PIKA

AKTUALNOSTI S PODROČJA IT STORITEV | november 2026, št. 5, letnik XXIX



ALI INTRANET ŠE POTREBUJE MENI, ČE IMA AI POMOČNIKA?

stran 7

SHAREPOINT ONLINE PREMIUM – OCR, IMAGE TAGGING IN ELEKTRONSKO PODPISOVANJE DOKUMENTOV

stran 15

NAPADI TIPA MAN-IN-THE-MIDDLE (MITM)

stran 27

KO SE SPREMINJA DELO, SE MORA RAZVIJATI TUDI ZNANJE

stran 40

Xnet
vaš zanesljiv IT partner

INFRASTRUKTURA • RAZVOJ APLIKACIJ IN SPLETNIH STRANI • OFFICE TEČAJI • SHAREPOINT • ITIL • SQL • BI

IT strokovnjaki

- Infrastruktura in oblačne storitve
- Podatkovne baze in analitika
- Poslovne aplikacije in avtomatizacija
- Upravljanje in varnost podatkov
- Upravljanje uporabnikov in naprav
- Razvoj in integracije
- Umetna inteligenca in avtomatizacija procesov

Xnet
vaš zanesljiv IT partner

Pokrivamo širok nabor izobraževanj za IT strokovnjake in končne uporabnike. Vsa izobraževanja lahko prilagodimo potrebam vaše organizacije.

Kompas Xnet d.o.o., Stegne 7, 1000 Ljubljana
01 5136 990
info@kompas-xnet.si
kompas-xnet.si

- Osnove Microsoft 365 & Office
- Excel & Power BI
- SharePoint & OneDrive
- Umetna inteligenca
- Digitalna higiena in varnost
- Power Automate & Power Apps
- Projektni management

Končni uporabniki

KAZALO

KOLOFON

ISSN: 1408-7863
Kompas Xnet d.o.o.
Stegne 7
1000 Ljubljana

Telefon: 01 5136 990
Email: info@kompas-xnet.si
Web: https://www.kompas-xnet.si

Direktorica:
Branka Slinkar

Urednik in oblikovalec:
Aleš Frelih

Člani uredništva:
Branka Slinkar, Aleš Lipušček, Robi Vončina, Klemen Vončina, Domen Gričar, Petra Militarev, Aida Kalender Avdić, Lenart Svetek, Dijana Babić, Žiga Brinc, Jan Žugelj

POVEŽIMO SE:

facebook.com/KompasXnet/

twitter.com/kompasxnet

linkedin.com/company/kompas-xnet-d-o-o-/

linkedin.com/showcase/thriveconf/

youtube.com/channel/UCfoMnj355AbRMeG66ePS8BQ

7 **UX/UI**

Ali intranet še potrebuje meni, če ima AI pomočnika?

Aleš Frelih
UX/UI oblikovalec, projektni vodja

11 **ADMINISTRACIJA**

Powershell kotiček

Aleš Lipušček
MCP, MCTS, MCITP

15 **SHAREPOINT**

SharePoint Online Premium – OCR, Image Tagging in elektronsko podpisovanje dokumentov

Robi Vončina
MVP, MCT, MCITP, MCSA, MCTS

20 **MICROSOFT OFFICE**

Excel – Seznami in matrike

Klemen Vončina
Microsoft Office Specialist Master, MCT

23 **RAZVOJ**

En HttpClient, prosim: zakaj ga v .NET ne ustvarjamo za vsak HTTP klic

Lenart Svetek
Frontend programer

27 **RAZVOJ**

Napadi tipa Man-in-the-Middle (MitM)

Žiga Brinc
Frontend programer

29 **RAZVOJ**

Dinamično nalaganje vsebine s pomočjo Intersection Observer API

Domen Gričar
Frontend programer, MCSD, MCSA, MCT

KAZALO

33 PIKIN VITAMIN

Ali ste vedeli?

Dijana Babić
Računovodstvo

40 DRUGO

Ko se spreminja delo, se mora razvijati tudi znanje

Petra Militarev
Vodja izobraževanj

45 DOGODKOVNIK

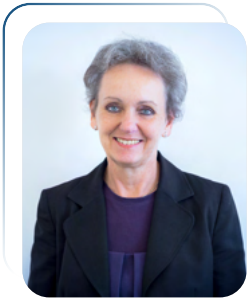
Tečaji, webinarji, seminarji in drugi pomembni prihajajoči dogodki

Kolofon

“

Upihnila desetko, hvala!	Petra, Klemen
Očitno deluje	Aida
Dela kot urca	Domen
V A'dam po nove ideje in znanje	Robi
Nova izdaja, nove barve	Aleš F.
Že kar na polno	Žiga
Cel teden o AI	Petra
Miška na hrbtu	Aleš
Prvi že sedijo	Jan
S CMS sta si na ti	Lenart
Malo zatišja	Dejan
Zdaj so kar v vrsti	Miha
Copilot ga na polno okupira	Tomaž
Za tečaj zdaj pripotuje	Jože





UVODNIK

BRANKA SLINKAR
Direktorica

Navkljub turobnim novemberskim dnem, narava občasno postreže z veličastnimi prizori, ki nam pokažejo, kako se s skupnimi močmi premaguje, praktično nepredstavljivo. Selitev žerjavov proti zahodu, ki smo jo pretekle dni lahko opazovali nad našimi kraji, osupne z neverjetno sposobnostjo ptic selivk za navigacijo, njihovo vztrajnost ter moč in neverjetno sodelovanje na dolgem potovanju.

Mi pa medtem skušamo zaključiti še vse tisto, kar smo si zadali za letošnje leto. Ostalo je le še malo časa, saj se »veseli december« naglo približuje, a s skupnimi močmi je marsikaj mogoče.

Če potrebujete nova znanja ali pa želite poglobiti in razširiti svoje IT kompetence, imamo do konca leta potrjenih še nekaj zanimivih tečajev. Počastite nas s svojo udeležbo na katerem od njih. **Koledar tečajev** >

Obenem pa vas ponovno vabimo, da nam zaupate svoje izobraževalne potrebe in plane za v prihodnje, saj nam s tem pomagate pri oblikovanju našega programa, hkrati pa lažje uskladimo termine, ki so bolj ugodni zainteresiranim kandidatom. Na ta način se izognemo neželenim odpovedim ali prestavljanju termina.
Že vnaprej srčna hvala za vsako vašo informacijo.



Radi bi vas opozorili še na brezplačne spletne dogodke, ki jih organiziramo v okviru **LLPA združenja**:

- **Cyber Security Day 2026**, ki bo potekal 3. decembra od 9 – 18 ure. **Prijave >**
- AI Festival – 17. – 18. februar 2027
- Digital Skills Cup – februar – maj 2027

Podrobnosti in prijave bodo v prihodnjih tednih na naši spletni strani in v novičkah.

Če vas zanimajo podrobnosti o tem kaj agenti so, kako jih uporabljati in na kaj morajo biti podjetja posebej pozorna, preden jih na široko vpeljejo v okolje, je **Agent Fest** kot naročen za vas. Dogodek je brezplačen in bo potekal online **9. decembra 2026**.

Prijave so obvezne: **Register for Agent Fest >**

Večino strokovnjakov Agent Fest-a ste imeli že priložnost spoznati na odru Thrive konference, ki vsako leto nudi izjemno priložnost, da se pogovorite z vrhunskimi strokovnjak s celega sveta, poglobite svoje znanje in spoznate trende in novosti v IT industriji.

Ponosni smo, da se nam izteka uspešno poslovno leto, posebej kar zadeva rešitve in storitve. Prevladujejo SharePoint projekti, ki so bili številni, zelo raznoliki in zanimivi. Tudi letos smo imeli naročnike iz tujine, kar nas zelo veseli.

Zaradi konca podpore za pretekle različice, smo izvedli kar nekaj tehničnih nadgradenj na SP 2022 SE in migracij v Microsoft 365. Če še vedno uporabljate SharePont 2019 ali starejše, pohitite z nadgradnjo, preden vas »ujamejo« težave.

Zelo nam je žal, da izobraževanje ne dosega pričakovanih ciljev. Potreb po dodatnem znanju je zelo veliko, še posebej na področju umetne inteligence in informacijske varnosti. Žal pa podjetja ne planirajo dovolj sredstev v ta namen, saj računajo na to, da bosta spletno učenje in izobraževalne platforme dala dobre rezultate. V praksi se izkaže, da je učenje s pomočjo video vsebin zelo koristno in v veliko pomoč, ne more pa nadomestiti učenja s predavateljem. Delavnice v živo (v učilnici ali prek spleta) prinašajo še dodatne prednosti za udeležence, saj se predavatelj prilagodi skupini, odgovarja na številna vprašanja, da napotke za delo in predstavi dobre prakse, deli lastne izkušnje...



Kaj, ko bi letos pod smrekico pustili ovojnico z željo, katerega šolanja bi se radi udeležili v prihodnjem letu. Kdo ve, morda pa čarobni praznični čas vašo željo posuje s čudežnim prahom in se vidimo na našem tečaju.

Ne pozabite, odlična priložnost za nova znanja je tud vaša in naša Thrive konferenca. Po letu 2019 se ponovno vračamo v prelepo okolje Postojnske jame. Dva polna dneva vam bodo na voljo izjemni strokovnjaki, ki si radi vzamejo čas za oseben pogovor z udeleženci. Vi pa do takrat pripravite čim več vprašanj zanje.

THRIVE KONFERENCA

Vabimo vas, da si v koledarju rezervirate datum **16. Thrive konference**, ki bo **25. – 26. maja 2027 v hotelu Jama, Postojna** in vstopnice kupite že sedaj, ko je na voljo **kar 30 % popust.**

Morda pa se tokrat odločite, da vstopnici dodate še **predkonferenčno delavnico**, ki bo **24. maja**?

Ko boste iskali partnerja za vaše projekte na Microsoft tehnologijah, ne pozabite, da imamo v ekipi izjemne

strokovnjake, z ogromno izkušnjami na On-prem in online.

Lahko vam zgolj svetujemo, kako začeti ali pa vam izdelamo rešitev “na ključ”.

Reference >

Microsoft Solutions Partner
Modern Work

Microsoft Solutions Partner
Data & AI
Azure

Microsoft Solutions Partner
Infrastructure
Azure

Microsoft Solutions Partner
Digital & App Innovation
Azure

Microsoft Solutions Partner
Security

Naj bo **Xnet vaša prva izbira, ko gre za IT rešitve in storitve**. Nismo številčna ekipa, imamo pa zato **»jagodni izbor« strokovnjakov**, ki vas ne bo pustil na cedilu.

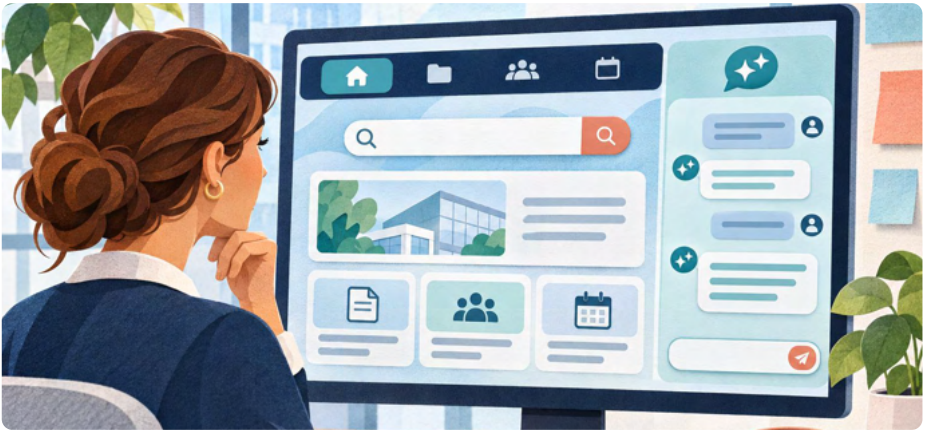


Lep dan in vse dobro vam želimo
Branka Slinkar



ALI INTRANET ŠE POTREBUJE MENI, ČE IMA AI POMOČNIKA?

Aleš Frelih
UX/UI oblikovalec, projektni vodja
ales.frelih@kompas-xnet.si



AI pomočnik obljublja, da bomo informacije na intranetu našli kar s pomočjo svojih vprašanj. Toda ali lahko pogovor nadomesti navigacijski meni, iskalnik in hitre povezave? Pri načrtovanju portalov je smiselno

ohraniti več odprtih poti ter vsako uporabiti tam, kjer zaposlenemu najbolj pomaga oz. kjer zares opazimo potrebo po tem.

Predstavljajmo si običajno delovno jutro. Zaposleni želi odpreti imenik, poiskati obrazec za službeno pot in preveriti navodila za selitev v nove prostore. Vse to želi opraviti na intranetu, vendar so njegove potrebe različne. Enkrat pozna cilj, drugič išče dokument, tretjič potrebuje razlago. Ali je pogovor res najboljša pot v vseh treh primerih? Ja in ne. Poglejmo si to skozi naslednje definicije.

Meni pokaže tudi tisto, česar ne iščemo

Dober meni zaposlenemu omogoči pregled nad vsebino portala. Nov sodelavec lahko tako odkrije usposabljanja, ugodnosti ali projektne prostore, čeprav zanje še ni vedel. Prazno pogovorno polje mu takšnega pregleda samo po sebi ne ponudi. Če ne ve, kaj je na voljo, težko oblikuje vprašanje, ki ga bo pripeljalo na željeni cilj.

Pri znanih - vsakodnevnih ponavljajočih se opravilih je neposredna povezava pogosto povsem dovolj. Če zaposleni vsak dan odpira isti imenik, je smiselno ohraniti viden gumb. Vsakokratno pisanje zahteve in čakanje na odgovor lahko takšno rutino podaljšata. Meni zato ostaja uporaben tudi na portalu z AI pomočnikom.

Tri potrebe, tri možne poti

Pri zahtevi »Odpri imenik zaposlenih« uporabnik pozna cilj. Pomagata mu jasna navigacija ali hitra povezava. Če išče obrazec za službeno pot, želi morda pregledati več zadetkov ter preveriti njihov naslov, datum in vrsto. Tu ima pomembno vlogo iskalnik s filtri.

Drugače je pri vprašanju »Kako pripravim dokumentacijo za selitev svojega oddelka?« Odgovor je lahko



razpršen po več navodilih. Dobro zasnovan pomočnik lahko iz dovoljenih virov pripravi razumljiv povzetek in poveže ustrezne dokumente. Zaposleni dobi izhodišče za delo, podrobnosti pa preveri v izvirniku.

Raziskave NN/Group o spletnih AI pomočnikih iz leta 2026 opozarja, da uporabniki pogosto ne prepoznajo njihove dodatne vrednosti. Pri nekaterih nalogah jim obstoječa navigacija ali iskanje omogočata učinkovitejšo - hitrejšo pot. Čeprav raziskava obravnava spletna mesta, je njeno vprašanje uporabno tudi za intranet portale, kjer se je prvo potrebno vprašati: "Katero konkretno težavo bo pomočnik rešil bolje?"

Od odgovora do opravljenega dela

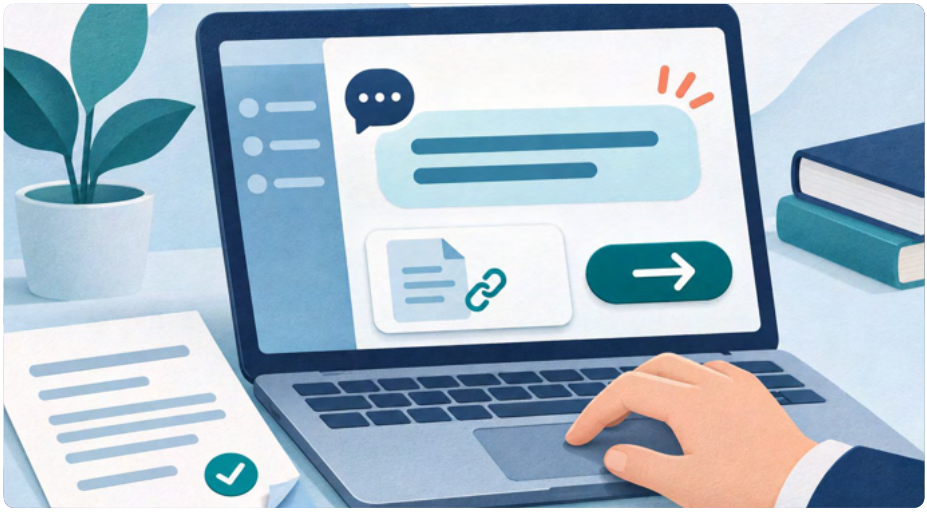
Razlaga postopka še ne pomeni, da je naloga opravljena. Odgovoru o službeni poti naj zato sledi povezava do ustreznega obrazca. Če pomočnik omogoča tudi oddajo zahtevka, mora uporabnik pred potrditvijo videti podatke in nato jasno potrdilo o uspehu. Besedilo v pogovoru ne sme ustvarjati vtisa, da je bil zahtevek oddan, če je pomočnik le opisal postopek.

Zaupanje potrebuje preverljiv odgovor

Na intranetu lahko odgovor vpliva na dejansko ravnanje zaposlenega. Ob povzetku zato potrebujemo povezavo do dokumenta ali strani, iz katere izhaja, po možnosti tudi podatek o veljavnosti. Vir mora podpirati zapisano trditev, saj sama prisotnost povezave še ni zagotovilo pravilnosti.

Pri nejasnih ali nasprotujočih si navodilih naj pomočnik opozori na omejitev in ponudi kontakt odgovorne osebe.

Pomembno je razlikovati tudi med nedostopno in neobstoječo informacijo. Microsoft navaja, da odgovori agentov v SharePointu



upoštevajo uporabnikova dovoljenja za vire. Dostop do pomočnika torej sam po sebi ne omogoči dostopa do vseh dokumentov, ampak samo do tistih za katere ima uporabnik nastavljene pravice. Pri prilagojenih rešitvah moramo takšno ravnanje izrecno načrtovati in preveriti.

Jasna vloga namesto obljube, da zna vse

Poimenovanje »AI pomočnik« še ne pove, pri čem nam lahko pomaga. Bolj uporabna je kratka razlaga, denimo »Pomagam pri vprašanjih o selitvi«, in nekaj ustreznih začetnih vprašanj. Tudi smernice NN/group priporočajo jasno predstavitev zmožnosti in smiselne predloge vprašanj.

Pomočnik naj bo dosegljiv tam, kjer ga zaposleni potrebuje, brez prekrivanja pomembnih gumbov. Načrtovati moramo odpiranje s tipkovnico, berljiva sporočila in jasno zapiranje. Če storitev ni na voljo ali nam v danem trenutku ne more pomagati, naj ostaneta vidni povezavi do iskanja (klasičen SharePoint iskalnik) in podpore. Zaposleni mora imeti možnost nadaljevati delo.

Kaj to pomeni za SharePoint portal?

Pri SharePoint Online so agenti del Microsoftove ponudbe, njihova uporaba pa je odvisna tudi od ustreznega licenciranja oziroma obračunavanja po porabi. Pri portalu On-Premise moramo možnosti povezovanja z izbrano AI rešitvijo preveriti posebaj. Enake uporabniške zamisli zato ne pomenijo enake tehnične izvedbe. Že na začetku določimo vire, prijavo, dovoljenja in obseg pomoči.

AI prav tako ne odpravi potrebe po urejenih vsebinah. Če sta objavljeni dve nasprotujoči si navodili, mora nekdo določiti veljavno različico in njenega skrbnika. Uvajanje pomočnika je zato primeren trenutek za pregled vsebin, zlasti tistih, na katerih bodo temeljili njegovi odgovori.

Prototip, preizkus, odločitev

V Figmi lahko načrtujemo zaslone in prehode med menijem, iskalnikom, pogovorom ter dokumenti. S prototipom preverimo, ali ljudje pomočnika opazijo, razumejo njegovo vlogo in najdejo naslednji korak. Pravilnost odgovorov, upoštevanje dovoljenj in dejansko izvajanje dejanj pa preverimo z delujočo rešitvijo.

Zaposlenim ponudimo nekaj resničnih nalog: najdi kontakt, poišči obrazec, poglej dnevne naloge. Primerjajmo uspešnost, čas ter potrebo po dodatni pomoči. Opazujmo tudi, ali prepoznajo napačen ali nepopoln odgovor. Število začetih pogovorov samo zase ne pokaže, ali je njihovo delo lažje.

Torej glavni namen vpeljave AI pomočnika v SharePoint intranet je zaposlenim omogočiti, da z vprašanjem v naravnem jeziku hitreje pridejo do ustreznih informacij in razumljive razlage internih postopkov. Pomočnik dopolnjuje meni in iskalnik ter uporabnika usmeri do preverljivega vira, ustreznega obrazca ali naslednjega koraka, da lažje opravi svoje delo.

Ali intranet torej še potrebuje meni? Za portal z raznolikimi vsebinami in opravili ga je vsekakor smiselno ohraniti. AI lahko doda uporabno pot do razlage in usmeritve, meni pa zagotavlja pregled in neposreden dostop.

Pri načrtovanju naj odloča naloga zaposlenega: za znan cilj jasna povezava, za pregled zadetkov iskanje, za zahtevnejše vprašanje pomočnik s preverljivimi viri. Kakovost intraneta se pokaže takrat, ko zaposleni uspešno opravi svoje delo.



POWERSHELL KOTIČEK

Aleš Lipušček
MCP, MCTS, MCITP
ales.lipuscek@kompas-xnet.si

V prejšnji Piki smo se prvič srečali z AST-jem. Začeli smo z iskanjem zastarelih ukazov WMIC in pri tem ugotovili, da zna PowerShell svojo kodo pregledovati precej drugače kot običajno besedilo. Ne vidi samo besed in vrstic, ampak tudi ukaze, izraze in druge dele programa.

Takrat smo uporabili CommandAst, torej vozlišča, ki predstavljajo ukaze, tokrat pa bomo pogledali nekaj drugega.

Skozi leta se mi je nabralo kar precej PowerShell skript. Nekatere uporabljam redno, druge sem napisal za enkratno opravilo, tretje pa so se večkrat preselile z računalnika na računalnik. In v njih je cel kup funkcij.

Za večino še približno vem, čemu služijo, za nekatere pa bi moral najprej odpreti datoteko in malo pobrskati. Pa sem pomislil: kaj če bi iz vseh teh skript naredil seznam funkcij? Ne samo njihovih imen, ampak tudi parametrov, ki jih sprejemajo.

Pravzaprav imamo iz prejšnjega članka večino potrebnega znanja že na voljo.

Začnimo z majhno skripto, ki jo shranimo kot C:\Scripts\ServerTools.ps1.

```
function Get-ServerInfo {
param(
[string]$ComputerName,
[switch]$IncludeDisks
)
Get-CimInstance
Win32_OperatingSystem `
-ComputerName $ComputerName
}
function Get-PrinterStatus {
param(
[string]$ComputerName
)
Get-CimInstance Win32_Service `
```

```
-ComputerName $ComputerName `
-Filter "Name='Spooler'"
}
```

Imamo dve funkciji. Prva ima dva parametra, druga enega. Kaj počneta, za naš primer niti ni posebej pomembno.

Nas zanima, ali lahko PowerShell sam prepozna njuni definiciji in parametre.

Spomnimo se parserja iz prejšnjega članka.

```
$tokens = $null
$errors = $null
$ast=
[System.Management.Automation.Language.Parser]::ParseFile(
'C:\Scripts\ServerTools.ps1',
[ref]$tokens,
[ref]$errors
)
```

Prejšnjič smo iskali objekte tipa CommandAst. Tokrat nas zanima: FunctionDefinitionAst.

Ime je precej zgovorno. Gre za del AST-ja, ki predstavlja definicijo funkcije.

```
$functions = $ast.FindAll({
param($node)
$node -is [
System.Management.Automation.Language.FunctionDefinitionAst ]
}, $true)
$functions | ForEach-Object {
$_.Name
}
```

Rezultat: Get-ServerInfo, Get-PrinterStatus. Parser je našel obe funkciji.

Pri tem ni iskal besede function v besedilu. Vrnil nam je dejanski definiciji funkcij, tako kot nam je v prejšnjem članku vračal dejanske ukaze. Ampak samo seznam imen ni ravno nekaj, zaradi česar bi se posebej navduševal nad AST-jem.

Poglejmo še parametre.
Tukaj naletimo na majhno zanimivost. PowerShell namreč dovoljuje dva načina njihovega zapisovanja.

Običajno uporabljamo:

```
function Test-One {
param([string]$Name)
}
```

Lahko pa napišemo tudi:

```
function Test-Two([string]$Name) {
}
```

Oba zapisa sta veljavna, AST pa parametre shrani na različnih mestih. Pri prvem jih najdemo v: $function.Body.ParamBlock.Parameters, pri drugem pa v: $function.Parameters

To je ena tistih podrobnosti, na katere pri običajnem pisanju skript niti ne pomislimo. Če pa želimo pregledati večjo zbirko tujih ali starih skript, moramo upoštevati oba načina.

Zato bomo parametre pobrali z obeh

mest.

```
foreach ($function in $functions) {
$names = @(
foreach ($p in $function.Parameters) {
$p.Name.VariablePath.UserPath
}
foreach ($p in
$function.Body.ParamBlock.Parameters) {
$p.Name.VariablePath.UserPath }
)
[pscustomobject]@{
Function = $function.Name
Parameters = $names -join ', ' }
}
```

Rezultat:

```
Function    Parameters
----------- ----------
Get-ServerInfo ComputerName, IncludeDisks
Get-PrinterStatus ComputerName
```

In zdaj imamo že nekaj precej bolj uporabnega.

Mimogrede, izraz VariablePath.UserPath je samo način, kako iz AST-jeve predstavitve parametra dobimo njegovo ime, brez znaka $ in drugih podrobnosti.

Ničesar še vedno nismo izvedli. Nismo se povezali z nobenim strežnikom in nismo preverjali tiskalnikov. Samo pogledali smo, kaj je v skripti napisano.

Ampak dve funkciji v eni datoteki nista ravno velik dosežek. Poglejmo zdaj celoten direktorij.

Princip je enak kot zadnjič: poiščemo datoteke, vsako razčlenimo v AST in iz nje poberemo podatke, ki nas zanimajo.

```
$inventory = foreach ($file in Get-ChildItem C:\Scripts -Filter *.ps1 -Recurse) {
$tokens = $null
$errors = $null
$ast =
[System.Management.Automation.Language.Parser]::ParseFile(
$file.FullName,
[ref]$tokens,
[ref]$errors
)
$functions = $ast.FindAll({
param($node)
$node -is [
System.Management.Automation.Language.FunctionDefinitionAst
]
}, $true)
foreach ($function in $functions) {
$names = @(
foreach ($p in $function.Parameters) {
$p.Name.VariablePath.UserPath
}
foreach ($p in
$function.Body.ParamBlock.Parameters) {
$p.Name.VariablePath.UserPath
}
)
[pscustomobject]@{
File = $file.FullName
Function = $function.Name
Line =
$function.Extent.StartLineNumber
Parameters = $names -join ', '
}
```

```
}
}
```

$inventory | Format-Table -AutoSize:
Tokrat smo dodali še Extent.StartLineNumber, ki ga poznamo iz prejšnjega članka. Tako ne dobimo samo imena datoteke in funkcije, ampak tudi vrstico, kjer je funkcija definirana.

Če nas zanima večja zbirka skript, je to precej priročno.

In ker bomo takšen seznam verjetno raje pregledovali v Excelu kot v PowerShell konzoli, ga še izvozimo.

```
$inventory |
Sort-Object File, Function |
Export-Csv .\Functions.csv -
NoTypeInformation -Encoding UTF8
```

Dobili smo preprost inventar funkcij. Seveda to še ni popolna dokumentacija naše kode. Ne vemo, ali funkcije pravilno delujejo, kakšne rezultate vračajo in katere druge funkcije kličejo.

Naš mali pregledovalnik ima tudi nekaj omejitev. Parametre, ki jih PowerShell samodejno doda naprednim funkcijam, na primer -Verbose, smo pustili pri miru. Zanimajo nas tisti, ki smo jih v kodi dejansko deklarirali.

Prav tako nam AST pove, kje je funkcija zapisana, ne pa nujno, katera njena definicija bo ob izvajanju dejansko uporabljena.

V večji zbirki imamo lahko tudi več funkcij z istim imenom.

Ampak ravno to je lahko ideja za nadaljevanje. Ker imamo zdaj seznam funkcij, lahko precej hitro poiščemo podvojena imena. Ali pa funkcije, ki imajo nenavadno veliko parametrov.

Morda tudi tiste, ki so zrasle na nekaj sto vrstic.

In če AST že zna prepoznati funkcije, parametre in ukaze, se pojavi še zanimivejše vprašanje.

Ali lahko ugotovimo, katere funkcije kličejo katere druge funkcije? To bi nam pri kakšni stari zbirki skript utegnilo priti še kako prav.

Zablestite pri delu z Microsoft Office

Spoznajte, kako učinkovito uporabljati Word, Excel, PowerPoint, Outlook in druge aplikacije, ki vam bodo olajšale delo in povečale vašo učinkovitost.



Preverite naše Microsoft tečaje ›



SHAREPOINT ONLINE PREMIUM – OCR, IMAGE TAGGING IN ELEKTRONSKO PODPISOVANJE DOKUMENTOV

Robi Vončina
MVP, MCT, MCITP, MCSA, MCTS
robi.voncina@kompas-xnet.si

V prvem delu serije smo pogledali, kaj SharePoint Online Premium sploh je in katere dodatne možnosti za upravljanje ter obdelavo dokumentov prinaša v Microsoft 365. V drugem delu pa smo se podrobneje posvetili Autofill Columns oziroma samodejnemu pridobivanju metapodatkov iz dokumentov.

Preden nadaljujem, moram pri drugem delu narediti manjši, vendar pomemben popravek.

Popravek glede Autofill Columns

V drugem delu sem Autofill Columns opisal kot storitev, za katero je treba vzpostaviti Pay-As-You-Go obračunavanje prek Azure naročnine.

Microsoft je medtem spremenil način licenciranja in vključevanja Autofill Columns v Copilot in SharePoint. Osnovna uporaba Autofill Columns je zdaj vključena v Microsoft 365 Copilot licenco, medtem ko se naprednejša uporaba oziroma obdelava vsebine v večjem obsegu lahko obračunava prek Copilot Credits. Za site-wide real-time Autofill je še vedno potrebno usage-based obračunavanje.

https://learn.microsoft.com/en-us/sharepoint/copilot-in-sharepoint-get-started

To je tudi dober primer, kako hitro se Microsoft 365 in Copilot trenutno razvijata. Funkcionalnost lahko ostane podobna, način licenciranja, konfiguracije in umeščanja v Microsoftove produkte pa se lahko precej hitro spremeni.

V zadnjem delu serije bomo pogledali še tri funkcionalnosti, ki se mi zdijo zanimive predvsem z vidika dejanske uporabe SharePointa v podjetjih: **OCR, Image Tagging in eSignature.**

OCR – ko skeniran dokument postane dejansko uporaben

Verjetno ima skoraj vsaka organizacija v SharePointu vsaj nekaj dokumentov, ki so bili nekoč natisnjeni, podpisani in nato ponovno skenirani.

To so lahko pogodbe, računi, soglasja, stara projektna dokumentacija ali celotni arhivi, ki so bili pri digitalizaciji preprosto pretvorjeni iz papirja v PDF.

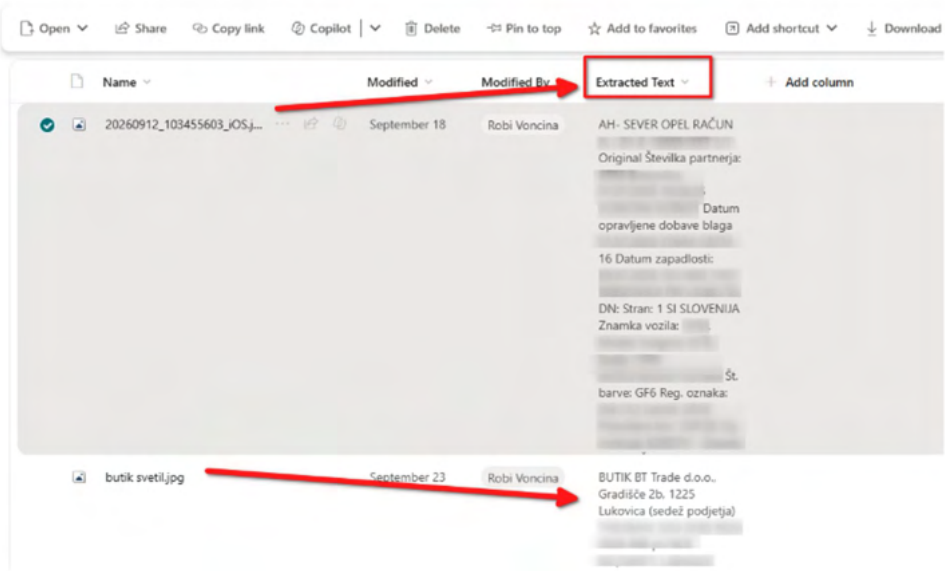
Problem pri takšnih dokumentih je, da

uporabnik dokument odpre in normalno prebere njegovo vsebino, za informacijski sistem pa je lahko celotna stran zgolj slika.

Če je na dokumentu napisano:
Pogodba št. P-2026-001 bo uporabnik to brez težav prebral. SharePoint pa tega besedila ne more nujno uporabiti pri iskanju, če PDF vsebuje samo skenirano sliko.

Tu nam pomaga **OCR oziroma Optical Character Recognition**.

Najbolj očiten primer je iskanje.
Predstavljajmo si, da imamo v SharePointu arhiv nekaj tisoč skeniranih pogodb. Uporabnik želi poiskati pogodbe, v katerih se pojavlja določena stranka ali številka pogodbe.
Brez OCR je takšno iskanje precej omejeno. Z OCR pa lahko SharePoint prepozna besedilo znotraj skeniranih strani in ga vključi v nadaljnjo obdelavo.
Zato OCR vidim kot zelo zanimivo funkcionalnost predvsem pri digitalizaciji obstoječih arhivov.



OCR prepozna natisnjeno ali ročno napisano besedilo na slikah oziroma skeniranih dokumentih in ga pretvori v besedilo, ki ga lahko sistem nadalje uporablja.

V praksi to pomeni, da lahko dokument, ki je bil prej samo slika, postane precej bolj uporaben.

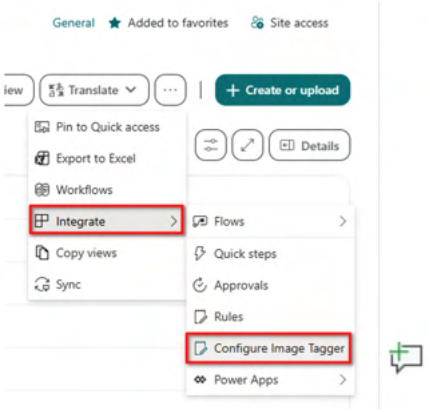
Ne gre več samo za:
papir → PDF
ampak za:
papir → PDF → prepoznano besedilo → uporabna informacija
Microsoftova storitev podpira obdelavo slik, kot so JPEG, JPG, PNG in BMP, ter dokumentov PDF, TIF in TIFF. Obračunavanje temelji na številu

obdelanih strani oziroma slikovnih elementov.

Image Tagging – nekoliko drugačen problem
OCR odgovarja na vprašanje:
Kaj piše na tej sliki?

Image Tagging pa rešuje nekoliko drugačen problem:
Kaj je na tej sliki?

To je pomembna razlika.



V dokumentnih knjižnicah se pogosto znajdejo tudi fotografije. To so lahko slike izdelkov, objekti, fotografije s terena, marketinški materiali, slike poškodb, tehnična dokumentacija ali preprosto večji arhiv slik

Tipičen problem se pojavi zelo hitro.
V knjižnici imamo datoteke:

- **IMG_3421.jpg**
- **IMG_3422.jpg**
- **IMG_3423.jpg**

Iz imen datotek seveda ne moremo ugotoviti praktično ničesar.

Lahko bi od uporabnikov zahtevali, da vsaki fotografiji ročno dodajo metapodatke, vendar iz izkušenj vemo, kako se takšne zgodbe običajno končajo. Pri desetih slikah še gre. Pri nekaj tisoč slikah pa precej težje.

Image Tagging poskuša ta problem rešiti z avtomatskim prepoznavanjem vsebine slike ter dodajanjem opisnih oznak. Zaradi teh oznak lahko slike pozneje lažje iščemo, filtriramo, razvrščamo in upravljamo.

Predstavljajmo si na primer podjetje, ki v SharePointu hrani fotografije svojih izdelkov. Namesto da bi moral uporabnik vsaki fotografiji ročno dodati informacije o tem, kaj je na njej, lahko Image Tagging samodejno analizira fotografijo in pripravi ustrezne oznake.

Podoben scenarij si lahko predstavljamo pri gradbenem podjetju, ki v SharePoint shranjuje fotografije z gradbišč, ali pri servisni službi, ki dokumentira stanje opreme in različne poškodbe.

To se mi zdi zanimivo predvsem zato, ker naenkrat koncept metapodatkov ne velja več samo za Office dokumente in PDF-je.

Tudi **slika lahko postane dokument z uporabnimi metapodatki**.

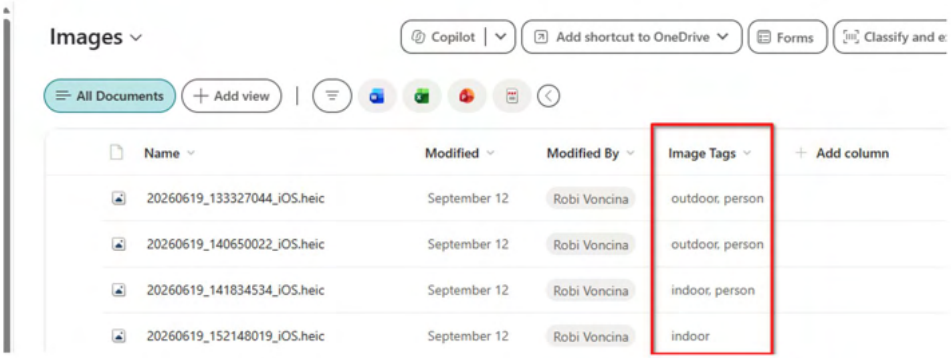
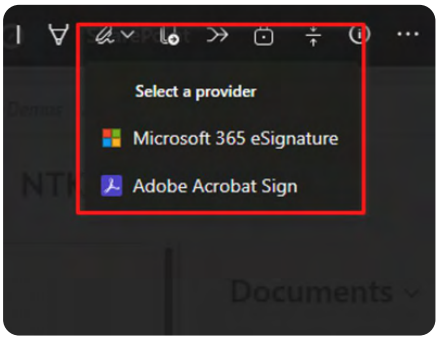


Image Tagging je še vedno Pay-As-You-Go storitev. Azure naročnino povežemo z Microsoft 365, administrator pa lahko določi, na katerih SharePoint mestih bo funkcionalnost na voljo. Strošek nastane, ko Image Tagging dejansko omogočimo na knjižnici in se slike začnejo obdelovati.

eSignature – ko mora dokument na koncu nekdo še podpisati

Za zaključek serije se mi zdi zanimiva še ena funkcionalnost, ki sama po sebi nima toliko opraviti z umetno inteligenco, ima pa veliko opraviti z življenjskim ciklom dokumenta.



eSignature.

Če pogledamo tipičen proces dela s pogodbo, lahko SharePoint uporabimo za pripravo dokumenta, shranjevanje, metapodatke, verzioniranje, odobritve in avtomatizacijo.

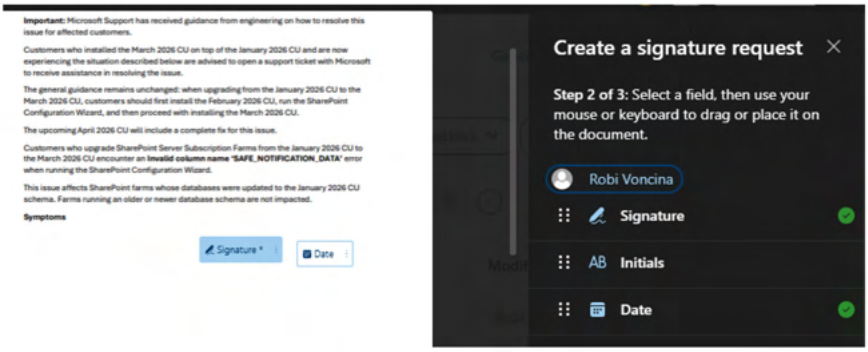
Na neki točki pa pogodbo običajno mora nekdo tudi podpisati.

Dokument zapusti proces, ki smo ga skrbno zgradili v Microsoft 365. Nekdo ga prenese iz SharePointa, pošlje po elektronski pošti, prejme podpisano različico in jo nato ponovno nalaga v SharePoint.

Z eSignature lahko ta del procesa vključimo neposredno v Microsoft 365.

Microsoft eSignature omogoča ustvarjanje elektronske zahteve za podpis dokumenta, pri čemer dokument med pregledovanjem in podpisovanjem ostaja v okolju Microsoft 365.

Tipičen scenarij bi bil:
priprava pogodbe → odobritev → zahteva za podpis → podpis →



shranjen podpisan dokument

Takšen način je uporaben pri pogodbah, soglasjih, kadrovskih dokumentih, naročilnicah in drugih dokumentih, pri katerih je podpis del poslovnega procesa.

Tudi eSignature se lahko uporablja kot Pay-As-You-Go storitev. Trenutna Microsoftova cena je 2 USD na zahtevo za podpis, ena zahteva pa lahko vključuje do deset prejemnikov.

Kako je s stroški?
Pri teh funkcionalnostih je pomembno razumeti, da Microsoft ne uporablja enega samega licenčnega modela za vse.

Pri OCR, Image Tagging in eSignature govorimo o Document Processing storitvah, ki se lahko obračunavajo glede na dejansko uporabo.

Trenutno Microsoft navaja naslednje cene:

- **OCR:** 0,001 USD na transakcijo,
- **Image Tagging:** 0,001 USD na obdelano sliko,
- **eSignature:** 2 USD na zahtevo za podpis.

Pri nekaj dokumentih to praktično ni pomemben strošek. Če pa govorimo o arhivu z milijonom skeniranih strani, potem je seveda pametno narediti izračun, preden funkcionalnost vključimo za celotno okolje.

Microsoft trenutno ponuja tudi določeno brezplačno mesečno kapaciteto za organizacije, ki imajo vzpostavljen Pay-As-You-Go. Do decembra 2026 je med drugim vključenih 2.500 OCR strani, 2.500 slik za Image Tagging in pet eSignature zahtev mesečno na tenant.

To je lahko dovolj, da funkcionalnosti najprej preizkusimo na manjšem praktičnem primeru.

Zaključek
S tem člankom bom zaključil serijo o SharePoint Online Premium in Document Processing funkcionalnostih.

Dolga leta smo se pri načrtovanju dokumentnih rešitev veliko ukvarjali s knjižnicami, mapami, Content Types, metapodatki, pogledi in dovoljenji.
Vse to je danes še kako pomembno, vendar pa v ospredje prihajajo tehnologije, ki bolje izkoriščajo dobro organiziranost in strukturiranost podatkov.

Autofill Columns lahko iz dokumenta pridobi metapodatke. OCR lahko prebere besedilo iz skeniranega dokumenta. Image Tagging lahko pomaga razumeti in organizirati zbirke slik. eSignature pa dokument vključi še v proces elektronskega podpisovanja. Microsoft vse te zmogljivosti danes umešča v širši nabor Document Processing storitev za Microsoft 365.

V naslednjih številkah Pike, pa se bom začel bolj podrobneje spuščati v funcionalnosti »Copilot-a«, ki se v zadnjem času res tako hitro spreminja, da je težko slediti. Več pa v prihodnji številki.



EXCEL – SEZNAMI IN MATRIKE

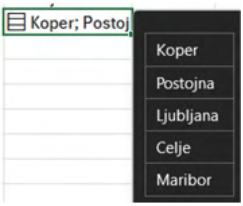
Klemen Vončina
Microsoft Office Specialist Master, MCT
klemen.voncina@kompas-xnet.si

V Insider različici Microsoft 365 aplikacij, torej v beta verziji, so s septembrsko posodobitvijo v Excel prišli seznami ter z njimi povezane funkcionalnosti. Kaj to pomeni za prihodnjo uporabo Excela?

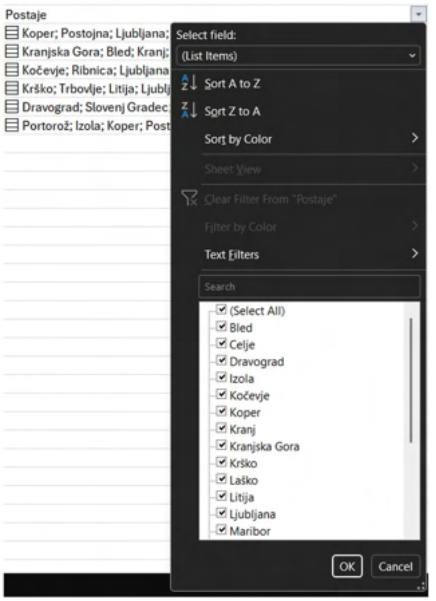
Kaj prinašajo seznami?
Od te posodobitve dalje, v Excelu ne obstaja več omejitev ene vrednosti na celico. Excelove celice lahko sedaj vsebujejo objekte, ki nosijo veliko število vrednosti. Poglejmo primer, kako se to odraža v praksi. Najprej, da naredimo seznam, na traku Insert kliknemo na ukaz List, ali uporabimo bližnjico Ctrl + J. V celici se bo pojavila majhna ikona seznama. Vsebino seznama vpišemo tako, da vsako novo vrednost seznama od prejšnje ločimo s podpičjem. Seznam potrdimo s tipko Enter.



Če nato kliknemo na ikono seznama, bomo dobili izpis vseh vrednosti seznama.



Seznami se obnašajo drugače kot klasične Excelove vrednosti. Če bomo na primer filtrirali stolpec, ki vsebuje sezname, naši kriteriji filtriranja ne bodo celotne vrednosti vsake posamezne celice, pač pa posamezne vrednosti vseh seznamov, ki so v celicah.

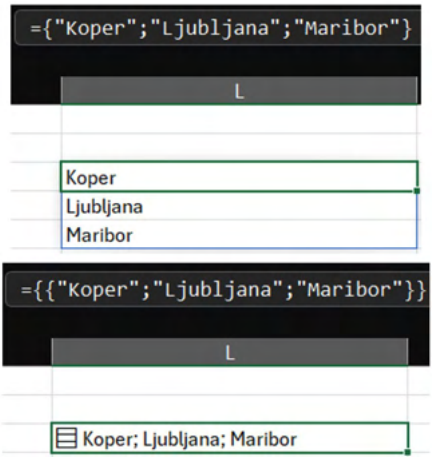


Seznami in matrike

Bolj izkušeni uporabniki Excela ste zagotovo že kdaj uporabljali matrične formule. Ali pa vsaj novejše Excelove funkcije, ki rezultat vračajo v dinamičnih matrikah.

Seznami so prilagojeni za delo z matrikami. V osnovi so Excelovi seznami gnezdene matrike.

Če vzamemo Excelovo matriko in jo ugnezdimo v še eno matriko, dobimo seznam:



Za poglobljene uvide v naše podatke, lahko na seznamih uporabljamo tudi funkcije:

Za potrebe dela s seznami, so v Excel dodali še štiri nove funkcije: HAS,



HASANY, HASALL in FLATTEN. S funkcijo FLATTEN gremo lahko v ravno obratno smer. Z njo odstranimo več nivojev gnezdenja – privzeto enega, torej seznam pretvorimo v dinamično matriko. Vzamemo lahko celotno območje, ki vsebuje sezname in ga pretvorimo v dinamično matriko, ki vsebuje vse zapise s tistega območja.

Zaključek

To je bil kratek uvod v nove Excelove sezname. Funkcionalnost je kot rečeno šele v beta fazi, tako da bomo pred splošno implementacijo v Excel zagotovo videli še kakšne spremembe in izboljšave, ki jih bomo z zanimanjem spremljali.







Z rezultati te funkcije lahko seveda operiramo naprej, kot smo sicer vajeni v Excelu. Dobimo lahko denimo sortiran seznam unikatnih vrednosti.



EN HTTPCLIENT, PROSIM: ZAKAJ GA V .NET NE USTVARJAMO ZA VSAK HTTP KLIC

Lenart Svetek
Frontend programer
lenart.svetek@kompas-xnet.si

Recimo, da želimo iz naše C# aplikacije poklicati zunanji API. Na prvi pogled se rešitev zdi precej preprosta:

```
public async Task<string>
GetDataAsync()
{
using var client = new HttpClient();
return await client.GetStringAsync(
"https://api.example.com/data");
}
```

Na tej kodi ni videti nič posebej sumljivega. HttpClient implementira IDisposable, zato z uporabo ključne besede using poskrbimo, da se objekt po zaključku operacije pravilno sprosti.

A težava ni HttpClient, temveč povezave pod njim

HttpClient uporablja http povezave, ki pa so do HTTP/2 uporabljale TCP protokol, z http/3 se to spreminja, a je to izven obsega tega članka.

Obratno od UDP, je TCP povezovalni protokol in drži povezavo med odjemalcem in strežnikom.

In ko sprostimo HttpClient, ni nujno da se bo TCP povezava sprostila, hkrati in lahko vzame dlje časa, da sprosti povezavo.

Dokler naš API, ni prevečkrat poklican, najbrž niti ne bomo opazili, ali je sploh sprožil, a takoj, ko bomo imeli malo večji val klicev API-ja, bodo pa uporabniki poročali o težavah z API-jem.

Naivna rešitev: uporabimo en HttpClient

Zdaj torej razumemo, zakaj ni najbolje, da za vsako zahtevo ustvarimo popolnoma nov HttpClient.

Rešitev se nam zdi očitna, ustvarimo en HttpClient in ga uporabimo skozi cel program.

Na prvi pogled bi bila rešitev povsem ustrezna, vendar se zaplete pri načinu, kako HttpClient obravnava DNS.

Kaj pa, če se spremeni IP naslov strežnika?

Kako HttpClient, razreši DNS naslov? Morda bi pričakovali, da HttpClient pred vsakim zahtevkom DNS strežnik vpraša za IP-naslov. A hitro ugotovimo, da to ne bi bilo najbolj smiselno. IP-naslovi se namreč ne spreminjajo tako pogosto, da bi morali DNS-poizvedbo izvajati pred vsakim HTTP-zahtevkom.

In prav po tem principu deluje tudi HttpClient. Ko vzpostavi povezavo,

prek DNS-ja pridobi IP-naslov in ga nato uporablja naprej. Tu pa se pojavi težava: HttpClient ne spremlja TTL-ja DNS-zapisa in zato ne ve, kdaj bi moral naslov ponovno razrešiti. Kaj se zgodi, če se IP-naslov strežnika medtem spremeni? Naš HttpClient lahko ostane vezan na star IP-naslov in zahtevki nenadoma začnejo odpovedovati.

To lahko rešimo, da ko ustvarimo HttpClient mu dodamo še SocektsHttpHandler z PooledConnectionLifeTime nastavljen z TimeSpan.FromMinutes(10).

```
var handler = new SocketsHttpHandler
{
PooledConnectionLifetime =
TimeSpan.FromMinutes(10)
};
var client = new HttpClient(handler);
```

S tem določimo, kako dolgo se lahko povezava ponovno uporablja. Ko njena življenjska doba poteče in se trenutna zahteva zaključi, se povezava zapre. Ob naslednji potrebi po povezavi se ustvari nova, s tem pa se ponovno izvede tudi DNS poizvedba.

Pomembno je poudariti, da **10 minut ni univerzalna vrednost**. Primerno življenjsko dobo povezav je treba izbrati glede na okolje in pričakovano pogostost sprememb DNS oziroma omrežne infrastrukture.

Od ene težave do druge. Ali se bo to sploh kdaj končalo?

Na srečo nam vsega tega ni treba vedno upravljati ročno.

.NET ponuja IHttpClientFactory, ki je namenjen prav ustvarjanju in upravljanju instanc HttpClient. Namesto da bi HttpClient neposredno ustvarjali sami, to nalogo prepustimo factoryju.

Najprej ga registriramo:

```
builder.Services.AddHttpClient();
```

Nato pa lahko IHttpClientFactory injiciramo v poljubno storitev:

```
public class MyService
{
private readonly IHttpClientFactory
_factory;
public MyService(IHttpClientFactory
factory)
{
_factory = factory;
}
public async Task<string>
GetDataAsync()
{
var client = _factory.CreateClient();
return await client.GetStringAsync(
"https://api.example.com/data");
}
}
```

Če ste prej pozorno brali, zdaj verjetno nekoliko postrani gledate naslednjo vrstico:

```
var client = _factory.CreateClient();
```

Ali nismo ravno ugotovili, da ne želimo za vsako operacijo ustvarjati novih HttpClient objektov? In kje je zdaj using?

Trik je v tem, da IHttpClientFactory ne upravlja življenjske dobe povezav na

enak način kot naš prvotni primer. Factory predpomni in ponovno uporablja pripadajoče HttpMessageHandler instance. Posledično lahko ustvarjamo kratkoživeče HttpClient instance, ne da bi pri vsakem CreateClient() zavrgli tudi connection pool.

To pomeni, da lahko naši HttpClient objekti ostanejo kratkoživeči, medtem ko se dražji del infrastrukture pod njimi ponovno uporablja.

Poimenovani HttpClienti

IHttpClientFactory nam prinese še eno prednost. Konfiguracijo posameznih zunanjih storitev lahko centraliziramo.

Recimo, da naša aplikacija uporablja vremenski API:

```
builder.Services.AddHttpClient("Weath
erApi", client =>
{
client.BaseAddress = new Uri("https://
api.example.com/");
client.Timeout =
TimeSpan.FromSeconds(10);
client.DefaultRequestHeaders.Add(
"Accept",
"application/json");
});
```

Kjer potrebujemo dostop do vremenskega API-ja, nato preprosto zahtevamo ustreznega poimenovanega klienta:

```
var client =
factory.CreateClient("WeatherApi");
var result = await
client.GetStringAsync("weather");
```

Na enem mestu lahko tako določimo:

- osnovni URL,
- timeout,
- privzete HTTP glave,
- avtentikacijo,
- dodatne DelegatingHandler komponente.

Če se naslov API-ja spremeni, ga ni treba iskati po celotni kodi. Konfiguracija je na enem mestu.

Zagotovo ne more biti še bolje?

Seveda je lahko. Poimenovani klienti so uporabni, vendar imajo eno pomanjkljivost:

```
factory.CreateClient("WeatherApi");
```

"WeatherApi" je še vedno samo niz znakov. Poleg tega mora vsak razred, ki uporablja API, vedeti, kako sestaviti njegove URL-je ter kako interpretirati njegove odgovore. Tu pridejo v igro typed clienti.

Namesto da po aplikaciji uporabljamo generičen HttpClient, izdelamo razred, katerega edina naloga je komunikacija z določeno zunanjo storitvijo:

```
builder.Services.AddHttpClient<Weath
erApiClient>(client =>
{
client.BaseAddress =
new Uri("https://api.example.com/");
client.Timeout =
TimeSpan.FromSeconds(10);
})
```

Nato definiramo klienta:

```
public class WeatherApiClient
{
private readonly HttpClient _client;
public WeatherApiClient(HttpClient
```

```
client)
{
_client = client;
}
public async Task<string>
GetWeatherAsync()
{
return await
_client.GetStringAsync("weather");
}
}
```

Drugim delom aplikacije zdaj sploh ni treba vedeti, kateri URL uporablja vremenska storitev.

Namesto: `var client = factory.CreateClient("WeatherApi"); var weather = await client.GetStringAsync("weather");` lahko preprosto uporabimo: `var weather = await weatherApiClient.GetWeatherAsync();` ki kliče:

```
public async
Task<WeatherResponse?>
GetWeatherAsync()
{
return await
_client.GetFromJsonAsync<WeatherRes
ponse>("weather");
}
```

Kaj torej uporabiti?

Imate **manjšo ali preprosto aplikacijo**, ki komunicira le z eno zunanjo storitvijo? Dolgoživeča oziroma singleton instanca HttpClient je lahko povsem primerna rešitev, vendar mora biti pravilno konfigurirana, na primer z PooledConnectionLifetime, da se povezave občasno obnovijo in s tem omogočijo ponovno razreševanje DNS-ja.

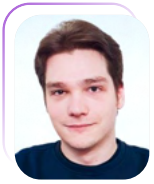
Če aplikacija uporablja **več različnih konfiguracij HttpClient-a**, vendar želite odjemalce upravljati na enem mestu, je IHttpClientFactory z **named clienti** preprosta in pregledna rešitev.

Pri **večjih aplikacijah, ki komunicirajo z več zunanjimi storitvami**, pa bi praviloma priporočil typed cliente. Vsaka zunanja storitev tako dobi svoj razred, ki vsebuje njeno konfiguracijo in logiko komunikacije.

Preostanek aplikacije pa se namesto z HttpClient ukvarja z lepo definiranim odjemalcem, na primer WeatherClient, PaymentClient ali RegistrationClient. IHttpClientFactory pri tem še vedno skrbi za upravljanje življenjske dobe osnovnih HTTP handlerjev.

Za konec

HttpClient se na prvi pogled zdi preprost kot luža. A ko se poglobimo v upravljanje povezav, DNS, življenjsko dobo handlerjev in ponovno uporabo socketov, hitro ugotovimo, da se pod površjem skriva pravo morje.



NAPADI TIPA MAN-IN-THE-MIDDLE (MITM)

Žiga Brinc
Frontend programer
ziga.brinc@kompas-xnet.si



Man-in-the-Middle (MitM) je vrsta kibernetskega napada, pri katerem se napadalec postavi med dve strani komunikacije ter prestreza in po potrebi spreminja podatke, ki se izmenjujejo med njima. Obe strani pri tem običajno menita, da komunicirata neposredno in varno, ne da bi vedeli za prisotnost napadalca.

Glavni cilj napada je pridobiti občutljive podatke, kot so prijavni podatki, osebni podatki, finančne informacije in vsebina komunikacije. Napadalec lahko prestrežene podatke uporabi za krajo identitete, prevzem uporabniških računov, finančne goljufije ali nadaljnje napade na sisteme.

Pogostejše metode napadov:

- Lažne brezžične dostopne točke (Evil Twin): Napadalec ustvari lažno brezžično omrežje (Wi-Fi), ki posnema legitimno omrežje, pogosto tudi z uporabo enakega ali podobnega imena (SSID). Če se uporabnik poveže nanj, lahko napadalec prestreza in spremlja njegov omrežni promet ter ga po potrebi tudi spreminja.
- ARP Spoofing (ARP Poisoning): Napadalec pošilja ponarejena sporočila ARP znotraj lokalnega omrežja in tako povzroči, da naprave njegov naslov MAC povežejo z naslovom IP druge naprave, na primer omrežnega prehoda (gateway). Posledično se lahko omrežni promet preusmeri prek napadalčeve naprave, kjer ga lahko prestreza ali spreminja.

- DNS Spoofing (DNS Poisoning): Napadalec ponaredi ali spremeni odgovore DNS, zaradi česar je uporabnik namesto na legitimno spletno mesto preusmerjen na napačen naslov IP, ki lahko vodi do lažnega oziroma zlonamernega spletnega mesta.

- SSL Stripping: Napadalec poskuša povezavo HTTPS prisilno spremeniti v nešifrirano povezavo HTTP. To mu omogoča prestrezanje podatkov, ki se med uporabnikom in napadalcem prenašajo brez šifriranja.

- Adversary-in-the-Middle (AiTM): Napadalec ustvari zlonamerno spletno mesto, ki deluje kot posrednik med uporabnikom in legitimno storitvijo. Tako lahko prestreza prijavne podatke in žetone za preverjanje pristnosti oziroma piškotke uporabniške seje. S tem lahko pridobi dostop do že prijavljene uporabniške seje in potencialno zaobide večfaktorsko preverjanje pristnosti (MFA).

Preprečevanje in zaščita

Za zaščito pred napadi MitM je najpomembnejša uporaba ustrezno šifrirane komunikacije in preverjanje pristnosti strežnikov. Poleg tega so priporočljivi naslednji varnostni ukrepi:

- HTTPS in TLS (Transport Layer Security): Zagotavljata šifriranje komunikacije med uporabnikom in strežnikom ter preverjanje identitete strežnika z digitalnim potrdilom. Tako napadalec brez ustreznih kriptografskih ključev praviloma ne more prebrati ali neopazno spreminjati prestreženih podatkov.

- Uporaba varnih omrežij (Wi-Fi): Izogibajte se neznanim in sumljivim brezžičnim omrežjem, pri lastnih omrežjih pa uporabljajte ustrezno šifriranje, kot sta WPA2 ali WPA3. Ta standarda ščitita komunikacijo med napravo in dostopno točko, vendar sama po sebi ne zagotavljata, da je dostopna točka legitimna.

- VPN (Virtual Private Network): Ustvari šifriran tunel med napravo in strežnikom VPN, kar pomaga zaščititi omrežni promet predvsem pri uporabi nezaupanja vrednih omrežij. Vendar VPN ne nadomesti HTTPS in ne prepreči vseh vrst napadov MitM, saj ščiti promet predvsem do strežnika VPN.

- Posodabljanje programske opreme: Redno posodabljanje operacijskih sistemov, brskalnikov in omrežnih naprav pomaga odpravljati znane varnostne ranljivosti, ki bi jih napadalci lahko izkoristili za prestrezanje ali spreminjanje komunikacije.

- HSTS (HTTP Strict Transport Security): Varnostni mehanizem, ki brskalniku določa uporabo povezav HTTPS namesto HTTP. S tem pomaga preprečevati napade SSL Stripping, saj preprečuje uporabo nešifrirane povezave in

nadaljevanje povezave, kljub napaki pri preverjanju digitalnega potrdila.

- FIDO2/WebAuthn in Passkeys: Uporaba načinov preverjanja pristnosti, odpornih proti lažnemu predstavljanju (phishing-resistant authentication), zmanjša možnost kraje prijavnih podatkov z napadi Adversary-in-the-Middle (AiTM). Te metode uporabljajo kriptografsko preverjanje pristnosti, vezano na domeno legitimnega spletnega mesta, zato jih ni mogoče preprosto uporabiti na lažnem spletnem mestu.
- Preverjanje digitalnih potrdil in spletnih naslovov: Ne prezrite varnostnih opozoril brskalnika glede neveljavnih, poteklih ali nezaupanja vrednih digitalnih potrdil. Preverite tudi, ali je spletni naslov (domena) pravilen. Veljavno digitalno potrdilo in povezava HTTPS namreč sama po sebi še ne zagotavljata, da je spletno mesto legitimno ali zaupanja vredno. Tudi lažna spletna mesta, namenjena lažnemu predstavljanju (phishingu), lahko uporabljajo veljavna digitalna potrdila.



DINAMIČNO NALAGANJE VSEBINE S POMOČJO INTERSECTION OBSERVER API

Domen Gričar
MCSD, MCSA, MCT
domen.gricar@kompas-xnet.si

Neskončno drsenje (infinite scrolling) je v sodobnih spletnih aplikacijah pogosto uporabljeno. Namesto, da bi bila vsa vsebina prikazana naenkrat ali da bi morali uporabniki klikniti gumb »Naloži več«, se nova vsebina samodejno naloži, ko se uporabnik približa dnu strani. Takšen pristop se pogosto uporablja po družbenih omrežjih, seznamih izdelkov, novičarskih spletnih straneh in galerijah slik. Pogost način implementacije te funkcionalnosti je opazovanje dogodka scroll v brskalniku. Vendar lahko nenehno preverjanje položaja drsenja povzroči nepotrebne izračune in težave z zmogljivostjo strani. Boljša alternativa je Intersection Observer API, ki omogoča zaznavanje, kdaj element pristane znotraj vidnega območja, ne da bi morali ročno spremljati vse dogodke drsenja.

Intersection Observer API omogoča asinhrono spremljanje sprememb v prekrivanju med elementom in njegovim nadrejenim vidnim območjem oziroma drugim določenim elementom. Omogoča zaznavanje, kdaj element vstopi v vidno območje zaslona ali ga zapusti. Za delovanje

API-ja uporabimo konstruktor IntersectionObserver, ki sprejme funkcijo povratnega klica (callback) in neobvezen konfiguracijski objekt.

```
const observer = new
IntersectionObserver((entries) => {
   entries.forEach((entry) => {
      if (entry.isIntersecting) {
         console.log('Element je viden.');
      }
   });
});
const target =
document.querySelector('#nalaganje');
observer.observe(target);
```

Funkcija prejme elemente (entries), ki predstavljajo spremembe pri opazovanih delih strani. Vsak element vsebuje lastnost isIntersecting, ki pove, ali se opazovani element trenutno prekriva z vidnim območjem opazovalca (observer). Za razliko od običajnih opazovanj dogodka scroll, brskalnik asinhrono upravlja zaznavanje prekrivanja, zato je takšen pristop učinkovitejši in enostavnejši za vzdrževanje.

Za implementacijo neskončnega drsenja potrebujemo element za vsebino in dodaten element, ki bo služil kot sprožilec nalaganja. Ta sprožilni element postavimo na dno vsebine. Ko postane viden, sprožimo zahtevo za nalaganje dodatnih podatkov.

```
<div id="vsebina">
   <!-- Dinamična vsebina -->
</div>
<div id="nalaganje">
Nalaganje...
</div>
```

Sprožilnega elementa ni treba posebej oblikovati, njegova glavna naloga je, da opazovalcu zagotovi referenčno točko, ki jo lahko spremlja.

Lahko pa dodamo tudi nekaj osnovnih CSS pravil, da uporabniku povemo, da vsebine še ni konec.

```
#vsebina {
   display: flex;
   gap: 1rem;
   padding: 1rem;
}
.element {
   padding: 1rem;
   color: #383838;
}
#nalaganje {
   padding: 2rem;
   text-align: center;
   color: #094879;
}
```

Preden povežemo opazovalca, potrebujemo funkcijo, ki pridobi in prikaže dodatno vsebino. V spodnjem primeru se vsebino dobi z uporabo preproste JavaScript funkcije.

```
const content =
document.querySelector('#vsebina');
let currentPage = 1;
async function loadMoreContent() {
   const response = await fetch(`/api/
posts?page=${currentPage}`);
   if (!response.ok) {
      throw new Error('Napaka pri
nalaganju vsebine.');
   }
```

```
    const posts = await response.json();
    posts.forEach((post) => {
      const article =
 document.createElement('article');
      article.className = 'element';
      article.textContent = post.title;
      content.appendChild(article);
    });
    currentPage++;
}
```

Funkcija pošlje zahtevo za naslednjo stran objav, pretvori odgovor v JSON in za vsako objavo ustvari nov article element. API spremlja sprožilni element in metoda naloži dodatno vsebino vsakič, ko ta postane viden.

```
const nalaganje =
document.querySelector('#nalaganje');
const observer = new
IntersectionObserver(
    async (entries) => {
      const entry = entries[0];
      if (entry.isIntersecting) {
        await loadMoreContent();
      }
    },
    {
      root: null,
      rootMargin: '200px',
      threshold: 0
    }
);
observer.observe(nalaganje);
```

Konfiguracijski objekt vsebuje tri lastnosti:

- root, ki določa element uporabljen kot vidno območje. Vrednost null pomeni, da se uporablja vidno območje brskalnika (viewport).
- rootMargin doda ali odstrani prostor okoli meja vidnega območja. Vrednost 200px omogoča sprožitev opazovalca še preden sprožilni element dejansko vstopi v vidno območje. Omogoča, da začnemo nalagati vsebino nekoliko prej, tako zmanjšamo možnost, da bi uporabnik dosegel dno strani in moral čakati na prikaz novih elementov.
- threshold določa, kolikšen delež opazovanega elementa se mora prekrivati z vidnim območjem, da se sproži funkcija povratnega klica. Vrednost 0 pomeni, da se lahko sproži že ob začetku prekrivanja.

Pomembno je preprečiti, da bi se več zahtev izvajalo hkrati. Če sprožilni element med nalaganjem ostane viden, se lahko funkcija opazovalca izvede ponovno. To lahko preprečimo z uporabo spremenljivke, ki spremlja stanje nalaganja.

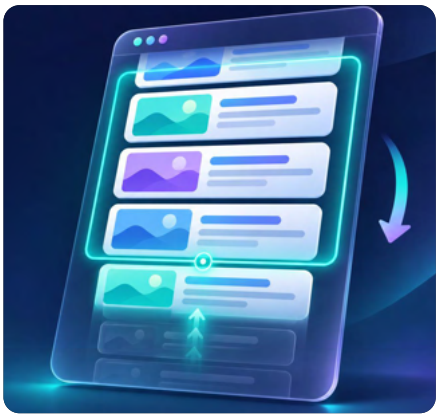
```
let isLoading = false;
let hasMoreContent = true;
async function loadMoreContent() {
    if (isLoading || !hasMoreContent)
return;
    isLoading = true;
    try {
      const response = await fetch(`/
api/posts?page=${currentPage}`);
      if (!response.ok) {
        throw new Error('Napaka pri
nalaganju vsebine');
      }
      const posts = await
response.json();
```

```
        posts.forEach((post) => {
            const article =
document.createElement('article');
            article.className = 'element';
            article.textContent = post.title;
            content.appendChild(article);
        });
        if (posts.length === 0) {
            hasMoreContent = false;
            observer.unobserve(sentinel);
            sentinel.textContent =
'Naložena je celotna vsebina.';
        } else {
            currentPage++;
        }
    } catch (error) {
        console.error(error);
        sentinel.textContent = 'Prišlo je do
napake. Prosimo poskusite ponovno.';
    } finally {
        isLoading = false;
    }
}
```

Spremenljivka isLoading zagotavlja, da se hkrati izvaja samo ena zahteva, medtem ko hasMoreContent preprečuje nepotrebne zahteve, ko je bila naložena vsa razpoložljiva vsebina. V produkcijski aplikaciji je priporočljivo dodati tudi možnost ponovnega poskusa nalaganja (retry) in bolj jasen indikator nalaganja.

Neskončno drsenje je še posebej primerno za spletne strani, kjer uporabniki brskajo po vsebini brez določenega cilja, na primer po objavah na družbenih omrežjih, novicah ali zbirkah slik. Vendar ta pristop ni vedno najboljša izbira. Spletne trgovine z velikim številom izdelkov, rezultati iskanja in vsebine, ki jih uporabniki želijo ponovno obiskati, imajo lahko več koristi od klasične postavitve s stranmi. Zaradi dostopnosti in uporabniške izkušnje je priporočljivo zagotoviti tudi nadomestni gumb »Naloži več«, uporabo ustreznega območja aria-live za obveščanje bralnikov zaslona o novo naloženi vsebini ter ohranjanje položaja drsenja med dodajanjem novih elementov.

Intersection Observer API ponuja preprost in učinkovit način implementacije neskončnega drsenja brez nenehnega spremljanja dogodkov drsenja. Z opazovanjem sprožilnega elementa lahko zaznamo, kdaj se uporabnik približuje dnu strani in dinamično naložimo dodatno vsebino. Kombinacija Intersection Observer API-ja, spremljanja stanja nalaganja in obravnave napak omogoča rešitev primerno za številne sodobne spletne aplikacije. To je odličen primer, kako lahko vgrajeni API-ji brskalnika poenostavijo spletni razvoj ter hkrati izboljšajo zmogljivost in vzdrževanje kode.





PIKIN VITAMIN - DOBRO JE VEDETI

Dijana Babić
Računovodstvo



Pomembne novosti in aktualne informacije

💶 **Zimski regres 2026: 740,94 €**
Zimski regres je zakonska pravica zaposlenega in znaša polovico minimalne plače. Ker minimalna plača za leto 2026 znaša **1.481,88 € bruto**, znaša polni zimski regres **740,94 €**.

📅 **Rok za izplačilo:** najpozneje 18 dni po izteku plačilnega obdobja za novembrsko plačo. V običajnih okoliščinah to pomeni do 18.12..

⚖️ **Zimski regres za leto 2025 je bil delno protiustaven.** Ustavno sodišče je v odločbi U-I-189/25 presodilo, da delodajalci v zasebnem sektorju (za javni sektor to ne velja) pri odmeri za leto 2025 ne bi smeli biti zavezani upoštevati obdobja zaposlitve pred **20. novembrom 2025**, ko je zakon začel veljati. Kljub temu zaposlenim že prejetega regresa **ni treba vračati**, pravica do zimskega regresa pa za prihodnja leta ostaja.

👨‍👧 **Olajšava za otroke: od 2027 prek eDavkov**
Za odmerno leto **2026 in naprej** se bo vloga za posebno olajšavo za vzdrževane družinske člane oddajala elektronsko prek **eDavkov**. Dobro je pravočasno preveriti dostop do portala ali mobilne aplikacije eDavki.

📅 **Rok za oddajo: do 15. februar 2027**
Priporoča se predhodno preverjanje rokov oddaje za posamezno leto.

👵 **Zagotovljena vdovska pokojnina: pomemben rok**
Vdove in vdovci, ki že prejemajo vdovsko pokojnino ali del vdovske pokojnine in bi lahko izpolnjevali pogoje za zagotovljeno vdovsko pokojnino, naj preverijo svojo upravičenost.

📅 **Rok za vložitev zahteve: 31. december 2026**

🏛️ Zahteva se vloži pri **ZPIZ**.

🗳️ **Referendum o ZIURS**
O uveljavitvi Zakona o interventnih ukrepih za razvoj Slovenije – ZIURS – bomo odločali na zakonodajnem referendumu.

📅 **Datum referenduma:**
6. december 2026

✉️ Za nekatere oblike glasovanja iz tujine je pomemben tudi rok **20. november 2026.**

Volivci, ki bodo takrat začasno v tujini in želijo glasovati na diplomatsko-konzularnem predstavništvu ali po pošti iz tujine, morajo svojo namero Državni volilni komisiji (DVK) sporočiti najpozneje do **20. novembra 2026**.

Odlok o razpisu zakonodajnega referenduma o Zakonu o interventnih ukrepih za razvoj Slovenije (OdZIURS) je bil objavljen 16.10.2026 v Uradnem listu št.1692/26. Vir: DVK

🛒 **Hrana se je samo oktobra podražila za 1,8 %**
Čeprav je skupna mesečna inflacija oktobra ostala nespremenjena, so se **cene hrane v enem mesecu zvišale za 1,8 %**. Na letni ravni je bila skupna inflacija 3,3 %, največ pa so k njej prispevali stanovanjski stroški in energija, ki so bili dražji za 8,6 %.
Vir: Statistični urad Republike Slovenije

ZIURS: O ČEM BOMO 6. DECEMBRA PRAVZAPRAV ODLOČALI?
6. decembra 2026 bomo na zakonodajnem referendumu odločali **o Zakonu o interventnih ukrepih za razvoj Slovenije – ZIURS**. Referendumsko vprašanje je kratko, zakon pa nikakor ni: v enem zakonu je združenih več davčnih, pokojninskih, delovnopravnih, zdravstvenih in drugih sprememb.

ZIURS je Državni zbor sprejel 11. maja 2026. Ker je bilo zbranih več kot 40.000 podpisov za referendum, zakon še ni začel veljati. Referendum bo **6. decembra 2026**, volivci pa bomo odločali o zakonu kot celoti – posameznih ukrepov ni mogoče potrditi ali zavrniti ločeno.

1. Normirani s.p.: višji pragovi in ugodnejši normirani odhodki
Ena večjih sprememb se nanaša na **normirance**. Za t. i. polnega normiranca bi bil prag prihodkov za vstop oziroma obstanek v sistemu praviloma **150.000 €**, če je bil samozaposlen za polni delovni čas najmanj pet mesecev. Za tiste, ki tega pogoja ne izpolnjujejo, je pomemben prag **70.000 €**, zakon pa pozna tudi vmesni prag 110.000 € glede na položaj v dveh predhodnih letih.

Tudi priznani normirani odhodki bi bili spremenjeni. Pri polnem normirancu bi se do 100.000 € prihodkov priznalo **80 % normiranih odhodkov**, nato pa bi se odstotek postopoma zmanjševal. Pri popoldanskem s. p. bi se 80 % odhodkov priznavalo do 50.000 € prihodkov.

Kaj to pomeni v praksi?

Za del samozaposlenih bi bila obdavčitev ugodnejša, predvsem pa bi lahko v sistemu normirancev ostali tudi pri višjih prihodkih.

Pomembna posebnost: del določb naj bi se uporabljal že **za davčno leto 2026**, čeprav zakon zaradi referenduma še ni začel veljati. Prav ta retroaktivnost je bila med pravnimi vprašanji, na katera je opozarjala zakonodajno-pravna služba Državnega zbora.

2. Oddajanje stanovanja: davek s 25 % na 15 %, v nekaterih primerih celo na 5 %

ZIURS precej znižuje obdavčitev prihodkov iz oddajanja nepremičnin. Splošna stopnja dohodnine bi se znižala s sedanjih **25 % na 15 %**.
Še nižja, **5-odstotna stopnja**, bi veljala pri oddajanju stanovanja ali hiše:

- družini z najmanj enim vzdrževanim otrokom do 18. leta ali
- osebi, mlajši od 30 let, če najemnik v stanovanju dejansko živi in ima tam prijavljeno stalno prebivališče.

Država bi dolgoročno oddajanje stanovanj mladim in družinam davčno precej bolj spodbujala.

3. Socialna oziroma razvojna kapica: nad 7.500 € bruto ne bi bilo več prispevkov

ZIURS uvaja **najvišjo osnovo za plačevanje socialnih prispevkov v višini 7.500 € mesečno**.

Če bi imel zaposleni na primer **10.000 € bruto plače**, bi se prispevki za socialno varnost obračunali samo od 7.500 €, od preostalih 2.500 € pa ne več.

To bi pomenilo:

- višji neto dohodek najbolje plačanih zaposlenih in
- nižji strošek dela za njihove delodajalce,
- hkrati pa manj prihodkov v pokojninski, zdravstveni in druge socialne blagajne, ampak ne za državni proračun, kajti razlika bi se vseeno upoštevala v dohodnini.

Ukrep je namenjen predvsem povečanju konkurenčnosti pri zaposlovanju visoko plačanih strokovnjakov.

4. Delo in polna pokojnina hkrati

Danes lahko človek, ki po izpolnitvi pogojev za upokojitev, še naprej dela s polnim delovnim časom, poleg plače prejema samo **del pokojnine**.

ZIURS uvaja tako imenovani **polni dvojni status**.

Oseba, ki izpolni pogoje za starostno pokojnino in še naprej dela oziroma opravlja dejavnost, bi lahko prejemala: **100 % plače oziroma dohodka iz dejavnosti + 100 % starostne pokojnine**.

Ob tem bi še naprej ostala vključena v socialna zavarovanja. Zakon želi finančno spodbuditi ljudi, da po izpolnitvi pogojev za upokojitev ostanejo aktivni. Po drugi strani pa

pomeni tudi večje izdatke pokojninske blagajne.

5. Ko izpolniš pogoje za upokojitev, bi lahko pogodba o zaposlitvi prenehala.

S prejšnjo spremembo je povezana še ena precej pomembna novost.

Po ZIURS bi pogodba o zaposlitvi praviloma **prenehala po samem zakonu na dan, ko delavec izpolni pogoje za starostno pokojnino.**

Delavec bi lahko najpozneje 90 dni prej predlagal, da želi nadaljevati delo. Delodajalec bi moral odgovoriti v 30 dneh. Če ne bi odgovoril, bi se štelo, da se zaposlitev nadaljuje pod enakimi pogoji. Delavec bi imel tudi možnost sodnega varstva.

Izpolnitev pogojev za upokojitev bi postala pomembna prelomnica v delovnem razmerju, vendar avtomatično ne bi pomenila, da zaposleni nujno preneha delati.

6. Nižji prispevki za samozaposlene in kmete z zelo nizkimi prihodki
ZIURS vsebuje tudi posebno pomoč samozaposlenim in kmetom z nizkimi prihodki.

Če njihov letni prihodek ne bi presegel **letne minimalne plače**, bi se lahko njihova osnova za prispevke določila največ v višini **45 % minimalne letne plače**.

Ukrep bi bilo mogoče koristiti **enkrat za največ 24 mesecev**.

Namen ukrepa je predvsem zmanjšati breme obveznih prispevkov v obdobjih, ko je dohodek iz dejavnosti zelo nizek.

7. Upokojenci ne bi več plačevali prispevka za dolgotrajno oskrbo
ZIURS posega tudi v sistem dolgotrajne oskrbe.

Od **pokojnin se prispevek za dolgotrajno oskrbo ne bi več obračunaval**.

To pomeni, da upokojencu od pokojnine ne bi več odtegovali sedanjega prispevka za dolgotrajno oskrbo.

Zakon hkrati preprečuje določeno dvojno plačevanje prispevka pri ljudeh, ki imajo več različnih vrst dohodkov.

8. Nižji DDV za osnovna živila
ZIURS spreminja tudi DDV pri izbranih osnovnih živilih.

Med živili, za katera predvideva ugodnejšo davčno obravnavo, so med drugim:

- kruh in pšenična moka,
- testenine,
- goveje, svinjsko in piščančje meso,
- mleko in jogurt,
- nekateri siri in maslo,
- jajca,
- sveže sadje in zelenjava,
- sončnično olje,
- beli sladkor ter

- nekatera živila za posebne zdravstvene potrebe.

Za kupca je pomembno opozorilo: nižji DDV še ne pomeni nujno, da se bo maloprodajna cena znižala za popolnoma enak znesek. Končno ceno določi trgovec.

9. Vlada bi lahko začasno znižala DDV na energente na 9,5 %
Če bi prišlo do večjega dviga cen energije, bi zakon vladi omogočil, da za največ devet mesecev določi **9,5 % DDV** za:

- električno energijo,
- zemeljski plin,
- daljinsko ogrevanje in
- les za kurjavo.

V času uporabe ukrepa bi morala vlada pod določenimi pogoji zagotavljati tudi regulacijo cen energentov.

Gre torej za nekakšno **varovalko za energetsko draginjo**, ne avtomatično stalno znižanje DDV.

10. Kratkoročno oddajanje stanovanj – leta 2027 brez časovne omejitve
ZIURS posega tudi v nov Zakon o gostinstvu in pravila za kratkotrajno oddajanje stanovanj, kot sta Airbnb in Booking.

Za **leto 2027** se časovne omejitve kratkotrajnega oddajanja stanovanj ne bi uporabljale. Prav tako se v letu 2027 ne bi uporabljala uredba, ki določa občine z visokim tveganjem zaradi kratkotrajnih najemov.

Lastniki stanovanj bi imeli vsaj začasno več svobode pri kratkoročnem turističnem oddajanju.

11. Zdravniki iz javnih zavodov bi lahko pod pogoji delali tudi drugje
ZIURS precej posega tudi v zdravstveno zakonodajo.

Zdravstveni delavec, zaposlen v javnem zavodu, bi lahko z **vnaprejšnjim pisnim soglasjem** delodajalca opravljal zdravstvene storitve tudi pri drugem izvajalcu.

Za zaposlenega za polni delovni čas bi bilo takšno dodatno delo praviloma omejeno na **osem ur tedensko**, pri čemer bi se omejitev lahko ugotavljala kot povprečje šestmesečnega obdobja.

Soglasje bi bilo mogoče izdati največ za 12 mesecev, delodajalec pa bi ga smel dati samo ob določenih pogojih – med drugim če delavec redno izpolnjuje svoje obveznosti v matičnem javnem zavodu.

Namen ukrepa: povečati razpoložljivost zdravstvenih kadrov in omogočiti njihovo delo pri več izvajalcih.

12. Manj stroga odgovornost delodajalcev za nekatere poškodbe pri delu
ZIURS spreminja tudi možnost, da ZZZS od delodajalca zahteva povračilo stroškov zaradi poškodbe ali bolezni

zaposlenega.

Po novi ureditvi bi moral biti pri določenih kršitvah izkazan **namen ali huda malomarnost**, ne zgolj navadna malomarnost.

Če bi bila škoda povzročena iz hude malomarnosti, bi bilo povračilo omejeno na največ **20-kratnik najnižje pokojninske osnove**. Pri namernem ravnanju bi delodajalec še vedno odgovarjal za celotno škodo.
To je sprememba, ki je pomembna predvsem za podjetja in odgovornost delodajalcev.

In koliko vse skupaj stane?
To je eden najbolj spornih delov razprave.

Ministrstvo za finance je **2. novembra 2026** predstavilo nov izračun. Po tako imenovani dinamični metodi ocenjuje, da bi ZIURS javnofinančne prihodke zmanjšal za približno **647 milijonov € letno**.
Če bi gledali samo neposredni oziroma statični učinek brez predvidenih pozitivnih učinkov na gospodarsko dejavnost, ministrstvo ocenjuje izpad na približno **848 milijonov €**.

Samo izpad pokojninske blagajne je ocenjen na približno **369 milijonov €**, ki naj bi ga nadomestil državni proračun.

Predlagatelji zakona računajo, da bi nižji davki in prispevki povečali razpoložljivi dohodek, investicije, potrošnjo in konkurenčnost gospodarstva ter tako del izpada povrnili skozi večjo gospodarsko aktivnost.

Nasprotniki opozarjajo, da so učinki gospodarske rasti negotovi, zmanjšanje prihodkov socialnih blagajn pa zelo konkretno.

Kaj torej pomeni glas ZA in kaj PROTI?
Na referendumu ne bomo obkrožali posameznih ukrepov.

✅ **Če glasujete ZA**
podpirate uveljavitev **celotnega ZIURS** – torej skupaj spremembe za normirance, nižjo obdavčitev najemnin, socialno kapico, spremembe pokojnin in zaposlovanja, dolgotrajne oskrbe, DDV, zdravstva in gostinstva.

❌ **Če glasujete PROTI**
glasujete proti uveljavitvi **celotnega zakona**.

Zakon je na referendumu zavrnjen, če proti glasuje večina tistih, ki veljavno glasujejo, **in hkrati proti glasuje najmanj petina vseh volilnih upravičencev**.

Tri vprašanja pred odločitvijo
Ko presojamo tak zakon, ni dovolj vprašanje:
»Ali bom zaradi njega jaz plačeval manj?«

Smiselno je pogledati še:
1. Kdo ima od posameznega ukrepa

korist?
Nekateri ukrepi koristijo zelo širokemu krogu ljudi, drugi precej ozkim skupinam.

2. Koliko manj denarja bo zato zbranega v javne blagajne?
Davki in prispevki financirajo pokojnine, zdravstvo, socialne pravice in druge javne storitve.

3. Ali bo gospodarska rast izpad nadomestila?
Prav tu se najbolj razlikujejo argumenti zagovornikov in nasprotnikov zakona.

ZIURS je več kot samo »davčni zakon«. Je odločitev o tem, **koliko denarja želimo pustiti posamezniku in gospodarstvu ter koliko ga želimo zbrati skupaj za financiranje javnih sistemov**.

Pri ZIURS je zelo nenavadno, da **en glas odloča o toliko različnih vprašanjih hkrati**. Sprejeti zakon posega v **deset obstoječih zakonov**, Ustavno sodišče pa je julija dovolilo izvedbo referenduma o celotnem zakonu.

ZIURS posega v naslednje obstoječe zakone:
Zakon o dohodnini, Zakon o pravici do zimskega regresa ter prenovi ugotavljanja davčne osnove z upoštevanjem normiranih odhodkov, Zakon o prispevkih za socialno varnost, Zakon o pokojninskem in invalidskem zavarovanju, Zakon o delovnih razmerjih, Zakon o davku na dodano vrednost, Zakon o zdravstveni dejavnosti, Zakon o dodatnih interventnih ukrepih na področju zdravstva, Zakon o zdravstvenem varstvu in zdravstvenem zavarovanju.
Vir: **dz-rs.si**

»Če bi o vsakem ukrepu glasovali posebej, bi pri vseh obkrožili enako?«

»Če razvoj financiramo vsi, ali bi morali njegove koristi občutiti tudi vsi – predvsem aktivni srednji razred?«





KO SE SPREMINJA DELO, SE MORA RAZVIJATI TUDI ZNANJE

Petra Militarev
vodja izobraževanj
petra.militarev@kompas-xnet.si

V novo leto z novimi znanji, AI-adoption pristopom in izobraževanji, prilagojenimi potrebam ljudi in organizacij

Ko se leto počasi približuje koncu, tudi pri nas radi pogledamo nazaj. Na ljudi, ki smo jih srečali v naših učilnicah. Na vprašanja, ki ste nam jih zastavljali. In predvsem na to, kako zelo se je v samo enem letu spremenilo znanje, ki ga potrebujemo pri delu.

Na naša izobraževanja prihajate z zelo različnimi izhodišči. Nekateri želite bolje obvladati Excel, Teams, SharePoint ali druga orodja Microsoft 365. Drugi skrbite za okolja Azure, strežnike, naprave, omrežja in varnost. Tretji delate s podatki, Power BI in Microsoft Fabric. Vedno več pa vas druži vprašanje, ki danes presega posamezno delovno mesto ali strokovno področje: **kako smiselno vključiti umetno inteligenco v svoje delo?**

Tudi pogovori, ki jih imamo z vami, so drugačni kot še pred letom dni. Ne sprašujete nas več samo, kaj zmoreta Copilot ali ChatGPT. Zanima vas, kaj lahko z AI izboljšate pri svojem delu. Kako naj jo začne uporabljati celotna ekipa? Kaj potrebuje vodja in kaj zaposleni? Kako se lotiti agentov? In kako po nekaj mesecih ugotoviti, ali je nova tehnologija dejansko prinesla spremembo?

Prav ti pogovori usmerjajo tudi naše načrte za prihodnje leto.

Verjamemo, da tehnologija sama po sebi še ne spremeni načina dela. **Spremenijo ga ljudje, ki jo znajo smiselno uporabiti**.

Zato želimo v prihodnjem letu še tesneje povezovati tehnološko znanje, praktične veščine in dejansko uporabo pri delu. Naš program gradimo okoli treh področij: **AI in njene adopcije v organizacijah, specialističnih znanj za IT- in podatkovne strokovnjake ter izobraževanj, prilagojenih konkretnim potrebam organizacij**.

1. AI: od prvega navdušenja do premišljene uporabe

ChatGPT, Microsoft 365 Copilot, Copilot Chat, generativna AI, agenti. Še pred nekaj leti so bili številni od teh pojmov nekaj novega, danes pa so del vsakodnevnih pogovorov o delu.

Pri delu z organizacijami opažamo pomembno razliko med tem, da ima nekdo AI-orodje na voljo, in tem, da ga

zna **redno, varno in predvsem koristno vključevati v svoje delo**.

V isti organizaciji je lahko uporabnik, ki mu AI že vsak dan pomaga pri pripravi dokumentov, analizi podatkov ali iskanju informacij, in sodelavec, ki še vedno ni povsem prepričan, pri kateri nalogi bi mu AI sploh lahko koristila. Zato pri nas vse bolj govorimo o **AI-adoption pristopu**.

Ne začnemo vedno s tečajem

Kadar z organizacijo načrtujemo širše uvajanje AI, je naš prvi korak lahko preprosto vprašanje: **Kje ste danes in kam želite priti?**

S kratkim pregledom trenutnega stanja oziroma vprašalnikom lahko najprej ugotovimo, katera AI orodja zaposleni že uporabljajo, kako samozavestni so pri njihovi uporabi, kje vidijo največ priložnosti in kaj jih pri uporabi ovira.

Na podlagi ugotovitev pripravimo **izobraževanja za posamezne skupine in vloge**. Vodstvo potrebuje drugačne kompetence in primere uporabe kot prodaja, kadrovska služba, marketing, uporabniki Microsoft 365, IT-administratorji ali podatkovni analitiki.

Z izvedenim izobraževanjem pa se pot ni nujno končala.

Po določenem obdobju uporabe lahko s ponovnim vprašalnikom preverimo, kaj se je spremenilo, katera znanja so uporabniki prenesli v svoje delo, kje še potrebujejo podporo in katere teme bi bilo smiselno nadgraditi. Za naročnika oziroma vodstvo lahko ugotovitve strnemo v **poročilo o izvedenem programu, odzivu uporabnikov in priporočilih za nadaljnje korake**.

Naša pot je tako lahko zelo preprosta:



Od ChatGPT-ja in Copilota do agentov

Do širše uporabe AI seveda ne moremo brez konkretnega znanja. Zato tudi v prihodnjem letu pripravljamo praktična izobraževanja za različne ravni predznanja, od ChatGPT-ja za poslovno uporabo in naprednejšega dela z generativno AI do Microsoft 365 Copilota, Copilot Studia, razvoja AI-agentov in njihovega vključevanja v poslovne procese.

Pri tem našega programa ne želimo zapisati enkrat na začetku leta in ga nato pustiti nespremenjenega. Področje AI se razvija izjemno hitro, zato novostim sledimo sproti ter v program vključujemo nova izobraževanja, ko se pojavijo nove tehnologije, rešitve in učne vsebine.

Med specialističnimi programi boste tako našli **MS-4014 Introduction to**

building AI agents, **MS-4022 Extend Microsoft 365 Copilot in Copilot Studio**, **PL-7008 Create agents in Microsoft Copilot Studio** in **AB-620 Design and build integrated AI agent solutions in Copilot Studio**.

Kar je danes še novost, je lahko jutri že del našega programa. Zato bomo koledar izobraževanj sproti dopolnjevali in ga razvijali skupaj s tehnologijo ter potrebami uporabnikov in organizacij.

AI že uporabljate, pa niste prepričani, kaj je vaš naslednji korak?
Pogovorimo se. Ni nujno, da začnemo s tečajem. Najprej lahko pogledamo, kje ste danes, kaj želite z uporabo AI doseči in katera znanja potrebujejo različne skupine zaposlenih. Nato skupaj zastavimo pot, ki je smiselna za vašo organizacijo.

2. AI prihaja. Dobri IT-temelji zato niso nič manj pomembni.
Ob pogovorih o AI bi skoraj lahko dobili občutek, da so vsa druga IT-znanja nenadoma stopila v ozadje.

Mi menimo ravno nasprotno.

V ozadju sodobnega digitalnega delovnega okolja še vedno potrebujemo ljudi, ki razumejo **oblak, identitete, naprave, omrežja, strežnike, podatke, razvoj, varnost in upravljanje okolij**. Z razvojem AI nekatera od teh znanj postajajo še pomembnejša.

Zato specialistična IT-izobraževanja ostajajo eden ključnih delov našega programa.

Kaj bi sami uvrstili na učni načrt IT-strokovnjaka?
Odgovor je seveda odvisen od njegove vloge in tehnologij, s katerimi dela.

Za strokovnjake, ki skrbijo za sodobna delovna okolja in končne naprave, je zanimiva pot **MD-102 Endpoint Administrator**. Za administratorje okolij Azure ostaja pomemben **AZ-104 Microsoft Azure Administrator**, na področju DevOps pa **AZ-400 Designing and Implementing Microsoft DevOps Solutions**.

Če delate z infrastrukturo in okolji Windows Server, velja pogledati tudi **AZ-802**, medtem ko se na podatkovnem področju hitro širi svet Microsoft Fabric. Tu bi posebej izpostavili **DP-600** in **DP-700**, za podatkovne analitike pa tudi **PL-300 Power BI Data Analyst**.

Del teh programov nas čaka že ob zaključku letošnjega leta, drugi termini pa se selijo v nov koledar izobraževanj.

»Ampak saj imam dokumentacijo, YouTube in AI ...«

Imate. In to je dobro.

Danes imamo za samostojno učenje na voljo več kakovostnih virov kot kadar koli prej. Microsoft Learn ponuja dokumentacijo, učne poti in praktične vsebine za najrazličnejše tehnologije. AI pa nam lahko tehnično vprašanje pomaga razložiti v nekaj sekundah.

Toda pri zahtevnejšem strokovnem učenju opažamo, da je največja vrednost predavatelja pogosto nekje drugje.

V izkušnjah. V kontekstu. V vprašanju »Kaj pa se zgodi, če ...?« V razlagi, zakaj določeno rešitev uporabiti in kdaj je ne.

Naši predavatelji ne predajajo samo vsebine iz učnega načrta. Z udeleženci odpirajo konkretna vprašanja iz njihovih okolij, povezujejo posamezne tehnologije in pomagajo razumeti tisto, česar iz dokumentacije včasih ni preprosto razbrati.

Zato na vodena izobraževanja ne gledamo kot na konkurenco samostojnemu učenju. Najbolje deluje **kombinacija obojega**.

Razmišljate, katero tehnično znanje nadgraditi v letu 2027?
Povejte nam, s katerimi tehnologijami delate danes in kam se želite strokovno razvijati. Pomagamo vam lahko sestaviti smiselno učno pot in izbrati programe, ki se med seboj dopolnjujejo.

3. Včasih pravega tečaja preprosto ni v koledarju
To je nekaj, kar smo se skozi leta dobro naučili.

Dve organizaciji lahko uporabljata podobna orodja, njuni zaposleni pa imajo povsem različne potrebe. V eni želijo uporabniki nadgraditi znanje Excela, od osnov učinkovitejšega dela do naprednih funkcij, analize podatkov ali Power Queryja. Drugje potrebujejo bolj strukturirano delo z Microsoft Projectom, SharePointom, Plannerjem ali Loopom. V tretji organizaciji je največji izziv učinkovitejše sodelovanje v Teamsih in delo na skupnih dokumentih.

Zakaj bi jim torej ponudili isti program?
Tretji steber našega pristopa so zato **izobraževanja po meri**.

Program lahko sestavimo iz ene vsebine ali kombinacije več področij. To so lahko osnovna ali napredna znanja Excela, Microsoft Project, SharePoint, Teams, Planner, ter druga orodja Microsoft 365, pa tudi širši programi, kot je Microsoft 365 v praksi: Sodelovanje in komunikacija. Po potrebi lahko več vsebin povežemo v program, ki ga v našem rednem koledarju sploh ni.

Pri tem ne začnemo pri seznamu tečajev, ampak pri vaših uporabnikih.

Kaj že znajo? Katera orodja uporabljajo pri vsakodnevnem delu? Kje izgubljajo največ časa? Kaj bi želeli delati hitreje ali bolje? In katera znanja organizacija potrebuje za naslednji korak?

Na podlagi odgovorov izberemo vsebine in jih prilagodimo posamezni skupini. Tako lahko za eno ekipo pripravimo **napredni Excel s poudarkom na njihovem načinu dela s podatki**, za drugo praktično delavnico sodelovanja v Microsoft 365, za tretjo pa kombinacijo SharePointa,

Teamsov, Plannerja in skupnega dela z dokumenti.

Tudi oblika ni vedno enaka. Včasih je prava rešitev celodnevno izobraževanje. Drugič so učinkovitejši **krajši, dve- ali triurni vsebinski sklopi**, med katerimi imajo udeleženci čas, da novo znanje preizkusijo pri svojem delu. Na naslednje srečanje se tako lahko vrnejo že s konkretnimi vprašanji in izkušnjami.

Takšen pristop nam je blizu, saj izobraževanje ni več samo **dogodek v koledarju, ampak proces razvoja kompetenc**, ki se razvija skupaj z ljudmi in potrebami organizacije.

Imate izobraževalno potrebo, pa zanjo ne najdete pravega tečaja? Povejte nam, kaj želite doseči. Skupaj lahko sestavimo program, prilagodimo vsebino predznanju in delovnim izzivom vaših uporabnikov ter izberemo obliko izvedbe, ki najbolj ustreza vaši ekipi.

Leto se končuje. Učenje pa še ne.
Preden dokončno zapremo letošnje koledarje, imamo pred seboj še nekaj priložnosti za nova znanja.

V aktualnem **koledarju tečajev** lahko preverite prihajajoča izobraževanja s področij AI, Microsoft 365, Azure, podatkov, Power BI, varnosti, razvoja in drugih tehnologij.

Če si že nekaj časa govorite »tega se moram enkrat lotiti«, je morda prav zaključek leta tisti mali potisk, ki ga potrebujete.

Če pa v podjetju že pripravljate načrt razvoja kompetenc za leto 2027, vas vabimo, da se z nami pogovorite čim prej. **Ni treba začeti s seznamom tečajev**.

Lahko začnemo z vašimi ljudmi. **Kaj potrebujejo danes? Katera znanja bodo potrebovali jutri? Kje želite kot organizacija narediti korak naprej? Nato skupaj poiščemo pravo pot**.

Tri poti, en cilj

- **Želite narediti naslednji korak z AI?** Skupaj lahko oblikujemo AI adoption program, od pregleda trenutnega stanja in izobraževanj po vlogah do merjenja napredka in priporočil za nadaljnji razvoj.
- **Želite poglobiti specialistična IT-znanja?** Preverite aktualne termine za MD-102, AZ-104, AZ-400, programe Windows Server, DP-600, DP-700, PL-300 in druga strokovna izobraževanja.
- **Iščete nekaj, česar v koledarju ni?** Povejte nam, kaj želite izboljšati. Program lahko vsebinsko in časovno prilagodimo vašim uporabnikom, tehnologijam in načinu dela.

V Kompas Xnetu si želimo, da se v prihodnjem letu ne bi pogovarjali samo o tem, **koliko ljudi smo izobrazili**, temveč predvsem o tem, **kaj lahko z novim znanjem naredijo**.

Ker cilj ni kljukica ob opravljenem tečaju. **Cilj je znanje, ki ga lahko naslednji dan uporabite pri svojem delu.**

Tečaji jesen-zima

Več odličnih tečajev vas čaka na tej povezavi

Microsoft Copilot in umetna inteligenca

19.11. - 20.11.2026	ChatGPT za poslovno uporabo	Prijava ›
10.12. - 11.12.2026	Napredni ChatGPT – Work, agenti in AI delovni procesi	Prijava ›

Microsoft Azure, infrastruktura in administracija

23.11. - 27.11.2026	MS102 - Microsoft 365 Administrator	Prijava ›
30.11. - 2.12.2026	AZ700 - Designing and Implementing Microsoft Azure Networking Solutions	Prijava ›
7.12. - 10.12.2026	AZ305 - Designing Microsoft Azure Infrastructure Solutions	Prijava ›
11.1. - 14.1.2027	AZ-400 - Designing and Implementing Microsoft DevOps solutions	Prijava ›

Power BI, SQL Server in podatkovna analitika

1.12. - 4.12.2026	DP-600 - Implement analytics solutions using Microsoft Fabric	Prijava ›
7.12. - 10.12.2026	DP-700 - Implement data engineering solutions using Microsoft Fabric	Prijava ›
14.12. - 16.12.2026	PL-300 - Design and manage analytics solutions using Power BI	Prijava ›

Office in sodobno digitalno delo

24.11. - 25.11.2026	Uvod v Excel BI	Prijava ›
27.11.2026	Vrtilne tabele	Prijava ›
4.12.2026	Microsoft Outlook	Prijava ›
14.12.2026	Outlook Copilot: Učinkovito upravljanje e-pošte in koledarja s pomočjo Copilota	Prijava ›
17.12.2026	Analize podatkov v Excelu	Prijava ›
18.12.2026	Informacijska varnost - ozaveščanje zaposlenih	Prijava ›
21.12.2026	Office Crash Course	Prijava ›
22.12.2026	Power Excel	Prijava ›

Projektno vodenje in ITSM

27.11.2026	Uporaba AI v projektnem vodenju	Prijava ›
7.12. - 8.12.2026	Microsoft Project	Prijava ›

Osvoji nova znanja in postani specialist!

Pearson CERTIPORT psi



Copilot
Tips & Tricks

Xnet
vaš zanesljiv IT partner



Copilot Excel: Predlagaj stolpec s formulo

Copilot v Excelu na podlagi obstoječih podatkov predlaga nov stolpec s primerno formulo ter ga samodejno izračuna za celotno tabelo.

Poglej



Copilot Word: Priprava govornih točk

Copilot na podlagi vsebine Word dokumenta samodejno pripravi jasne in strukturirane govorne točke, ki olajšajo predstavitev in poudarijo ključne informacije.

Poglej



Copilot PowerPoint: Prilagajanje vsebine

Uporabi Microsoft Copilot za preureditev vsebine glede na podana navodila, pri tem pa samodejno prilagodi strukturo, poudarke in tok besedila.

Poglej

ŠE VEČ COPILOT TRIKOV IN NASVETOV VAS ČAKA NA NAŠEM KANALU **TUKAJ**

kompas-xnet.si