

```
(venv) ihne@ihneportal:~/ihne_iroda$ cd ihnecore_os/
(venv) ihne@ihneportal:~/ihne_iroda/ihnecore_os$ ls -l
total 52
-rwxr-xr-x 1 ihne www-data  0 Feb 28 23:11 __init__.py
drwxr-xr-x 2 ihne www-data 4096 Mar 19 22:54 __pycache__
-rwxr-xr-x 1 ihne www-data 258 Mar  1 00:04 admin_urls.py
-rw-rw-r-- 1 ihne ihne   3365 Mar 17 21:51 blocks.py
-rwxr-xr-x 1 ihne www-data 310 Mar  1 00:03 hosts.py
drwxrwxr-x 3 ihne ihne   4096 Mar 18 18:55 migrations
-rw-rw-r-- 1 ihne ihne   1334 Mar 17 21:58 models.py
drwxr-xr-x 3 ihne www-data 4096 Mar 29 07:35 settings
drwxr-xr-x 4 ihne www-data 4096 Feb 28 23:11 static
drwxr-xr-x 2 ihne www-data 4096 Feb 28 23:11 templates
-rwxr-xr-x 1 ihne www-data 2808 Mar 19 22:44 urls.py
-rwxr-xr-x 1 ihne www-data 4470 Mar  3 20:24 views.py
-rwxr-xr-x 1 ihne www-data 403 Feb 28 23:11 wsgi.py
(venv) ihne@ihneportal:~/ihne_iroda/ihnecore_os$ cat blocks.py
from django.utils import timezone
```

```
from esemeny.models import Esemeny
from wagtail import blocks
from wagtail.images.blocks import ImageChooserBlock
```

```
class ServiceCardBlock(blocks.StructBlock):
    title = blocks.CharBlock(label="Kártya címe")
    text = blocks.TextBlock(label="Leírás")
    icon = blocks.CharBlock(required=False, help_text="pl. fa-rocket")
```

```
    link_text = blocks.CharBlock(
```

```
    required=False,  
    label="Hivatkozás neve"  
)
```

```
link_url = blocks.URLBlock(  
    required=False,  
    label="Hivatkozás link"  
)
```

```
class Meta:  
    icon = "placeholder"  
    label = "Szolgáltatás kártya"
```

```
class ServicesBlock(blocks.StructBlock):  
    title = blocks.CharBlock(default="Szolgáltatásaink")  
    cards = blocks.ListBlock(ServiceCardBlock())
```

```
class Meta:  
    template = "blocks/services.html"  
    icon = "list-ul"  
    label = "Szolgáltatások"
```

```
class HeroBlock(blocks.StructBlock):  
    image = ImageChooserBlock()  
    title = blocks.CharBlock()  
    subtitle = blocks.TextBlock()
```

```
class Meta:
```

```
template = "blocks/hero.html"
```

```
icon = "image"
```

```
label = "Hero szekció"
```

```
class FeatureBlock(blocks.StructBlock):
```

```
    image = ImageChooserBlock()
```

```
    title = blocks.CharBlock()
```

```
    text = blocks.TextBlock()
```

```
class FeaturesBlock(blocks.StructBlock):
```

```
    title = blocks.CharBlock(default="Bemutatózás")
```

```
    items = blocks.ListBlock(FeatureBlock())
```

```
class Meta:
```

```
    template = "blocks/features.html"
```

```
    icon = "folder"
```

```
    label = "Features blokk"
```

```
class CTABlock(blocks.StructBlock):
```

```
    text = blocks.CharBlock()
```

```
    button_text = blocks.CharBlock(default="Kapcsolat")
```

```
    button_link = blocks.CharBlock()
```

```
class Meta:
```

```
    template = "blocks/cta.html"
```

```
    icon = "link"
```

```
    label = "CTA blokk"
```

```
class ImageSectionBlock(blocks.StructBlock):
```

```
    image = ImageChooserBlock()
```

```
    text = blocks.TextBlock(required=False)
```

```
class Meta:
```

```
    template = "blocks/image_section.html"
```

```
    icon = "image"
```

```
    label = "Háttér kép szekció"
```

```
class ContactBlock(blocks.StructBlock):
```

```
    title = blocks.CharBlock(default="Kapcsolat")
```

```
    address = blocks.CharBlock()
```

```
    phone = blocks.CharBlock()
```

```
    email = blocks.EmailBlock()
```

```
    extra_info = blocks.TextBlock(
```

```
        required=False,
```

```
        label="További hivatalos adatok"
```

```
    )
```

```
    map_image = ImageChooserBlock(
```

```
        required=False,
```

```
        label="Térkép kép"
```

```
    )
```

```
class Meta:
```

```
    template = "blocks/contact.html"
```

```
icon = "mail"
```

```
label = "Kapcsolat blokk"
```

```
class EventsBlock(blocks.StructBlock):
```

```
    title = blocks.CharBlock(default="Események", required=False)
```

```
    def get_context(self, value, parent_context=None):
```

```
        context = super().get_context(value, parent_context)
```

```
        context["events"] = Esemeny.objects.filter(
```

```
            idopont__gte=timezone.now()
```

```
        ).order_by("idopont")[:4]
```

```
        return context
```

```
class Meta:
```

```
    template = "blocks/events_block.html"
```

```
    icon = "date"
```

```
    label = "Események blokk"
```

```
class StreamStudioBlock(blocks.StructBlock):
```

```
    title = blocks.CharBlock(
```

```
        required=False,
```

```
        default="Stream Stúdió :: Powered by Ihne"
```

```
    )
```

```
class Meta:
```

```
template = "blocks/stream_studio.html"

icon = "media"

label = "IHNE Stream Stúdió"
```

```
(venv) ihne@ihneportal:~/ihne_iroda/ihnecore_os$ cat models.py
```

```
from wagtail.models import Page
from django.shortcuts import redirect

from django.db import models
from wagtail.fields import StreamField
from wagtail import blocks
from wagtail.images.blocks import ImageChooserBlock
from wagtail.admin.panels import FieldPanel

from ihnecore_os.blocks import StreamStudioBlock

from .blocks import (
    HeroBlock,
    ServicesBlock,
    EventsBlock,
    FeaturesBlock,
    CTABlock,
    ImageSectionBlock,
    ContactBlock
)
```

```
class RedirectPage(Page):
```

```
    """Ez az oldal minden kérést a /belepes/ útvonalra irányít"""
```

```
    # Nem lesz saját admin panelje, csak egy "üres" oldal típus
```

```
content_panels = Page.content_panels
```

```
def serve(self, request, *args, **kwargs):
```

```
    return redirect('/belepes/')
```

```
class Meta:
```

```
    verbose_name = "Átírányító oldal"
```

```
# Ihne - IT Lovagok - Digitális Világot Page Builder
```

```
class BaseHomePage(Page):
```

```
    body = StreamField([
```

```
        ("hero", HeroBlock()),
```

```
        ("services", ServicesBlock()),
```

```
        ("events", EventsBlock()),
```

```
        ("features", FeaturesBlock()),
```

```
        ("cta", CTABlock()),
```

```
        ("image_section", ImageSectionBlock()),
```

```
        ("contact", ContactBlock()),
```

```
        ("stream", StreamStudioBlock()),
```

```
    ], use_json_field=True)
```

```
content_panels = Page.content_panels + [
```

```
    FieldPanel("body"),
```

```
]
```

```
(venv) ihne@ihneportal:~/ihne_iroda/ihnecore_os$ cat views.py
```

```
from django.http import JsonResponse
```

```
from django.shortcuts import render
```

```
from django.contrib.auth.decorators import login_required
```

```
from django.contrib.auth.models import User

from django.db import transaction

from django.views.decorators.csrf import csrf_exempt

from django.contrib.auth import update_session_auth_hash

from django.utils import timezone

# Saját appok importjai

from tagsag.models import Tag

from irattar.models import Dokumentum, Mappa

from karrier.models import GyakornokProfil

@login_required

def iroda_asztal(request):

    minden_tag = Tag.objects.all().order_by('teljes_nev')

    if request.user.is_superuser:

        fajlok = Dokumentum.objects.filter(mappa__isnull=True).order_by('-feltoltes_ideje')

        mappak = Mappa.objects.filter(szulo_mappa__isnull=True).order_by('nev')

    else:

        fajlok = Dokumentum.objects.filter(tulajdonos=request.user,

mappa__isnull=True).order_by('-feltoltes_ideje')

        mappak = Mappa.objects.filter(szulo_mappa__isnull=True).order_by('nev')

    # ● IDE HOZD FEL

    profilok = GyakornokProfil.objects.all().order_by('-letrehozva')

    context = {

        'felhasznalo': request.user,

        'minden_tag': minden_tag,
```

```
'irattar_fajlok': fajlok,  
'mappak': mappak,  
'rang': "Elnök" if request.user.is_superuser else "Tag",  
'status_choices': Tag.STATUS_CHOICES,  
'position_choices': Tag.POSITIONS,  
"profilok": profilok,  
}
```

```
context['kategoriak'] = list(Dokumentum.KATEGORIA_CHOICES)
```

```
uj_jelentkezesek = GyakornokProfil.objects.filter(statusz='jelentkezo').count()
```

```
context["uj_jelentkezesek"] = uj_jelentkezesek
```

```
return render(request, 'ihnecore_os/iroda.html', context)
```

```
def get_members(request):
```

```
    # A values() most már az ékezetmentes 'tisztseg' mezőt kéri le
```

```
    members = Tag.objects.all().values('teljes_nev', 'tisztseg', 'statusz', 'belepess_ideje')
```

```
    return JsonResponse(list(members), safe=False)
```

```
@csrf_exempt
```

```
@login_required
```

```
def add_member_api(request):
```

```
    if request.method == 'POST':
```

```
        try:
```

```
            username = request.POST.get('username')
```

```
            teljes_nev = request.POST.get('teljes_nev')
```

```
            tisztseg_adat = request.POST.get('tisztseg')
```

```
            statusz_adat = request.POST.get('statusz')
```

```
            profilkep_adat = request.FILES.get('profilkep')
```

```

datum_str = request.POST.get('belepess_ideje')

belepess_ideje = datum_str if datum_str else timezone.now().date()

if not username or not teljes_nev:
    return JsonResponse({'status': 'error', 'message': 'Hiányzó adatok!'}, status=400)

with transaction.atomic():
    user = User.objects.create_user(username=username,
password='TemporaryPassword123!')
    Tag.objects.create(
        user=user,
        teljes_nev=teljes_nev,
        tisztseg=tisztseg_adat,
        statusz=statusz_adat,
        profilkep=profilkep_adat,
        belepess_ideje=belepess_ideje,
    )
    return JsonResponse({'status': 'success'})
except Exception as e:
    return JsonResponse({'status': 'error', 'message': str(e)}, status=400)

```

@login_required

```

def update_profile_api(request):
    if request.method == 'POST':
        try:
            tag_profil = request.user.iroda_profil
            tag_profil.teljes_nev = request.POST.get('teljes_nev')
            tag_profil.telefonszam = request.POST.get('telefonszam')
            tag_profil.lakcim = request.POST.get('lakcim')

```

```
if request.FILES.get('profilkep'):
    tag_profil.profilkep = request.FILES.get('profilkep')

    tag_profil.save()

    return JsonResponse({'status': 'success'})

except Exception as e:
    return JsonResponse({'status': 'error', 'message': str(e)}, status=400)
```

@login_required

```
def change_password_api(request):
    if request.method == 'POST':
        user = request.user
        old_pass = request.POST.get('old_password')
        new_pass = request.POST.get('new_password')

        if not user.check_password(old_pass):
            return JsonResponse({'status': 'error', 'message': 'A régi jelszó hibás!'}, status=400)

        user.set_password(new_pass)
        user.save()
        update_session_auth_hash(request, user)
        return JsonResponse({'status': 'success', 'message': 'Jelszó módosítva!'})
```

```
(venv) ihne@ihneportal:~/ihne_iroda/ihnecore_os$
```

```
(venv) ihne@ihneportal:~/ihne_iroda/ihnecore_os$ cat urls.py
```

```
from . import views
from django.conf import settings
from django.urls import include, path
from django.contrib import admin
from wagtail.admin import urls as wagtailadmin_urls
```

```
from wagtail import urls as wagtail_urls

from wagtail.documents import urls as wagtaildocs_urls

from search import views as search_views

# Itt importáljuk be az Iroda nézetét közvetlenül, hogy ne legyen körbe-hivatkozás
from .views import iroda_asztal, get_members

from django.contrib.auth import views as auth_views # <--- EZ A SOR HIÁNYZIK!

from irattar import views as irattar_views

urlpatterns = [

    path("django-admin/", admin.site.urls),

    path("admin/", include(wagtailadmin_urls)),

    path("documents/", include(wagtaildocs_urls)),

    path("search/", search_views.search, name="search"),

    # Az OS asztala közvetlenül ide kerül

    path("belepes/", auth_views.LoginView.as_view(template_name='ihnecore_os/login.html'),
name='login'),

    path("iroda/", iroda_asztal, name="iroda_asztal"),

    path('api/members/', get_members, name='get_members'),

    path('api/members/add/', views.add_member_api, name='add_member_api'), # Ez a sor kell!

    # Profil adatok (név, lakcím, kép) frissítése

    path('api/update-profile/', views.update_profile_api, name='update_profile_api'),

    # Biztonságos jelszóváltás

    path('api/change-password/', views.change_password_api, name='change_password_api'),

    path('logout/', auth_views.LogoutView.as_view(), name='logout'),

    # Ezzel kötjük be az irattar app saját útvonalait

    path('irattar/get-contents/', irattar_views.get_folder_contents, name='get_contents'),
```

```
path('irattar/upload/', irattar_views.upload_to_folder, name='upload_to_folder'),
path('irattar/create-folder/', irattar_views.create_folder, name='create_folder'),
path('irattar/delete-item/', irattar_views.delete_item, name='delete_item'),
path('irattar/rename-item/', irattar_views.rename_item, name='rename_item'),
path('irattar/move-item/', irattar_views.move_item, name='move_item'),
path('irattar/attach-item/', irattar_views.attach_item, name='attach_item'),

# Események beemelése
path('esemeny/', include('esemeny.urls')),

# Karrier Kezelő beemelése
path('karrier/', include('karrier.urls')),

# Hírközvetítő rendszer beemelése
path("hirek/", include("hirek.urls")),

# TudásKözpont rendszer beemelése
path("tudaskoz/", include("tudaskoz.urls")),

# A Wagtail oldalak (ez legyen az utolsó!)
path("", include(wagtail_urls)),
]

if settings.DEBUG:
    from django.conf.urls.static import static
    from django.contrib.staticfiles.urls import staticfiles_urlpatterns
    urlpatterns += staticfiles_urlpatterns()
    urlpatterns += static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

```
(venv) ihne@ihneportal:~/ihne_iroda/ihnecore_os$
```