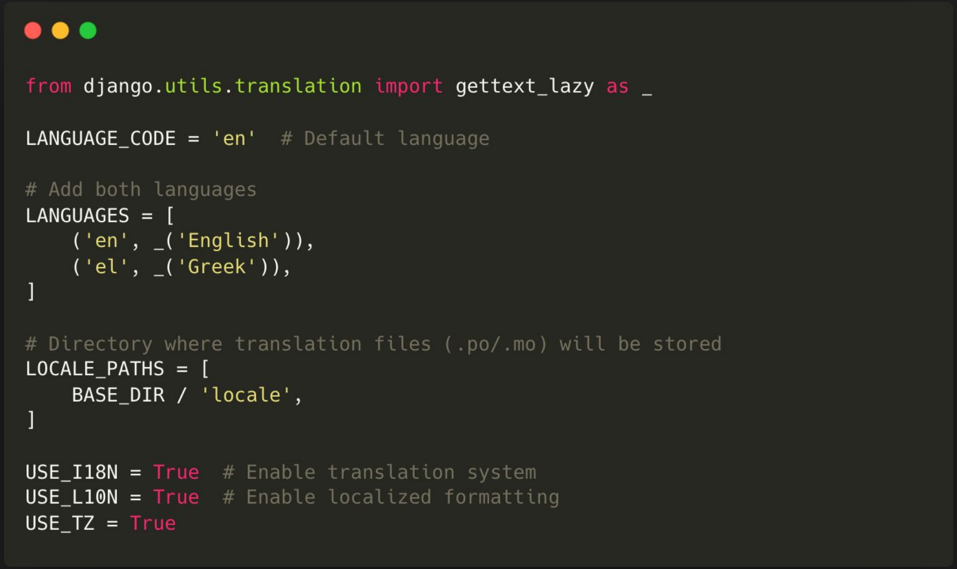
```python
from django.db import models
from django.utils.translation import gettext_lazy as _


class MyThing(models.Model):
    name = models.CharField(help_text=_("This is the help text"))
```

1. gettext but lazy



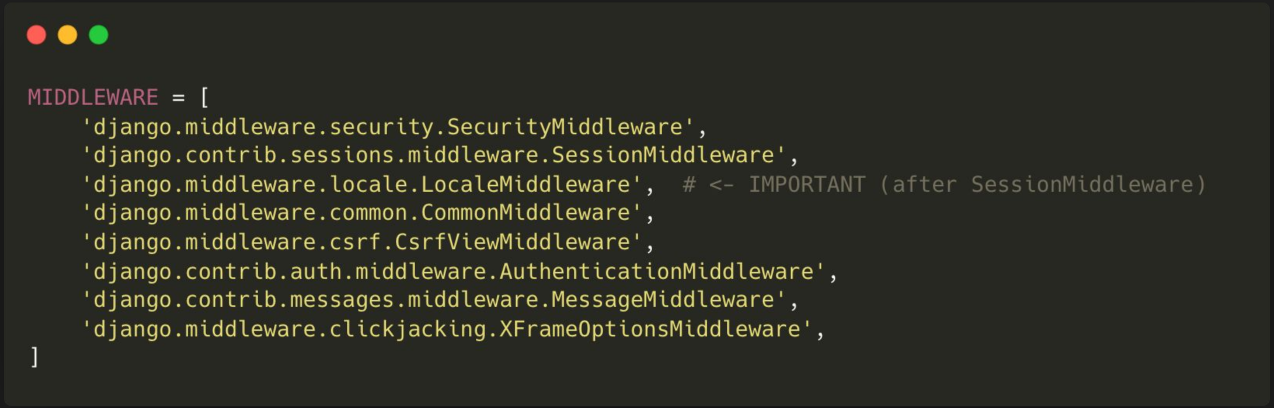
```python
from django.db import models
from django.utils.translation import gettext_lazy as _


class MyThing(models.Model):
    name = models.CharField(help_text=_("This is the help text"))
```

2. settings.py



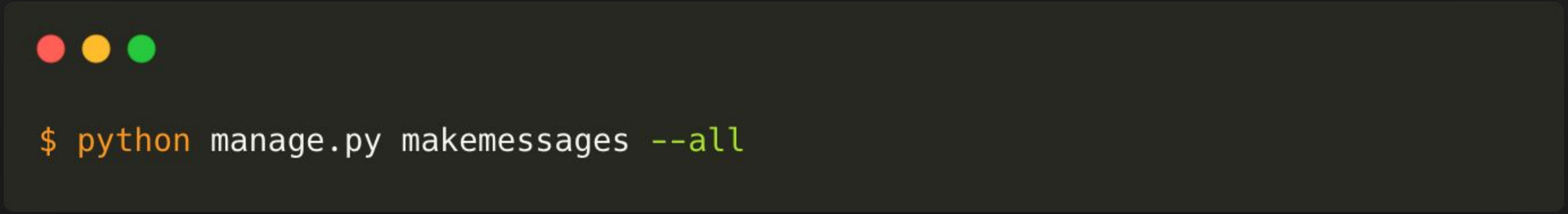
```python
from django.utils.translation import gettext_lazy as _

LANGUAGE_CODE = 'en'  # Default language

# Add both languages
LANGUAGES = [
    ('en', _('English')),
    ('el', _('Greek')),
]

# Directory where translation files (.po/.mo) will be stored
LOCALE_PATHS = [
    BASE_DIR / 'locale',
]

USE_I18N = True  # Enable translation system
USE_L10N = True  # Enable localized formatting
USE_TZ = True
```

3. MIDDLEWARE



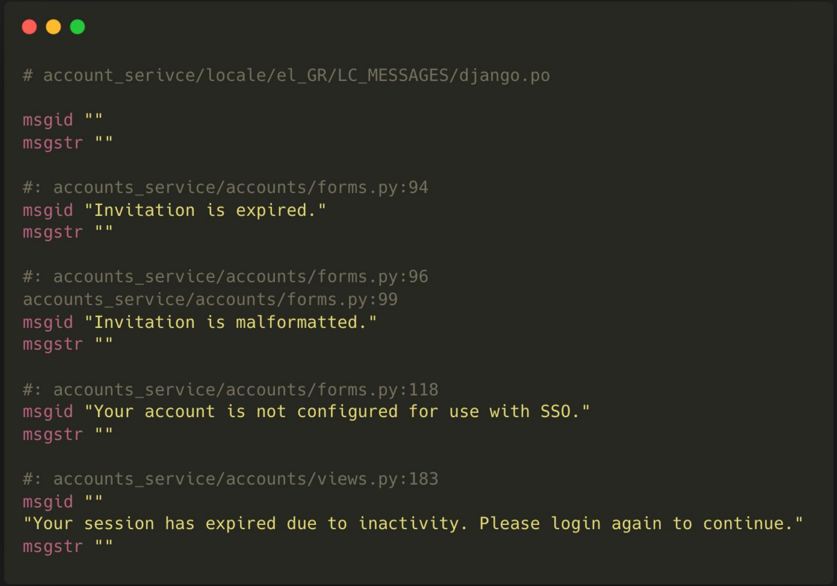
```
MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.locale.LocaleMiddleware',  # <- IMPORTANT (after SessionMiddleware)
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]
```

4. manage.py makemessages



```
$ python manage.py makemessages --all
```

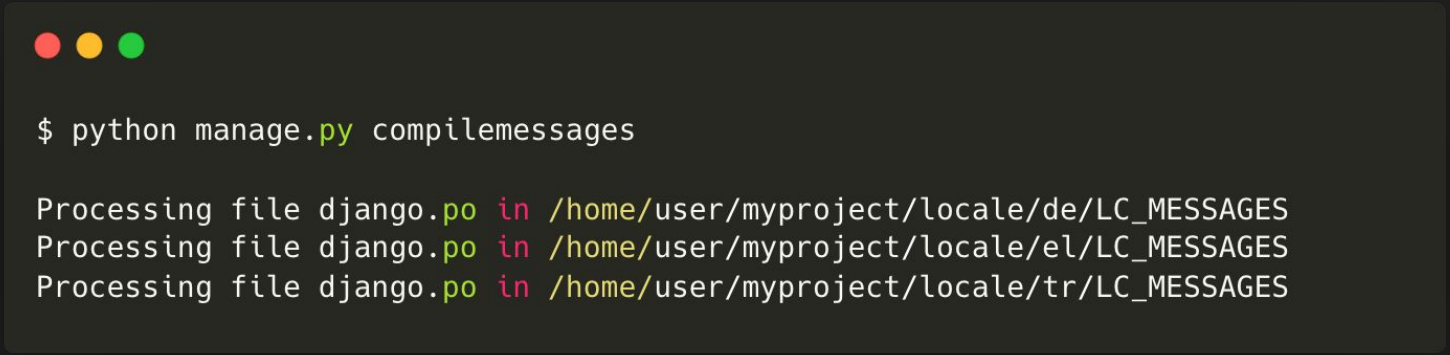
4. manage.py makemessages



```
# account_serivce/locale/el_GR/LC_MESSAGES/django.po

msgid ""
msgstr ""

#: accounts_service/accounts/forms.py:94
msgid "Invitation is expired."
msgstr ""

#: accounts_service/accounts/forms.py:96
accounts_service/accounts/forms.py:99
msgid "Invitation is malformatted."
msgstr ""

#: accounts_service/accounts/forms.py:118
msgid "Your account is not configured for use with SSO."
msgstr ""

#: accounts_service/accounts/views.py:183
msgid ""
"Your session has expired due to inactivity. Please login again to continue."
msgstr ""
```

5. Find a translator ✨

6. python manage.py compilemessages



```
$ python manage.py compilemessages

Processing file django.po in /home/user/myproject/locale/de/LC_MESSAGES
Processing file django.po in /home/user/myproject/locale/el/LC_MESSAGES
Processing file django.po in /home/user/myproject/locale/tr/LC_MESSAGES
```

Checklist for multilingual Django application

1. Use **gettext** for customer facing strings
2. Define **supported languages** in settings.py
3. Use **LocaleMiddleware** before SessionMiddleware
4. **manage.py makemessages** to export translatable files to .po files
5. Have a translator ✨
6. **manage.py compilemessages** to convert .po to .mo
7. “Boró na milíso elliniká” 🇬🇷

Checklist for multilingual Django application

1. Use gettext for customer facing strings
2. Define supported languages in settings.py
3. Use LocaleMiddleware before SessionMiddleware
4. manage.py makemessages to export translatable files to .po files
5. **Have a translator ✨**
6. manage.py compilemessages to convert .po to .mo
7. “Boró na milíso elliniká” 🇬🇷

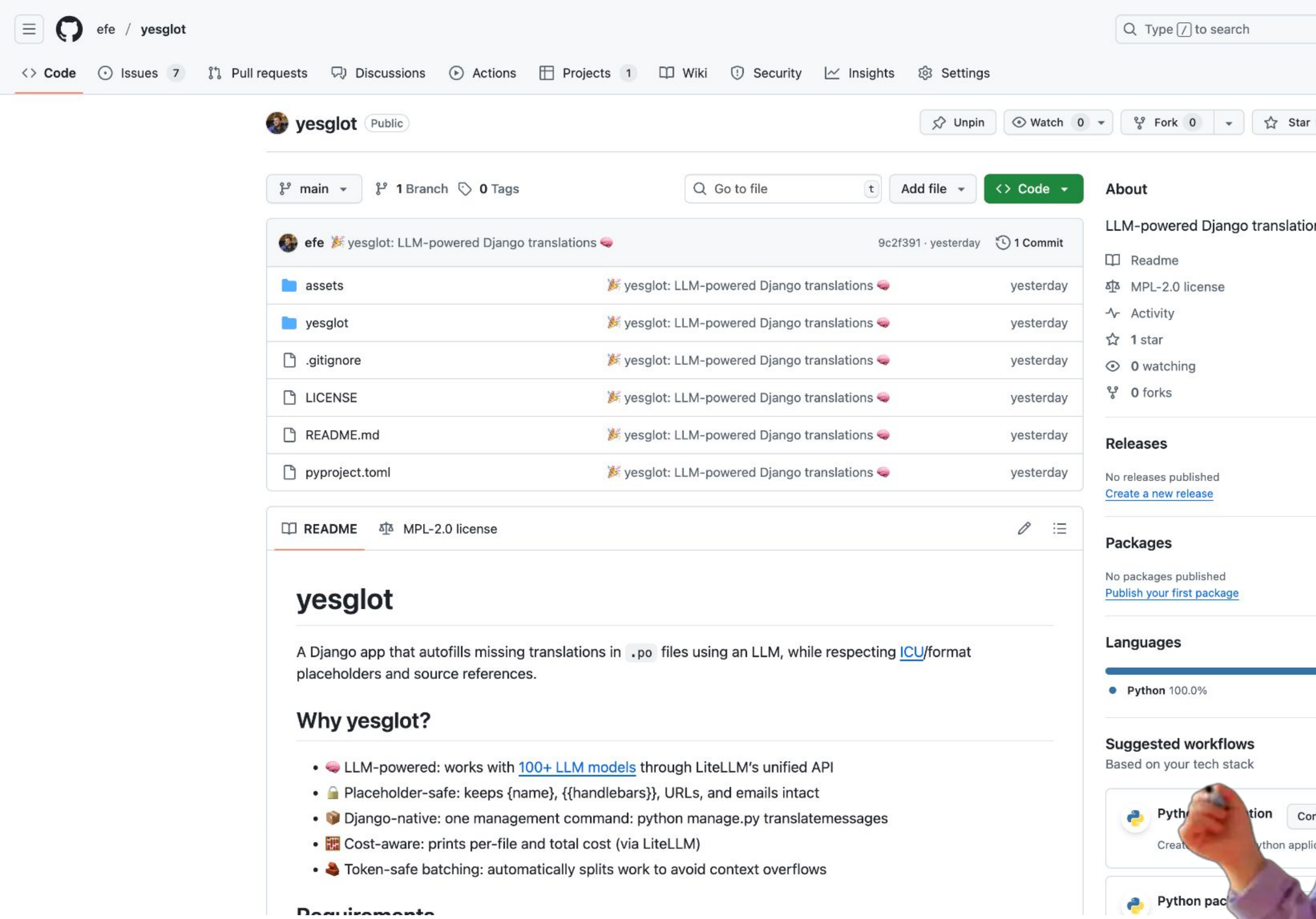
1) What is multilingual application?

2) Understanding user’s language

3) Multilingual app with Django

4) Speeding up translation with AI ✨ (yesglot)

5) Q&A



yesglot Public

efe 🎉 yesglot: LLM-powered Django translations 🧠 9c2f391 · yesterday 1 Commit

assets	🎉 yesglot: LLM-powered Django translations 🧠	yesterday
yesglot	🎉 yesglot: LLM-powered Django translations 🧠	yesterday
.gitignore	🎉 yesglot: LLM-powered Django translations 🧠	yesterday
LICENSE	🎉 yesglot: LLM-powered Django translations 🧠	yesterday
README.md	🎉 yesglot: LLM-powered Django translations 🧠	yesterday
pyproject.toml	🎉 yesglot: LLM-powered Django translations 🧠	yesterday

README MPL-2.0 license

yesglot

A Django app that autofills missing translations in `.po` files using an LLM, while respecting ICU/format placeholders and source references.

Why yesglot?

- 🧠 LLM-powered: works with 100+ LLM models through LiteLLM's unified API
- 🔒 Placeholder-safe: keeps {name}, {{handlebars}}, URLs, and emails intact
- 📦 Django-native: one management command: python manage.py translatemessages
- 🧮 Cost-aware: prints per-file and total cost (via LiteLLM)
- 🧱 Token-safe batching: automatically splits work to avoid context overflows

Requirements

About

LLM-powered Django translations 🧠

Readme
MPL-2.0 license
Activity
1 star
0 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

● Python 100.0%

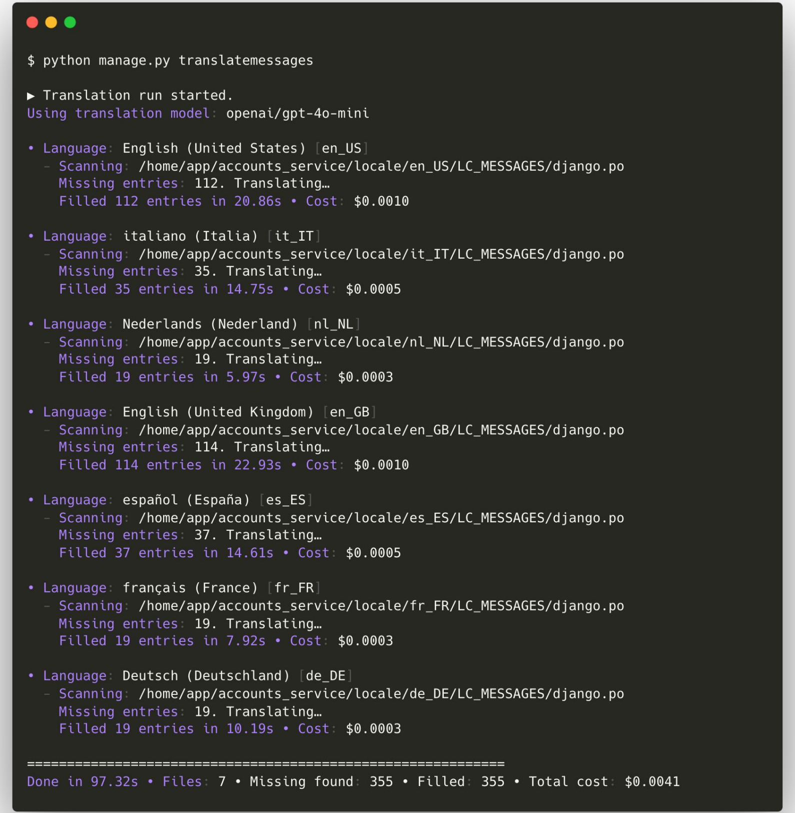
Suggested workflows

Based on your tech stack

Pyth… …tion Configu…
Creat… …ython applicati…

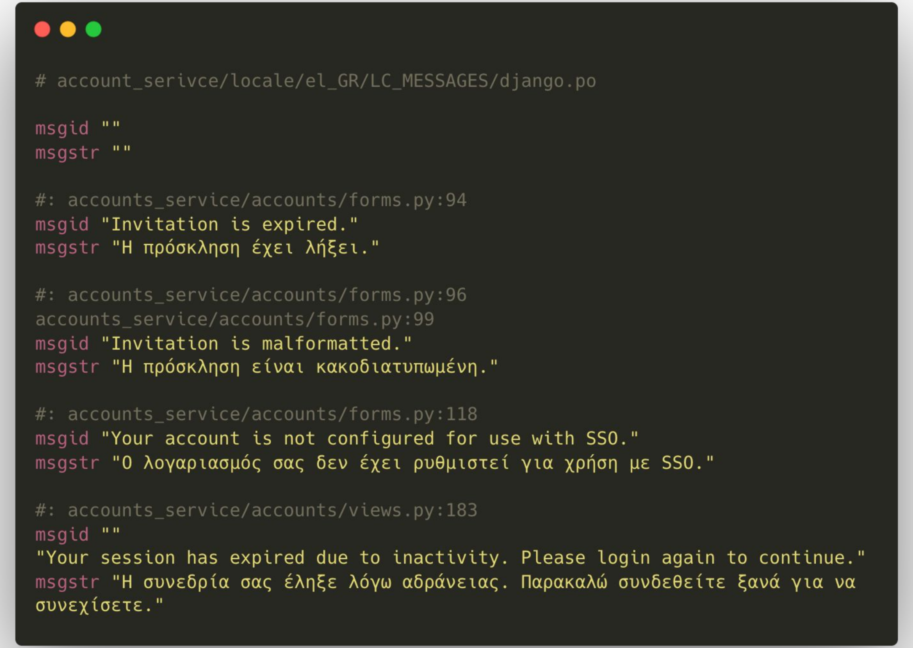
Python pac…

5. manage.py translatemessages ✨



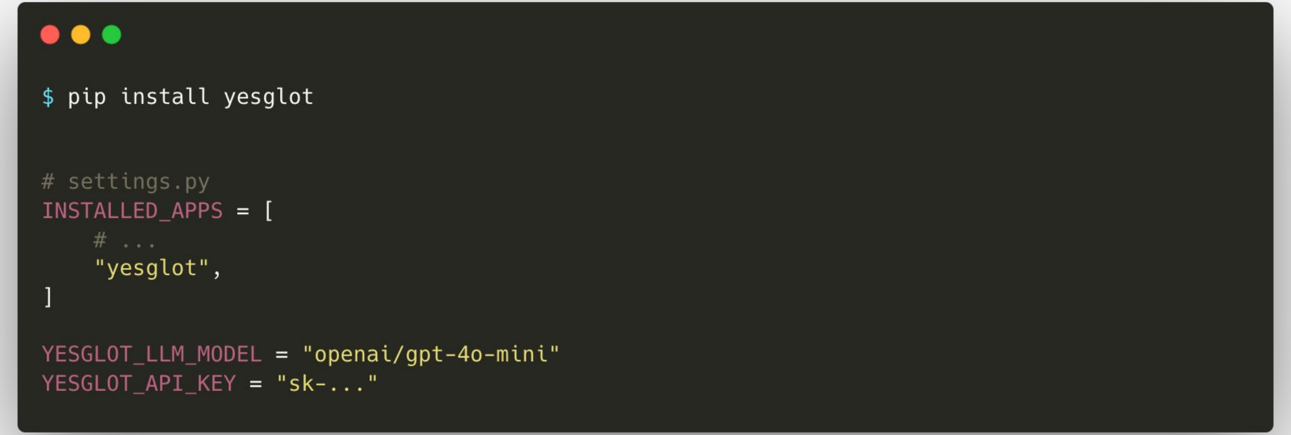
```
$ python manage.py translatemessages

▶ Translation run started.
Using translation model: openai/gpt-4o-mini

• Language: English (United States) [en_US]
  - Scanning: /home/app/accounts_service/locale/en_US/LC_MESSAGES/django.po
    Missing entries: 112. Translating…
    Filled 112 entries in 20.86s • Cost: $0.0010

• Language: italiano (Italia) [it_IT]
  - Scanning: /home/app/accounts_service/locale/it_IT/LC_MESSAGES/django.po
    Missing entries: 35. Translating…
    Filled 35 entries in 14.75s • Cost: $0.0005

• Language: Nederlands (Nederland) [nl_NL]
  - Scanning: /home/app/accounts_service/locale/nl_NL/LC_MESSAGES/django.po
    Missing entries: 19. Translating…
    Filled 19 entries in 5.97s • Cost: $0.0003

• Language: English (United Kingdom) [en_GB]
  - Scanning: /home/app/accounts_service/locale/en_GB/LC_MESSAGES/django.po
    Missing entries: 114. Translating…
    Filled 114 entries in 22.93s • Cost: $0.0010

• Language: español (España) [es_ES]
  - Scanning: /home/app/accounts_service/locale/es_ES/LC_MESSAGES/django.po
    Missing entries: 37. Translating…
    Filled 37 entries in 14.61s • Cost: $0.0005

• Language: français (France) [fr_FR]
  - Scanning: /home/app/accounts_service/locale/fr_FR/LC_MESSAGES/django.po
    Missing entries: 19. Translating…
    Filled 19 entries in 7.92s • Cost: $0.0003

• Language: Deutsch (Deutschland) [de_DE]
  - Scanning: /home/app/accounts_service/locale/de_DE/LC_MESSAGES/django.po
    Missing entries: 19. Translating…
    Filled 19 entries in 10.19s • Cost: $0.0003

============================================================
Done in 97.32s • Files: 7 • Missing found: 355 • Filled: 355 • Total cost: $0.0041
```

5. manage.py translatemessages ✨



```
# account_serivce/locale/el_GR/LC_MESSAGES/django.po

msgid ""
msgstr ""

#: accounts_service/accounts/forms.py:94
msgid "Invitation is expired."
msgstr "Η πρόσκληση έχει λήξει."

#: accounts_service/accounts/forms.py:96
accounts_service/accounts/forms.py:99
msgid "Invitation is malformatted."
msgstr "Η πρόσκληση είναι κακοδιατυπωμένη."

#: accounts_service/accounts/forms.py:118
msgid "Your account is not configured for use with SSO."
msgstr "Ο λογαριασμός σας δεν έχει ρυθμιστεί για χρήση με SSO."

#: accounts_service/accounts/views.py:183
msgid ""
"Your session has expired due to inactivity. Please login again to continue."
msgstr "Η συνεδρία σας έληξε λόγω αδράνειας. Παρακαλώ συνδεθείτε ξανά για να
συνεχίσετε."
```

5. manage.py translatemessages (installation)



```
$ pip install yesglot


# settings.py
INSTALLED_APPS = [
    # ...
    "yesglot",
]

YESGLOT_LLM_MODEL = "openai/gpt-4o-mini"
YESGLOT_API_KEY = "sk-..."
```

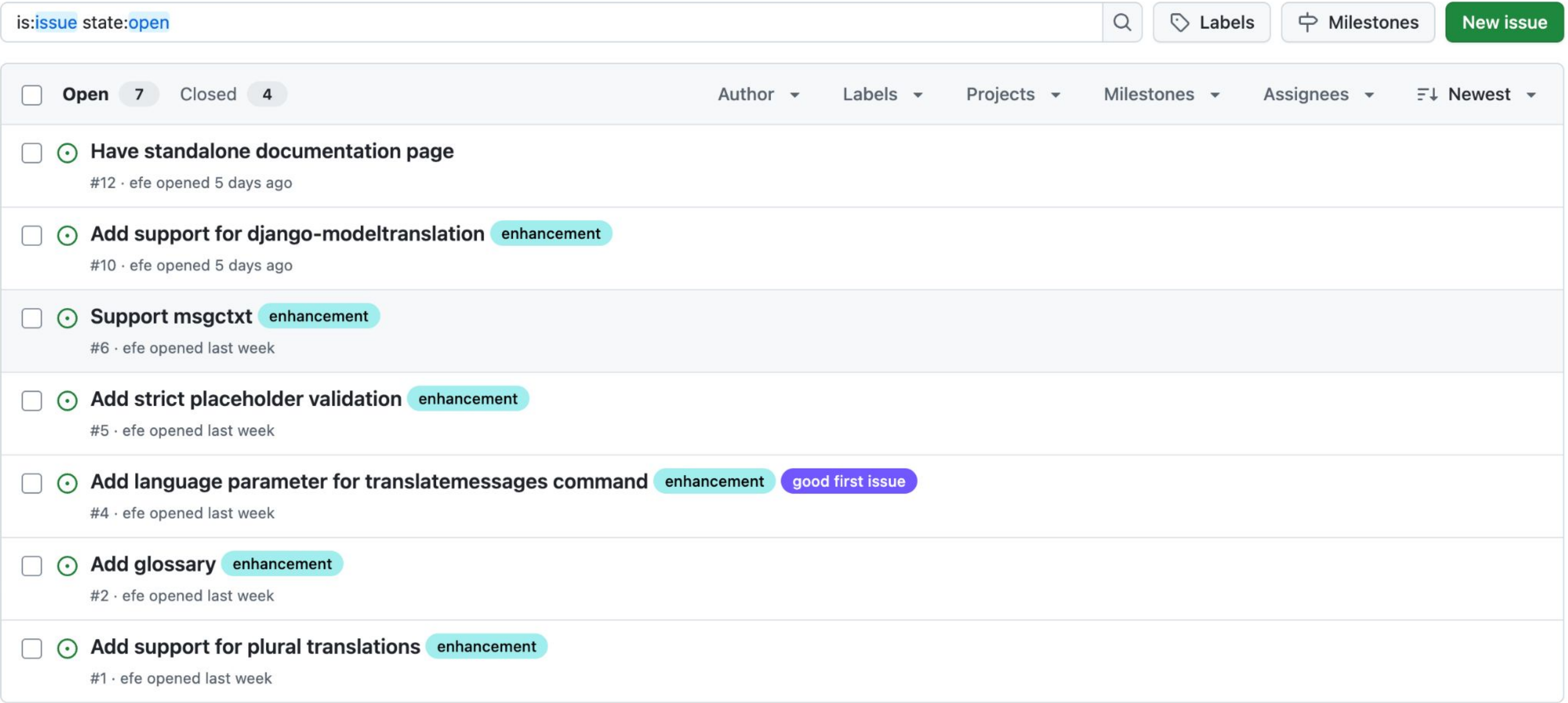
Checklist for multilingual Django application

1. Use gettext for customer facing strings
2. Define supported languages in settings.py
3. Use LocaleMiddleware before SessionMiddleware
4. manage.py makemessages to export translatable files to .po files
5. **manage.py translatemessages ✨**
6. manage.py compilemessages to convert .po to .mo
7. “Boró na milíso pio grígora sta Elliniká.” 🇬🇷

5. manage.py translatemessages ✨

- 🧠 LLM-powered: works with 100+ LLM (OpenAI, Claude, Gemini) models through LiteLLM's unified API
- 🔒 Placeholder-safe: keeps {name}, {{handlebars}}, URLs, and emails intact
- 📦 Django-native: one management command: python manage.py translatemessages
- 🧮 Cost-aware: prints per-file and total cost
- 🧱 Token-safe batching: automatically splits work to avoid context overflows

free to use, free to contribute



is:issue state:open

Have standalone documentation page
#12 · efe opened 5 days ago

Add support for django-modeltranslation enhancement
#10 · efe opened 5 days ago

Support msgctxt enhancement
#6 · efe opened last week

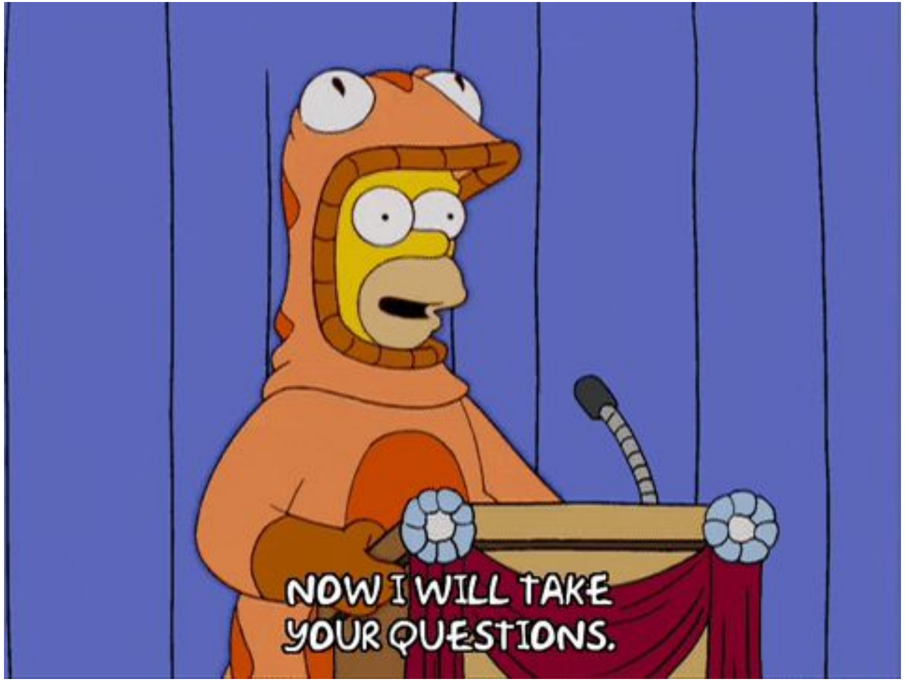
Add strict placeholder validation enhancement
#5 · efe opened last week

Add language parameter for translatemessages command enhancement good first issue
#4 · efe opened last week

Add glossary enhancement
#2 · efe opened last week

Add support for plural translations enhancement
#1 · efe opened last week

Έτοιμο (e-tee-mo)



Multilingual Django Applications with a sprinkle of AI ✨

Efe Öge