

AKTUALNOSTI S PODROČJA IT STORITEV

XPIKA

SEPTEMBER 2023

OSNOVNA AVTOMATIZACIJA Z MICROSOFT POWER AUTOMATE: UČINKOVITO OBVEŠČANJE O OPRAVILIH

stran 8

MICROSOFT STREAM

stran 14



.NET MAUI DELAVNICA 3. DEL – PRIPRAVA RAZVOJNIH EMULATORJEV ALI NAPRAV (ANDROID, IOS)

stran 22

A MICROSOFT APPROACH TO ZERO-TRUST

stran 31



BOOTCAMP



{21.–27. oktober, Ribčev laz, Bohinj}

Windows Server

Prijava →



Organizator



Spoštovane in spoštovani,
Namesto sproščujočega in
počitniško naravnane

avgusta, nam je narava, kot v
posmeh, pokazala svojo
neznansko moč. V neverjetno
kratkem času, so prijetni potočki
postali zastrašujoča pošast, ki ruši
pred seboj in, kot za šalo, odnaša

vse, kar doseže. Odnaša in
premetava, kot da to ne bi bili
domovi nesrečnih ljudi, mostovi,
avtomobili, prikolice in tovarne, ...
pač pa papirnate ladijce, ki jih
razigrani otroci spuščajo v potok
za hišo.

Pošast je odnesla vse: neznanska
odrekanja ljudi, da so si zgradili
domove, leta trdega dela, otrokom
najljubše igranje, celo spomine.
Pustila pa obup, strah in
negotovost, bolečino, ...

Z željo, da hitro priskočimo na
pomoč ter ljudem vlijemo vsaj
malo upanja in omilimo bolečino,
smo organizirali dobrodelno
konferenco. V neverjetno kratkem
času so se odzvali predavatelji iz
vsega sveta in podprli naša
prizadevanja, da zberemo čim več
denarja za ljudi v stiski.

Novica o strašni katastrofi v



BRANKA SLINKAR

Direktorica

Sloveniji in poziv k donaciji, se je
po družbenih omrežjih širila kot
požar. Odziv je bil visoko nad
pričakovanji. Iz srca hvala
vsakomur za podporo in za vsak €
posebej – v imenu tistih, ki to
pomoč še kako potrebujejo.

Res lepo je videti, da je med nami
še nadvse veliko srčnosti,
solidarnosti in v nesreči nikogar ne
pustimo samega. Spoštovanje,
razumevanje in medsebojna
pomoč, to navdihuje in daje
neverjetno moč.

Iskrena hvala vsem, ki nam
zaupate, sodelujete z nami in nas
podpirate v prizadevanjih za boljši
jutri. Vaše zaupanje nas še
dodatno zavezuje, da delamo po
najboljših močeh in vam nudimo
samo najboljše, kar zmoremo in
znamo. Zato se naši strokovnjaki
nenehno izpopolnjujejo in sledijo

hitremu napredku v tehnologiji.

Ravno znanje pa je tisto, kar mora postati največja vrednota. Visoka stopnja usposobljenosti nam omogoča nenehne izboljšave, hitrejša in lažje delo, večjo učinkovitost vse, kar naše gospodarstvo potrebuje, da bi bilo bolj konkurenčno in omogočalo višjo dodano vrednost ter boljše življenje za nas vse.

Ko se boste odločili, da nadgradite svoje IT znanje, zaupajte našim strokovnjakom. Dovolite našim svetovalkam, da vam pomagajo izbrati najbolj primerne programe. Naši izkušeni predavatelji pa komaj čakajo, da delijo svoje znanje in praktične izkušnje s projektov. Prav dobre prakse in »male skrivnosti«, ki vam jih razkrijejo na tečajih, so velika dodana vrednost in dobra popotnica za optimalno izvedbo vašega projekta.

Zelo velike rezerve so pri končnih uporabnikih, ki praviloma poznajo osnovne funkcionalnosti MS Office orodij. Vsaka nova generacija letih pa prinaša nove ali izboljšane

funkcije ter omogoča vedno več povezovanja med njimi. V večini primerov uporabniki vsem tem novostim ne sledijo, kar pomeni še veliko ročnega dela, ki je zamudno in dopušča več napak.

Če želimo že plačane licence mnogo bolje izkoristiti in zaposlenim omogočiti učinkovito in bolj avtomatizirano delo, moramo nekaj investirati v znanje.

Naši predavatelji bodo delavnice prilagodili vam po meri (po vsebini, kot tudi časovno) in za praktične vaje uporabili primere iz vašega okolja. Tako boste na koncu usposabljanja imeli že takoj uporabne rešitve in tudi snov bo udeležencem na ta način bistveno bolj »domača«. Pokličite nas za prilagojene delavnice s tematikami: Excel, Power BI, Teams, Project, Access, O365 idr... Vsekakor si lahko poljubno izberete lokacijo izvedbe.

Preverite našo bogato ponudbo na spletni strani www.kompas-xnet.si ali še bolje, naročite se na novice info@kompas-xnet.si, da boste o vsem na tekočem.

Povabite nas k sodelovanju, ko boste potrebovali strokovnjaka za:

- Infrastrukturne storitve in rešitve (migracije, nadgradnje, ...)
- Azure / Office 365 storitve
- Razvoj spletnih aplikacij po naročilu
- Spletne strani, internet / intranet portali
- SharePoint storitve in rešitve
- Katalog SharePoint gradnikov >
- Šolanje uporabnikov in IT osebja

... naj bo Xnet vaša prva izbira, ko gre za IT rešitve in storitve. Microsoft tehnologije so naša strast.

Čuvajte se in ostanite zdravi!

Branka Slinkar

branka.slinkar@kompas-xnet.si



KATALOG Storitev

kompas-xnet.si



MICROSOFT POWER AUTOMATE

8

Osnovna Avtomatizacija z Microsoft Power Automate: Učinkovito obveščanje o opravih

Matic Vukovič

Microsoft Office Specialist Master, MCT

POWER BI

10

Power BI Window funkcije – 3. del

Klemen Vončina

MCT, Microsoft Office Specialist Master

SHAREPOINT

14

Microsoft Stream

Robi Vončina

MVP, MCT, MCITP, MCSA, MCTS

SQL

18

Optimizing Analytical Queries Part 11: Clustered Columnstore Indices

Dejan Sarka

MVP, MCT

RAZVOJ

22

.NET MAUI delavnica 3. del – Priprava razvojnih emulatorjev ali naprav (Android, iOS)

Gašper Rupnik

MCT, MS, MCSD, MCPS

RAZVOJ

25

Redux in SPFx

Domen Gričar

MCSD, MCSA, MCT

RAZVOJ

27

Postgres.js – kako se povezati z PostgreSQL bazo

Lenart Svetek

ADMINISTRACIJA

29

Powershell kotiček

Aleš Lipušček

MCP, MCTS, MCITP

ADMINISTRACIJA

31

A Microsoft Approach to Zero-Trust

Saša Kranjac
MCT, MVP

UX/UI DESIGN

35

Spletno oblikovanje za različne mobilne naprave

Urška Premzl

Xnet ekipa

Juhu, punčka bo! Iskrene čestitke

In je rekla DA. Čestitamo

3 tedne je res kratek čas 😊

Nečaki so zakon!

Moja prva knjiga – Bravo!

Veliko jih že stoji.

Z nadebudnima pomočnicama 😊

Išče in išče, dokler ne najde

Več majhnih tudi šteje

V Prago po nova znanja

Spodbuja, usmerja, uči, ... Hvala

Hitro in učinkovito. Bravo

Jaz grem kr! nazaj v Baku..

Streha že prihaja

Za dobro stvar, kadarkoli

Še bodo predavanja: Barcelona, Portorož

Anja & Gašper

Urška

Robi

Petra, Miha

Klemen

Domen

Aida, Dijana

Manca

Matic

Petra

Dejan

Lenart

Jože

Miha

Luka

Robi

KOLOFON

ISSN: 1408-7863

Kompas Xnet d.o.o.

Stegne 7

1000 Ljubljana

Telefon: 01 5136 990

Fax: 01 5136 999

Email: info@kompas-xnet.si

Web: <https://www.kompas-xnet.si>

Urednica in oblikovalka

Urška Premzl

Člani uredništva

Aleš Lipušček, Aida Kalender Avdić, Gašper

Rupnik, Miha Pihler, Jože Markič, Klemen

Vončina, Matic Vukovič, Robert Vončina, Anja

Rupnik, Petra Militarev, Dejan Sarka, Manca

Gruden.

SLEDITE NAM NA:



facebook.com/KompasXnet/



twitter.com/kompasxnet



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



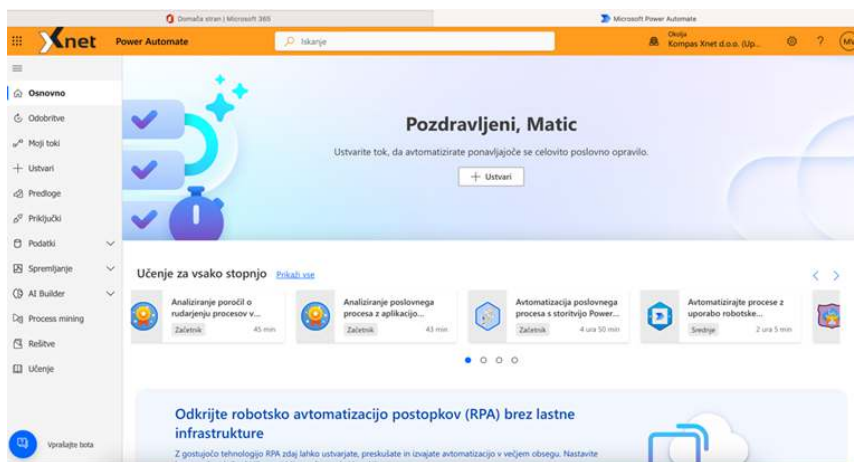
youtube.com/channel/UCfoMnj355AbRMMeG66ePS8BQ



Matic Vukovič
predavatelj, MOS
matic.vukovic@kompas-xnet.si



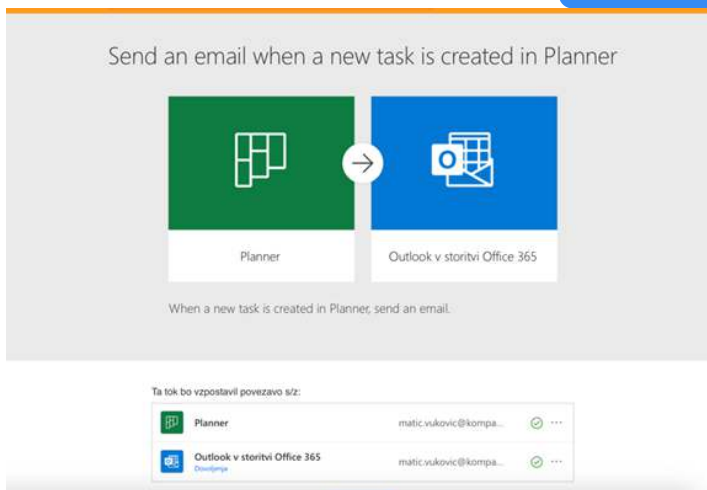
OSNOVNA AVTOMATIZACIJA Z MICROSOFT POWER AUTOMATE: UČINKOVITO OBVEŠČANJE O OPRAVILIH



V digitalni dobi, kjer hitrost in učinkovitost igrata ključno vlogo, so orodja za avtomatizacijo postala nepogrešljiv del vsakodnevnega poslovanja. Eden izmed vodilnih igralcev v tem področju je Microsoft Power Automate, platforma, ki omogoča ustvarjanje, upravljanje in izvajanje avtomatiziranih delovnih tokov brez zahtevnega programiranja. Ena izmed ključnih uporab teh delovnih tokov je obveščanje in sledenje opraviлом, kar pomaga organizacijam ohraniti pregled nad nalogami ter izboljšuje skupinsko

sodelovanje.

Power Automate omogoča enostavno izdelovanje delovnih tokov s pomočjo vizualnega vleci-in-spusti pristopa. Sestavite lahko zapletene delovne tokove, ki povezujejo različne aplikacije in storitve ter avtomatizirajo celoten proces obveščanja glede opravil. Na primer, si zamislimo, da želimo prejeti obvestilo, ko je nova naloga dodana v določenem orodju za upravljanje opravil. Z uporabo Power Automate lahko enostavno ustvarimo delovni tok, ki bo sledil tem korakom:



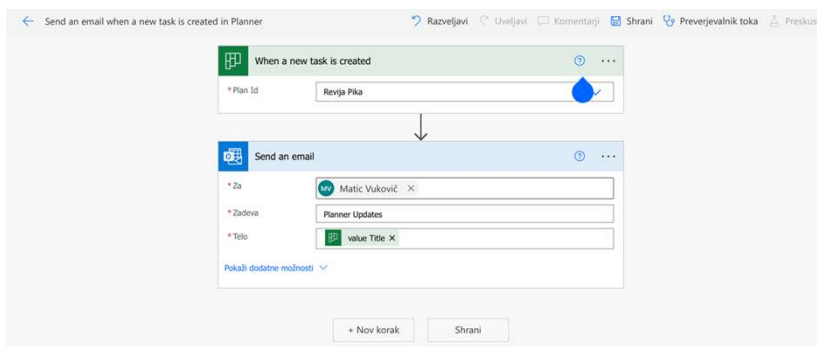
- Sprožilec: Nalogo dodamo v orodje za upravljanje opravil (npr. Microsoft Planner, Trello itd.).
- Akcija: Power Automate bo zaznal novo nalogo in sprožil delovni tok.
- Obveščanje: Na podlagi sprožilca se lahko samodejno pošlje e-poštno obvestilo.

Poleg e-poštnih obvestil Power Automate omogoča tudi obveščanje prek drugih kanalov, kot so sporočila v Microsoft Teams, obvestila v Slacku ali celo ustvarjanje opravil v drugih orodjih. To pripomore k temu, da so vsi vpleteni vedno na tekočem in da

ni prostora za zamujanje ali nesporazume.

Poleg tega Power Automate ponuja možnosti za dodajanje pogojev, časovnih zamikov in večjih logičnih vej, kar omogoča prilagajanje delovnih tokov specifičnim potrebam organizacije. S tem se zagotovi, da se obvestila o opravilih pošiljajo le v primerih, ko so resnično potrebna, kar zmanjšuje hrup in preobremenjenost s sporočili.

Vabim vas k branju članka v naslednji številki Pike, v katerem si bomo ogledali izdelavo toka po meri.



**Klemen Vončina**

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



POWER BI WINDOW FUNKCIJE – 3. DEL

V drugem delu te serije člankov smo si ogledali funkcijo INDEX, s pomočjo katere v obliki tabele dobimo podмноžico podatkov, ki glede na vrednost v nekem stolpcu spadajo na n-to mesto v tabeli, ki smo jo določili kot osnovo. Tokrat si bomo ogledali ta trenutek najkompleksnejšo, a tudi najzmožljivejšo Window funkcijo, ki se ji imenuje WINDOW. V primerih bomo zopet uporabili AdventureWorksDW2019.

Funkcija WINDOW

Pri analizah podatkov pogosto želimo zajeti podatke iz nekega »okna«. Najbolj tipičen primer bi bilo časovno okno, na primer vsota prodaj od določenega datuma 30 dni naprej in 30 dni nazaj. Ali pa kumulativne prodaje znotraj enega leta. Kot izhaja že iz imena funkcije, lahko to »okno« zgradimo s pomočjo funkcije WINDOW.

Sama po sebi nam WINDOW funkcija vrne tabelo s podмноžico podatkov iz prvotne tabele. Sama po sebi nam ta tabela verjetno ne bo zanimiva, pač pa jo uporabljamo za spreminjanje konteksta filtra v kompleksnejših izračunih. Poglejmo prvi primer, ki

smo ga prej omenili, le da bomo zaradi poenostavitve primera gledali vsoto prodaj od določenega datuma za 1 dan naprej in 1 dan nazaj. Za osnovo bomo potrebovali mero _TotalSales, ki nam bo seštelala vse prodaje:

```
_TotalSales =  
SUMX (  
FactInternetSales,  
FactInternetSales[OrderQuantity] *  
FactInternetSales[UnitPrice]  
)
```

S pomočjo funkcije WINDOW lahko potem definiramo časovno okno prodaj, ki jih želimo zajeti skupaj:

```
_TotalSales3DayWindow =  
CALCULATE (  
[_TotalSales],  
WINDOW (  
-1,  
REL,  
1,  
REL,  
SUMMARIZE (  
ALLSELECTED ( FactInternetSales ),  
DimDate[FullDateAlternateKey]  
)
```



```
ORDERBY
DimDate[FullDateAlternateKey], ASC )
)
)
```

S pomočjo funkcije `CALCULATE` spremenimo kontekst filtra, v katerem se vrednoti `_TotalSales` mera. Kontekst je definiran preko `WINDOW` funkcije na sledeč način:

- Prvi argument, `-1`, definira kje se naše podatkovno okno začne, v našem primeru torej 1 dan nazaj.
- Drugi argument, `REL`, definira, naj se začetni položaj gleda relativno, glede na trenutni položaj v vizualizaciji.
- Tretji argument, `1`, definira koliko pozicij naprej se naše okno konča, v našem primeru 1 dan v prihodnosti.
- Četrti argument, `REL`, definira, naj se končni položaj gleda relativno, zopet glede na trenutni položaj v vizualizaciji.
- Peti argument je tabela, iz katere sestavljamo naše okno. V našem primeru smo sestavili skupaj tabeli `FactInternetSales` ter stolpec `DimDate[FullDateAlternateKey]`, ki vsebuje datume prodaj.
- Šesti argument definira vrstni red sortiranja te sestavljene tabele. Sortirali smo po `DimDate[FullDateAlternateKey]` naraščajoče (`ASC`). Ta korak je v našem primeru pomemben, saj nam `WINDOW` funkcija v nasprotnem primeru lahko vrne

napačno okno, če si datumi ne sledijo v pravem vrstnem redu.

Na sliki spodaj je prikaz delovanja te mere. Na enostavno vizualizacijo tabele sem dodal `DimDate[FullDateAlternateKey]`, ter meri `_TotalSales` in `_TotalSales3DayWindow`. Vidimo, da je rezultat prodaj pri enem datumu enak vsoti prodaj tistega dneva, prejšnjega dneva in prihodnjega dneva.

FullDateAlternateKey	_TotalSales	_TotalSales3DayWindow
23. april 2012	12.423,1703 €	46.172,868 €
24. april 2012	15.475,8471 €	34.705,4924 €
25. april 2012	6.806,475 €	33.053,7871 €
26. april 2012	10.771,465 €	27.649,8842 €
27. april 2012	10.071,9442 €	44.217,5241 €

Poglejmo še en zanimiv primer, in sicer kumulativno prodajo znotraj enega leta:

```
_YearlyRunningTotal =
CALCULATE (
[TotalSales],
WINDOW (
1,
ABS,
0,
REL,
SUMMARIZE (
ALLSELECTED ( FactInternetSales ),
DimDate[FullDateAlternateKey],
DimDate[CalendarYear]
),
ORDERBY
DimDate[FullDateAlternateKey], ASC ),
PARTITIONBY
DimDate[CalendarYear] )
)
```

)

V tem primeru zopet uporabljamo funkcijo WINDOW kot modifikator filtra znotraj CALCULATE. V primerjavi s prejšnjo funkcijo lahko opazimo sledeče razlike:

- Prvi in drugi argument sta sedaj 1 in ABS. To pomeni, naj se okno začne na prvem mestu znotraj naših podatkov, gledano absolutno, torej ne glede na trenutni položaj znotraj vizualizacije. Ker želimo gledati kumulativno prodajo znotraj enega leta, bomo seveda začeli s prvim datumom. Problem pa je, če naši datumi obsegajo več let. To pa rešujemo z argumentom na sedmem mestu.
- Tretji in četrti argument sta 0 in REL. To pomeni, naj se naše okno zaključi na tistem mestu, kjer se trenutno nahajamo tudi na vizualizaciji. Ker želimo računati kumulativno prodajo znotraj enega leta, bomo zajemali prodaje do datuma, ki je v neki vrstici prikazan na vizualizaciji.
- S petim argumentom smo zopet sestavili tabelo, le da smo tokrat zaradi potreb sedmega argumenta dodali še DimDate[CalendarYear].
- S šestim argumentom smo podatke zopet spravili v pravi vrstni red za potrebe grajenja našega okna

- S sedmim argumentom smo zgradili particije, po katerih funkcija gradi naša okna. Ker želimo kumulativno prodajo računati znotraj vsakega leta posebej, smo s funkcijo PARTITIONBY definirali, naj se vsako koledarsko leto (DimDate[CalendarYear]) obravnava kot ločena celota.

Kot vidimo na sliki spodaj, se z novim letom kumulativna vsota prodaj začne računati z ničle (oziroma je enaka vsoti prodaj na prvi dan leta).

FullDateAlternateKey	_TotalSales	_YearlyRunningTotal
26. december 2012	13.210,0435 €	5.695.714,8086 €
27. december 2012	19.556,7666 €	5.715.271,5752 €
28. december 2012	23.892,13 €	5.739.163,7052 €
29. december 2012	25.080,36 €	5.764.244,0652 €
30. december 2012	34.815,4 €	5.799.059,4652 €
31. december 2012	43.425,73 €	5.842.485,1952 €
1. januar 2013	29.456,25 €	29.456,25 €
2. januar 2013	12.775,33 €	42.231,58 €
3. januar 2013	32.708,45 €	74.940,03 €

Za zadnji primer pokažimo še računanje premikajočega se povprečja (Moving Average), v našem primeru za prodaje zadnjih 180 dni:

```
_180DayMovingAverage =
AVERAGEX (
    WINDOW (
        -179,
        REL,
        0,
        REL,
        SUMMARIZE ( ALLSELECTED (
```

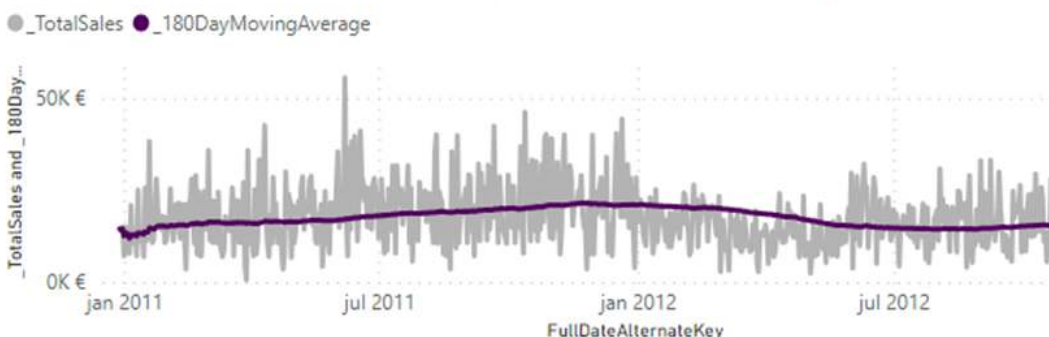
```
FactInternetSales ),
DimDate[FullDateAlternateKey] ),
ORDERBY
DimDate[FullDateAlternateKey], ASC )
),
[_TotalSales]
)
```

Tokrat smo uporabili funkcijo AVERAGEX, ki bo iterirala po tabeli, dobljeni s funkcijo WINDOW, in računala povprečje _TotalSales. Kot

vidimo, smo v tem primeru začetek in konec okna definirali relativno, glede na naš trenutni položaj v podatkih. Za nek datum gleda podatke za do 179 dni v preteklost in 0 v prihodnost, da dobimo okno 180 dni.

Rezultat takšne mere je zanimivo prikazati hkrati z rezultati _TotalSales mere, da dobimo občutek, ali prihaja v naših prodajnih obdobjih do občutnih nihanj.

_TotalSales and _180DayMovingAverage by FullDateAlternateKey



Šolanje za vaše napredovanje

AKTUALNI TEČAJI

Excel Mojster 3

Kdaj: 7.9.2023

Power BI

Kdaj: 25. - 27. 9. 2023
– izvedba potrjena

Sodobna pisarna in uporaba Microsoft 365 orodij 1. del

Kdaj: 14. - 15. 9. 2023

Microsoft Excel začetni

Kdaj: 19. - 20. 10. 2023
– izvedba potrjena

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

MICROSOFT STREAM

500 milijonov video posnetkov si ljudje vsak dan ogledamo na facebook-u, v zadnjih 30 dneh je bilo naloženih na splet več video vsebin kot je bilo v zadnjih 30ih letih TV posnetkov... Vse to so razlogi, zakaj je tudi Microsoft zelo veliko truda vložil v video portal in hrambo video vsebin za podjetja.

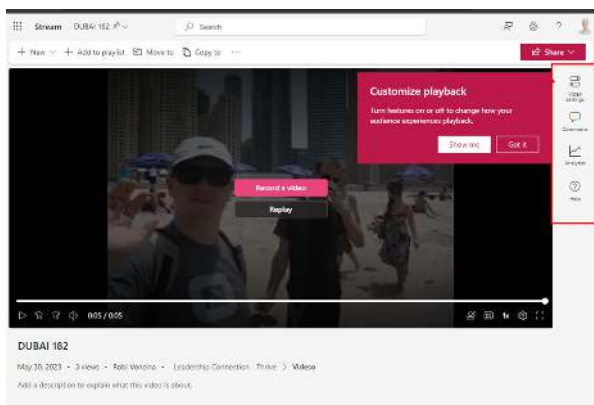
V ne tako daljni preteklosti, smo vsi veselo nalagali video vsebine na klasičen Stream portal, ki nam ga je pripravil Microsoft in nato embedali video vsebine v naše intranet portale na SharePoint Online. Klasični Stream portal je omogočal, da smo na ločeno mesto, ločen prostor shranjevali vse video posnetke. Video posnetki so bili lahko razdeljeni tudi po kanalih, posnetki sestankov in podobno. Težava klasičnega Stream portala je bila, da je bilo to preveč centralizirano in video vsebine na tak način niso bile del ostalih vsebin, ki smo jih

shranjevali na SharePoint Online.

Z mislijo boljše integracije video vsebin za uporabnike in predvsem lažje uporabe, nam je Microsoft pripravil »novo« video platformo, ki se imenuje »Stream on SharePoint«.

Stream on SharePoint

Posebnost nove verzije Stream-a je ta, da nimamo več enega centralnega mesta za nalaganje video vsebin, temveč je »Stream« katerakoli knjižnica na SharePoint, ki jo uporabnik pripravi in notri naloži video. Ko SharePoint prepozna, da gre za video datoteko, so na voljo tudi vsa orodja za upravljanje video vsebin. Ob kliku na video datoteko, se odpre nova stran, kot jo lahko vidimo na spodnji sliki. Na desnem, osrednjem delu se predvaja video, medtem ko imamo na desni strani možnost/orodja.

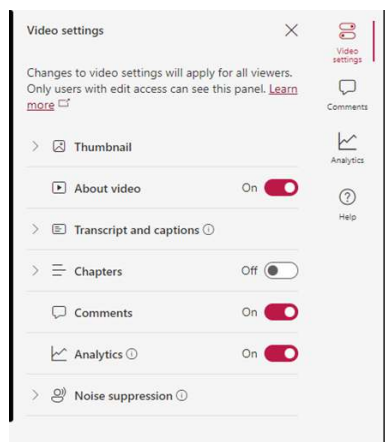


Video setting / Nastavitve videa

Med nastavitvami najdemo možnosti za nastavev predstavitevne slike videa, lahko naredimo prepis besedila, nastavimo lahko poglavja videa, s katerimi se uporabniki lahko lažje premikajo med posameznimi temami v video vsebini.

Komentarji in analitika ne potrebujejo posebne razlage, saj imamo za enkrat na voljo le vklop ali izklop možnosti.

Pri odstranitvi šumov, pa lahko le nastavi, ali je to privzeta nastavev za to datoteko.

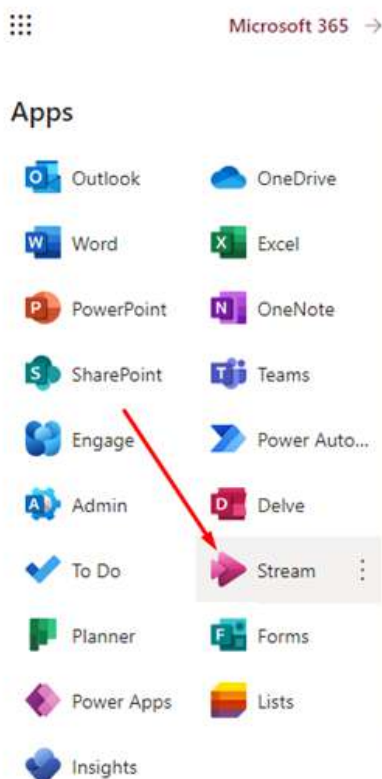


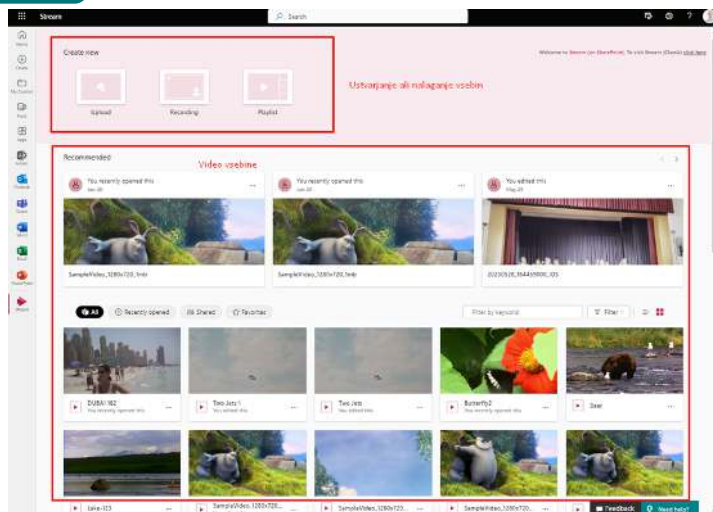
Stream on SharePoint portal

Ob kliku na aplikacijo Stream v »App launcherju« nas le ta popelje na novo spletno mesto, ki se uporablja kot agregat vseh video vsebin, ki so naložena na SharePoint.

Na spletnem naslovu:

<https://www.microsoft365.com/launch/stream> nas čaka portal, ki pa, vsaj za enkrat, druge funkcije kot za pregledovanje video vsebin niti nima. Celo, lahko bi rekli, da je malo zavajajoče, saj v primeru, da uporabnik klikne na gumb »Upload/ Naloži«, misleč da bo to naložilo video datoteko na »Stream portal« kot je bilo to včasih, v resnici pa se video datoteka naloži na uporabnikov OneDrive. S tem pride do velike zmede, saj uporabniki, ko želijo naložiti video datoteko na Stream, jo s tem želijo tudi vključiti v vsebine/ novice na SharePoint portalih, kjer se nato pojavi težava z dostopi.

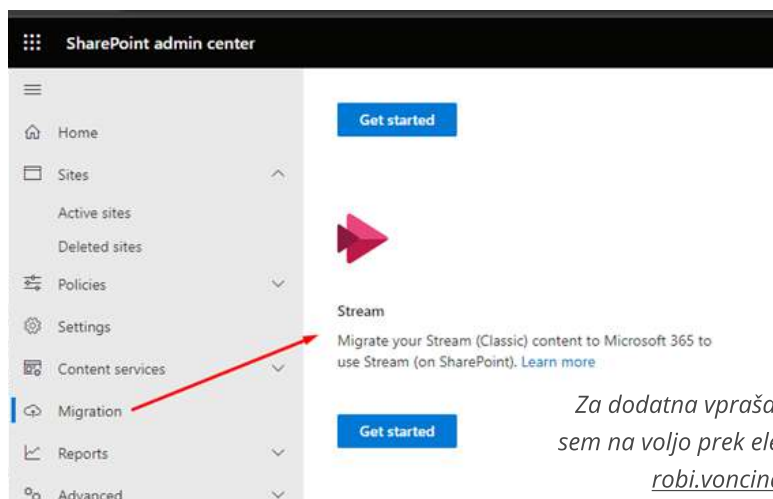




Kot pri vsakih novi storitvi, ki smo jih bili v O365 deležni že kar nekaj, bo potrebno tudi tukaj kar konkretno razmisliti in planirati, na kakšen način se bo Stream na SharePoint-u uporabljal v organizacijah. Poleg tega, pa se za organizacije, ki so do sedaj uporabljale klasičen Stream, pojavlja nova težava, saj bo potrebno zaradi ukinitve storitve, vse vsebine s klasičnega Stream—a prenesti na novi Stream.

Migracija Stream classic na Stream on SharePoint

Za namene migracije Stream classic na Stream on SharePoint nam je Microsoft pripravil tudi že orodje za migracijo, ki ga najdem v SharePoint Admin centru, v razdelku »Migration«. Več o migraciji Stream classic na Stream on SharePoint pa v naslednji številki Pike.



*Za dodatna vprašanja in komentarje sem na voljo prek elektronskih sporočil
robi.voncina@kompas-xnet.si.*

REŠITVE PO MERI

Razvijamo digitalno prihodnost

Tečaj mi je bil zelo všeč. Vse teme so bile predstavljene na razumljiv način, tako, da ni bilo težko slediti in tudi ni bilo težko narediti laboratorijskih vaj. Pohvalil bi predavatelja, ker je novo snov podajal na zelo poljuden način, tako, da smo slušatelji izvedeli še veliko več novega in pridobili na širšem znanju s področja informatike.

Tečaj M55125 SharePoint Online Power User,

Darko Brvar, DARS d.d.

»Neprecenljivo je, če delaš z ljudmi, na katere se lahko zanesesh. Hvala celotni ekipi za super sodelovanje.«

Ksenija Štrancar, SIJ d.d.

*Postavitev in vzdrževanje
SharePoint online*



**Razvoj
programske
opreme**



**SharePoint
in Bi
rešitve**



**Postavitev
ali prenova
IT infrastrukture**



**Izobraževalni
in izpitni
center**

Microsoft
Partner

Microsoft
SharePoint Online
Power User
Tečaj



Microsoft
Partner

2020 Partner of the Year Winner
Learning Award





Dejan Sarka
MVP, MCT
dsarka@solidq.com



OPTIMIZING ANALYTICAL QUERIES PART 11: CLUSTERED COLUMNSTORE INDICES

If you read my previous article about nonclustered columnstore indices, you can imagine what is the topic of this one. Of course, it is clustered columnstore indices (CCI).

Compression

Since this article continues the previous one, I am starting with dropping both indexes from the demo fact table, the NCCI and the CI, to make a heap again:

```
USE WideWorldImportersDW;
-- Drop the NCCI
DROP INDEX NCCI_FactTest
ON dbo.FactTest;
-- Drop the CI
DROP INDEX CL_FactTest_DateKey
ON dbo.FactTest;
GO
Now let's create the CCI:
CREATE CLUSTERED COLUMNSTORE
INDEX CCI_FactTest
ON dbo.FactTest;
GO
```

And, of course, the next step is to recheck the space used by the test fact table:

```
EXEC sys.sp_spaceused
```

```
N'dbo.FactTest', @updateusage =
N'TRUE';
```

```
GO
```

The result is as follows:

```
Name rows reserved data index_size
unused
```

```
-----
```

```
dbo.FactTest 2279810 23944 KB 23824
KB 0 KB 120 KB
```

The CCI even uses slightly less space than the NCCI. You can check the number of segments with the following query:

```
SELECT ROW_NUMBER() OVER (ORDER
BY s.column_id, s.segment_id) AS rn,
COL_NAME(p.object_id, s.column_id)
AS column_name,
S.segment_id, s.row_count,
s.min_data_id, s.max_data_id,
s.on_disk_size
FROM sys.column_store_segments AS
s
INNER JOIN sys.partitions AS p
ON s.hobt_id = p.hobt_id
INNER JOIN sys.indexes AS i
ON p.object_id = i.object_id
WHERE i.name = N'CCI_FactTest'
```

44. The CCI does not show a dictionary

```
ORDER BY s.column_id, s.segment_id;
```

This time, the number of segments is segment per row group; it has a global dictionary that apparently covers the whole table. How does that influence the queries?

Testing the CCI

Again, the first test is with the simple query:

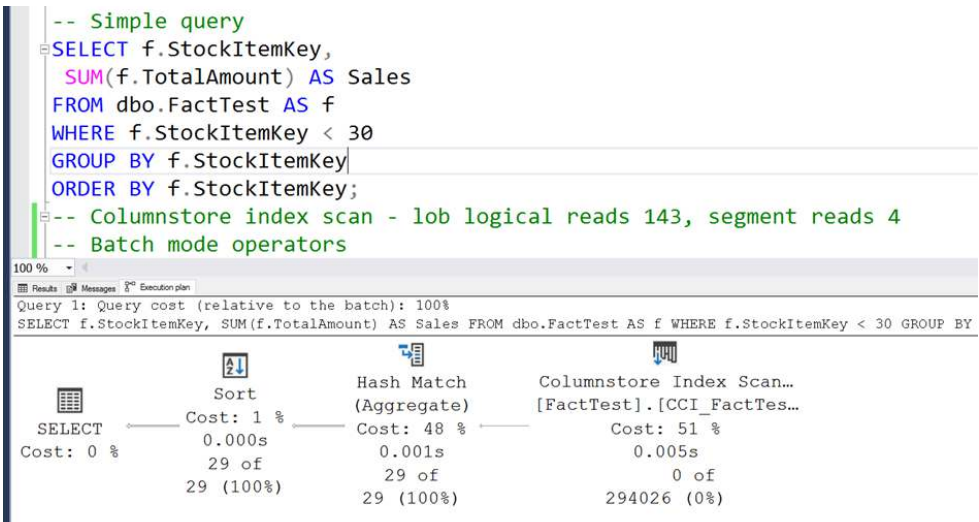
```
SET STATISTICS IO ON;
SELECT f.StockItemKey,
SUM(f.TotalAmount) AS Sales
FROM dbo.FactTest AS f
WHERE f.StockItemKey < 30
GROUP BY f.StockItemKey
ORDER BY f.StockItemKey;
```

From the IO output, you can see that SQL Server used only 143 logical reads. This time, this is really

impressive. Note that the execution plan again used the Columnstore Index Scan operator, this time scanning the CCI and selecting only the rowgroups and segments needed to satisfy the query. You can see the execution plan in the following screenshot.

The next query to test is the complex query:

```
SELECT f.SaleKey,
f.CustomerKey, f.Customer, cu.[Buying
Group],
f.CityKey, f.City, ci.Country,
f.DateKey, d.[Calendar Year],
f.StockItemKey, f.Product,
f.Quantity, f.TotalAmount, f.Profit
FROM dbo.FactTest AS f
INNER JOIN Dimension.Customer AS
cu
ON f.CustomerKey = cu.[Customer
```



```
Key]
INNER JOIN Dimension.City AS ci
ON f.CityKey = ci.[City Key]
INNER JOIN Dimension.[Stock Item] AS
s
ON f.StockItemKey = s.[Stock Item
Key]
INNER JOIN Dimension.Date AS d
ON f.DateKey = d.Date;
```

This time, SQL Server needed 5,656 LOB logical reads. SQL Server used a serial plan on my virtual machine, a mixed batch (for CCI scan and for hash joins), and row mode operators (for other index scans). This is only slightly better than when using an NCCI. How about the point query?

```
SELECT CustomerKey, Profit
FROM dbo.FactTest
WHERE CustomerKey = 378;
SET STATISTICS IO OFF;
```

The point query this time used 229 LOB logical reads. Better, but still not the best possible.

Using archive compression

You might remember that there is still one option left for columnar storage compression—archive compression. Let's turn it on with the following code:

```
ALTER INDEX CCI_FactTest
ON dbo.FactTest
REBUILD WITH (DATA_COMPRESSION
```

```
= COLUMNSTORE_ARCHIVE);
GO
```

You can imagine what comes next; recheck the space used by the test fact table:

```
EXEC sys.sp_spaceused
N'dbo.FactTest', @updateusage =
N'TRUE';
GO
```

The result is as follows:

```
Name rows reserved data index_size
unused
-----
-----
-----
-----
-----
dbo.FactTest 2279810 18632 KB 18536
KB 0 KB 96 KB
```

The LZ77 algorithm added some additional compression. Compare the data size now with the initial data size when the data size was 498,696 KB; now it is only 18,536 KB. The compression rate is more than 25 times! This is really impressive. Of course, you'd expect test queries to be even more efficient now. For example, here is the simple query:

```
SET STATISTICS IO ON;
SELECT f.StockItemKey,
SUM(f.TotalAmount) AS Sales
FROM dbo.FactTest AS f
WHERE f.StockItemKey < 30
GROUP BY f.StockItemKey
ORDER BY f.StockItemKey;
```

This time, SQL Server needed only 30 LOB logical reads. The next query to test is the complex query:

```
SELECT f.SaleKey,
f.CustomerKey, f.Customer, cu.[Buying
Group],
f.CityKey, f.City, ci.Country,
f.DateKey, d.[Calendar Year],
f.StockItemKey, f.Product,
f.Quantity, f.TotalAmount, f.Profit
FROM dbo.FactTest AS f
INNER JOIN Dimension.Customer AS
cu
ON f.CustomerKey = cu.[Customer
Key]
INNER JOIN Dimension.City AS ci
ON f.CityKey = ci.[City Key]
INNER JOIN Dimension.[Stock Item] AS
s
ON f.StockItemKey = s.[Stock Item
Key]
INNER JOIN Dimension.Date AS d
ON f.DateKey = d.Date;
```

This time, SQL Server needed 2,317

LOB logical reads in the test fact table. It can't get much better than this for scanning all of the data. And what about the point query?:

```
SELECT CustomerKey, Profit
FROM dbo.FactTest
WHERE CustomerKey = 378;
```

This time, it used 200 LOB logical reads. This number can still be improved.

Conclusion

We got really efficient queries. Mostly. The point query performance is still not very impressive. Classical index seek in a rowstore nonclustered index would be more efficient. In the next article, I will show how you can add classical balanced tree nonclustered indices and constraints to a table that is stored as a clustered columnstore index.

Šolanje za vaše napredovanje

AKTUALNI TEČAJI

Napredna uporaba strežnika **SQL Server in jezika** **Transact-SQL**

Kdaj: 18. - 20. 9. 2023

Advanced T-SQL Querying, **Programming and Tuning for** **SQL Server**

Kdaj: 25. - 29. 9. 2023



Gašper Rupnik
MCT, MS, MCSD, MCPS
gasper.rupnik@kompas-xnet.si

.NET MAUI DELAVNICA 3. DEL – PRIPRAVA RAZVOJNIH EMULATORJEV ALI NAPRAV (ANDROID, IOS)

Kot že lahko veste, sem v preteklih Pikah začel s t.i. .NET MAUI delavnico, serijo člankov na temo najnovejše Microsoft platforme za razvoj ene in iste aplikacije, ki jo lahko po želji objavíš kot spletno aplikacijo, namizno aplikacijo ali pa mobilno aplikacijo za bilokateri tip pametne naprave. Od zadnjega članka na to temo (2. del) je sicer minilo že kar nekaj mesecev oz. nekaj števil Pike. Moje delo je namreč kar raznoliko, zato sem v vmesnem času dal v mojih člankih .NET MAUI malo na stran in tako pisal tudi o drugih rečeh. To pa ne pomeni, da v tem času ni bilo tudi na področju .NET MAUIja kar nekaj novih spoznanj, o katerih bi lahko pisal v bodoče.

V zadnjem delu serije teh člankov sem tako govoril o sami postavitvi razvojnega okolja, danes pa si bomo lahko pogledali par osnovnih temeljev in posebnosti .NET MAUI platforme, katere je potrebno poznati oz. so ključne, pri razumevanju same platforme.

Ena izmed poglavitnih stvari je testiranje ter razhroščevanje naše nove aplikacije. To pride v poštev tekom samega razvoja ali pa na koncu, kjer lahko samo testiranje opravi tudi nekdo drug, ali pa stranka sama. Ker gre pri .NET MAUI za platformo, ki omogoča izdelavo aplikacije za bilokateri tip pametne naprave, je potrebno iti čez vsako in ob tem predebatirati specifikke.

Android

Z orodjem Visual Studio lahko enostavno testirate in razhroščujete svojo .NET MAUI aplikacijo z emulatorji, ki vam jih Visual Studio ponudi ali pa uporabite svojo Android napravo.

Za namen debugiranja preko emulatorjev, Visual Studio že vsebuje t.i. Android Device Manager, preko katerega si lahko kreirate svoj emulator oz. vam le tega ob prvem zagonu vaše .NET MAUI aplikacije

kreira sam Visual Studio. Kreira vam izbrano Android napravo z izbrano verzijo Androida, ki ima neko API oznako. Npr: Pixel 5 – API 33 (Android 13.0). Ob uporabi Android emulatorjev je potrebno povedati to, da je potreben tudi t.i. vklop hardware acceleration (Hyper-V & AEHD), saj se v bistvu v ozadju ustvarijo virtualne naprave znotraj vašega razvojnega okolja. Če to ni mogoče oz. na voljo, potem bodo vaši emulatorji tekli zelo počasi. Za postopek vklopa samega hardware acceleration-a, lahko dobite podrobnejša navodila na sledečem naslovu: <https://learn.microsoft.com/en-us/dotnet/maui/android/emulator/hardware-acceleration>

Za namen debugiranja preko svoje Android naprave, pa je potrebno najprej vklopiti na vaši napravi t.i. Developer mode, nato pa samo napravo priklopiti na vaš računalnik. To je več ali manj vse, računalnik in hkrati Visual Studio, vam napravo zazna in jo tudi ponudi kot napravo, preko katere lahko testirate aplikacijo.

Če vam le te ne zazna, potem je morebiti potrebno namestiti gonilnik naprave ali pa namestiti t.i. Google USB Driver preko Android SDK Manager-ja, ki ga ravno tako dobite v Visual Studiu. Omogočeno je debugiranje tudi preko WiFija. Več o razhroščevanju in testiranju preko ciljne Android naprave si lahko

preberete na sledečem naslovu: <https://learn.microsoft.com/en-us/dotnet/maui/android/device/setup>

Ob vsem skupaj bi rad izpostavil, da lahko razvoj opravljate tako na MacOS napravi (preko Visual Studio for Mac). Ali pa razvoj opravljate na Windows napravi. Postopek je v obeh primerih enak.

iOS

Pri testiranju in debugiranju na neki iOS napravi je zgodba malce drugačna. Namreč znotraj Windows okolja ne obstajajo emulatorji oz. možnost priklopa vaše iOS naprave direktno na Visual Studio znotraj Windows okolja. Le to je mogoče zgolj, če imate neko MacOS napravo in na njej nameščeno razvojno okolje Visual Studio for Mac z .NET MAUI razvojnimi paketi. V tem primeru imate možnost razvoja v Visual Studiu znotraj MacOS naprave, ali pa vaše Visual Studio znotraj Windows okolja povežete z Mac napravo, ki se nahaja znotraj vašega istega službenega omrežja. Torej, za samo testiranje in razhroščevanje vaše .NET MAUI aplikacije znotraj iOS naprave potrebujete:

1. MacOS napravo

a. Na kateri je nameščen Xcode razvojno okolje (z njim dobite tudi iOS emulatorje – iPhone, iPad, iWatch, itd)

- b. Visual Studio for Mac + nameščeno .NET MAUI razvojno okolje
- c. Na MacOS napravo lahko priklopite tudi svojo iOS napravo (iPhone, iPad), preko katere želite testirati svojo .NET MAUI aplikacijo

2. Windows napravo

- a. Na kateri je nameščen Visual Studio 2022 + nameščeno .NET MAUI razvojno okolje
- b. Visual Studio je povezan z zgornjo MacOS napravo preko možnosti »Pair to Mac«
- c. Tako imate možnost iz Windows okolja debugirati svojo .NET MAUI aplikacijo preko iOS emulatorjev na MacOS napravi ali preko ciljne iOS vaše naprave, ki je priklopljena (preko kabla ali preko WiFija) na vašo MacOS napravo.

Ob vsem skupaj bi rad izpostavil, da lahko razvoj opravljate zgolj na MacOS napravi. V primeru, da želite razvoj opravljati na Windows napravi, pa potrebujete tako MacOS kot Windows napravo.

Več o tem si lahko preberete na naslednji povezavi: <https://learn.microsoft.com/en-us/dotnet/maui/ios/cli>

MacOS

Pri testiranju in debugiranju na neki MacOS napravi je zgodba enaka, kot pri testiranju in debugiranju na iOS napravi, le da v tem primeru testirate na MacOS napravi in ne na iOS napravi.

Več o tem si lahko preberete na naslednji povezavi: <https://learn.microsoft.com/en-us/dotnet/maui/mac-catalyst/cli>

Windows

Pri testiranju in debugiranju na neki Windows napravi je zgodba za Windows razvojno okolje enostavna. V svojem Windows okolju zgolj vklopite Developer Mode in je to vse, kar je potrebno storiti.

Iz MacOS razvojnega okolja, pa samo testiranje in razhroščevanje na Windows napravi zaenkrat ni mogoče. Več o tem si lahko preberete na naslednji povezavi: <https://learn.microsoft.com/en-us/dotnet/maui/windows/setup>

To je zaenkrat vse za tokratno številko, naslednjič pa malo več o samem razvoju, kjer bomo lahko že nekaj več konkretno tudi ustvarili. Lep pozdrav do takrat.

Ustvari mobilno ali namizno aplikacijo z .NET MAUI

Kdaj: 18. - 21.9. 2023



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

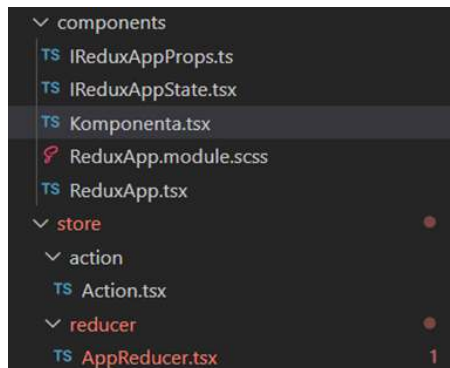
REDUX IN SPFX

Redux je priljubljena knjižnica za upravljanje stanja (state) za aplikacije, ki temeljijo na JavaScriptu. Zagotavlja centralizirano shrambo za stanje, ki omogoča predvidljivo upravljanje le tega med različnimi komponentami. Redux se običajno uporablja s frameworki, kot je React, lahko pa se uporablja tudi v SharePoint Framework aplikacijah (SPFx).

Osnovni koncepti Reduxa so shramba (store), ki hrani podatke o stanju aplikacije, akcije (actions) navadni JavaScript objekti, ki opisujejo dogodke in reduktorji (reducers) funkcije, ki izračunajo novo stanje glede na prejšnje stanje in akcijo. Komponente lahko izvajajo dejanja glede na dogodke, ki se izvajajo v aplikaciji, reduktorji pa bodo prejeli ta dejanja in jih uporabili za izračun novega stanja.

Da lahko redux uporabljamo v SPFx aplikaciji je treba najprej namestiti Redux, kar lahko naredimo preko NPM-ja, najprej zaženemo ukaz `npm install @reduxjs/toolkit` in v primeru, da uporabljamo React namestimo še `react-redux` z ukazom `npm install react-redux`.

Ko je klasična SPFx struktura ustvarjena, dodamo še podmapo store, v kateri so akcije in reduktorji. V glavno mapo komponent lahko dodamo ločene datoteke za lastnosti in stanje ter podrejene komponente.



Akcije pošiljajo podatke med aplikacijo in Redux shrambo. So dogodki, ki opisujejo dejanja v aplikaciji. Pri akcijah je pomembno da jim določimo vrednost `type`. Za Redux ni pomembno kako tip poimenujemo, bo pa gledal vrednost tipa za preverjanje sprememb, uporaben pa bo tudi pri razhroščevanju zato je dobro, da mu damo logično ime. Običajno se vse dodatne podatke shrani v spremenljivko `payload`, ki je lahko običajna spremenljivka ali pa objekt.

```
export const NALOZI_VSEBINO =
(items: any) => {
return {
type: 'NALOZI_VSEBINO',
payload: items
};
};
```

Reduktorji so edina možnost za spremembo stanja v Reduxu. Reduktorji sprejmejo trenutno stanje in argumente akcij ter vrnejo novo stanje. Aplikacija ima samo en reduktor, ki je odgovoren za upravljanje vseh akcij. Vsak reduktor potrebuje začetno stanje (v spodnjem primeru spremenljivka `initialState`), ki vsebuje začetno stanje aplikacije.

```
const RootReducer = (state =
initialState, action) => {
if (action.type === 'NALOZI_VSEBINO')
{
return {
...state,
items: action.payload
};
}
return state;
}
```

Pri renderiranju React komponent v glavno funkcijo `'export default class ReduxApp extends React.Component<IReduxAppProps, {}> {'` dodamo

```
private store: Store<IReduxAppState>
public constructor(props:
```

```
IReduxAppProps) {
super (props);
this.store = createStore(RootReducer,
applyMiddleware(logger));
}
```

Zgornja koda povezuje glavno komponento z Redux shrambo.

Da lahko uporabimo Redux shrambo sta na voljo funkciji `mapStateToProps`, ki bere vrednosti iz shrambe in jih mapira v stanje, ter `mapDispatchToProps`, ki posodablja vrednosti v shrambi.

```
const mapStateToProps = (state) => {
const { items } = state
return { items: items }
}

const mapDispatchToProps =
(dispatch) => {
return {
NALOZI_VSEBINO: (items: any) =>
dispatch(NALOZI_VSEBINO(items))
}
}

export default
connect(mapStateToProps,
mapDispatchToProps)(Seznam);
```

Prednosti uporabe Reduxa je centralizirano upravljanje stanja, eno samo stanje za vse komponente, kar omogoča predvidljive posodobitve stanja ter lažje odpravljanje napak. Slabosti so predvsem večja količina kode in ločene komponente, akcije in reduktorji ter privajanje na same elemente Reduxa.

Prednosti običajnega stanja je predvsem enostavnejša uporaba, manj kode, saj ni potrebe po ločenih akcijah in reduktorjih ter enostavna integracija, saj je že vključeno v React. Slabost je predvsem necentraliziranost stanja med različnimi komponentami, kar lahko privede do nedoslednosti in posledično so lahko spremembe stanja nepredvidljive.

Redux ponuja rešitev za upravljanje stanja za kompleksne potrebe komponent, medtem ko je stanje lokalne komponente React enostavnejše za osnovno upravljanje. Če je aplikacija zahtevnejša in potrebuje skupno stanje v številnih komponentah je Redux prava izbira, kadar pa aplikacija nima veliko komponent pa je najbolj priročno uporabiti osnovno stanje.

Lenart Svetek

lenart.svetek@kompas-xnet.si



POSTGRES.JS – KAKO SE POVEZATI S POSTGRESQL BAZO

Če ste kdaj delali z sql-om v javascriptu na bolj nizkem nivoju veste, kako težko je preverjanje vseh možnih napak in seveda pregledovanja uporabniškega vnosa, da slučajno ni notri tako imenovanega sql injection-a. Za ta problem so ustvarili razne ORM (Object-relational mapping slo. Objektno-relacijsko preslikovanje), ki poveže dva sicer nekompatibilna podatkovna Sistema in deluje kot translator. V javascript-u je več takih ORM, ki so namenjeni specifično za SQL od najmanj abstraktne implementacije, katere implementirajo le komunikacijo med JS in SQL, kot je knjižnica pg do najbolj abstraktirane implementacije, ki ne le

preveri za SQL injection ampak tudi abstraktira celoten SQL jezik, tako da razvijalci nikoli ne pridejo v stik z SQL-om, kot to naredi knjižnica Prisma.

Jaz vam bom pa predstavil Postgres.js, ki je "spiritual sucesor", knjižnici pg. Od uporabnika še vedno zahteva da sestavi svoje SQL poizvedbe, a preveri uporabniški vnos za kakršen koli SQL injection, da se s tem razvijalcu ni treba ukvarjati.

Knjižnico dobite lahko na tem naslovu [porsager/postgres: Postgres.js - The Fastest full featured PostgreSQL client for Node.js and Deno \(github.com\)](https://porsager/postgres: Postgres.js - The Fastest full featured PostgreSQL client for Node.js and Deno (github.com)).

Ko knjižnico dodate v svoj program,

ustvarite v svojem programskem okolju novo js datoteko, jaz jo poimenujem, kar db.js, ki sedaj izgleda:

```
import postgres from 'postgres';
let DATABASE_URL = "postgres://user:password@ghost/database"; // ?options=project%3DProject

const sql = postgres(DATABASE_URL, { ssl: 'require' });
```

Da preverimo, če smo se uspešno povezali na bazo, lahko zaženemo spodnjo poizvedbo, ki nam bo vrnila tabele v naši bazi:

```
async function getTables(){
  return await sql`SELECT table_name FROM information_schema.tables`; // WHERE table_schema = 'public'
}
```

Nazaj dobimo postgres objekt, čez katerega moramo iterirati ali s for zankami, z znanimi funkcijami, kot so .map ali .forEach, katere nam dajo dostop do podatkov po eno vrstico.

Ko osvojimo klicanje SQL poizvedb se lahko začnemo igrati z bolj močnimi funkcijami postgres.js knjižnice, kot so dinamični vnosi:

Pozor! Vse vrednosti obda z narekovaji tako da je »case sensitive«.

```
let sqlReturn = await sql`
  SELECT ${SELECT? sql(SELECT) : sql`*`} FROM ${sql(String(TABLE))}
  ORDER BY created DESC
  ${LIMIT? sql`FETCH FIRST ${LIMIT} ROWS ONLY` : sql``}
`;
```

Ki nam dovolijo več funkcionalnosti in nadzorom nad poizvedbo. Če želimo dati karkoli dinamičnega damo v \${}. Za vrednosti jih samo notri napišemo, kakor je napisana vrednost LIMIT v zgornjem primeru. Če želimo dodati SQL stavek ga obdamo v sql`` kakor je FETCH, identifikatorji, kot so imena tabel damo samo v sql(ime tabele), in ko želimo v SELECT stavku pridobiti, npr. poljubna polja jih lahko v seznamu pošljemo sql funkciji, katera jih bo preverila in vnesla v poizvedbo. Pri insert-u ti dovoli dodajanje poljubnih vrednostih in vrstic v poizvedbo.

Ko želite končati povezavo z bazo, pokličete le stavek sql.end().

**Aleš Lipušček**

MCP, MCTS, MCITP

ales.lipuscek@kompas-xnet.si



POWERSHELL KOTIČEK

Včasih potrebujemo obvestilo/ informacijo, da določen računalnik, še posebej, če je to strežnik, potrebuje ponovni zagon. Vse informacije, ki jih za to potrebujemo, se nahajajo v registru računalnika, ki nas zanima. S powershell skripto zelo enostavno pridemo do njih:

```
if(Test-Path »HKLM:
\SOFTWARE\Microsoft\Windows\Curre
ntVersion\Component Based
Servicing\RebootPending«) {
$PonovniZagonCaka = $true }
```

No, razlogi za ponovni zagon so lahko različni (sistemski popravki, preimenovanje datotek na sistemskem nivoju, nedokončani sistemski popravki ali posegi v sistem s strani SCCMja), zato moramo preiskati par različnih lokacij v registru:

```
Test-Path »HKLM:
\SOFTWARE\Microsoft\Windows\Curre
ntVersion\Component Based
Servicing\RebootPending
```

```
Test-Path »HKLM:
\SOFTWARE\Microsoft\Windows\Curre
ntVersion\WindowsUpdate\Auto
```

```
Update\RebootRequired"
```

```
Get-ItemProperty "HKLM:
\SYSTEM\CurrentControlSet\Control\S
ession Manager" -Name
PendingFileRenameOperations
```

```
Test-Path "HKLM:
\SOFTWARE\Microsoft\SMS\Mobile
Client\Reboot
Management\RebootData"
```

```
Test-path "HKLM:
\SOFTWARE\Microsoft\Updates\Updat
eExeVolatile"
```

Za potrditev zahteve za ponovni zagon mora vsaj ena od zgornjih poizvedb vrniti True.

Seveda pa bo veliko lažje, če nam skripta omogoči poizvedbo tudi na oddaljenih računalnikih, ne le na lokalnemu:

```
$reg=[Microsoft.Win32.RegistryKey]::O
penRemoteBaseKey(»LocalMachine«,
$oddaljen_racunalnik)
```

Nato pobrskamo po pridobljenih podatkih iz oddaljenega registra

```
$RegSubKeys =  
$RegCon.OpenSubKey("SOFTWARE\Microsoft\Windows\CurrentVersion\Component Based  
Servicing\").GetSubKeyNames()
```

Nato moramo le še potrditi, da je med izbranimi podatki ključ »Reboot pending«

```
$CBSreboot = $RegSubKeysCBS -  
contains "RebootPending"
```

Postopek je enak za 4 od zgoraj naštetih primerov, razen za PendingFileRenameOperations. Pri njem je poizvedba malo drugačna:

```
$RegSubKeySM =  
$RegCon.OpenSubKey("SYSTEM\CurrentControlSet\Control\Session  
Manager\")$RegPFRO =  
$RegSubKeySM.GetValue("PendingFile  
RenameOperations",$null)
```

Ko opravimo vse poizvedbe, se seveda spodobi, da vse odprte povezave (tj. vse klice na OpenRemoteBasekey) lepo zapremo z
\$reg.close

Možnih razlogov za zahteve po ponovnem zagonu je sicer več kot 12 (od večkratnih zagonov ob inštalacijah, do pridružitve domeni in podobno) a ob njih po navadi ne bomo zaganjali naših skript.

Šolanje za vaše napredovanje

AKTUALNI TEČAJI

MS700

Managing Microsoft Teams

Kdaj: 11. – 13.9.2023

AZ400

Designing and Implementing Microsoft DevOps solutions

Kdaj: 18. – 22.9.2023

MS102

Microsoft 365 Administrator

Kdaj: 27.11. – 1.12.2023

MD102

Microsoft 365 Endpoint Administrator

Kdaj: 18.12. – 11.12.2023

**Saša Kranjac**

MCT, MVP

sasa@kranjac.eu

A MICROSOFT APPROACH TO ZERO-TRUST

Microsoft defines several components in Zero Trust model that deserve our security focus. Each of these parts are foundational to security approach and are sources of signals and a security control plane:

1. Identities. Today, identities are not just people but devices and services as well.
2. Endpoints. Continuous monitoring of device health and security is crucial, from personal and IoT devices to servers and desktops.
3. Applications. Cloud or on-premises applications and APIs need access controls, usage analytics, monitoring and secure configuration.
4. Data. Securing data is fundamental. We need to ensure data is secure when in motion or at rest, on devices or when it leaves infrastructure and applications.

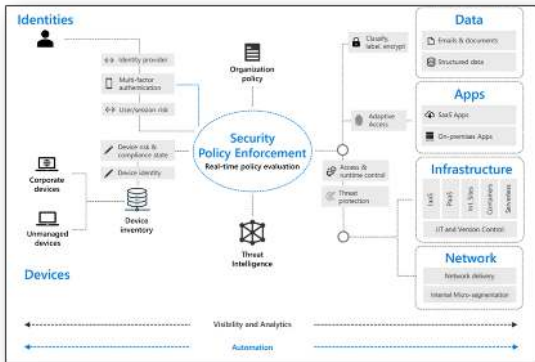
5. Infrastructure. On-premises or cloud infrastructure deserves proper security too. Restricting and controlling access using Just-in-Time and Just-Enough Administration (JIT, JEA), monitoring for anomalous behaviour, or using automation to shorten response time to risky behaviour.

6. Networks. Segmenting networks using subnets, multiple networks, user-defined routes, end-to-end encryption, monitoring and analytics. What is more important, Zero Trust approach includes automated security policy enforcement to ensure compliant behaviour throughout entire organisation

What is more important, Zero Trust approach includes automated security policy enforcement to ensure compliant behaviour throughout entire organisation



Picture 1 Zero Trust Security Components.

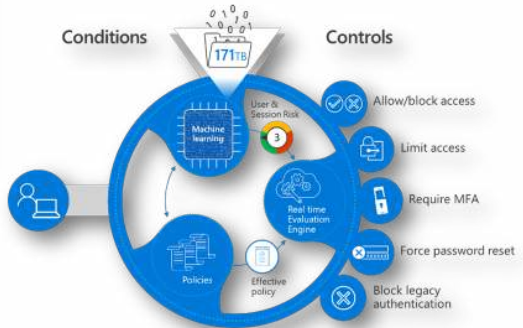


Picture 2 Zero Trust Security Policy Enforcement

So, which Microsoft products should we use to establish Zero Trust strategy? Well, different organisations will have different approaches to implementing zero trust methodology to security simply because no organisation is the same, does not use the same products and does not have the same infrastructure architecture. But the principles are still the same. Let us take a look at one approach.

Zero Trust approach should start with protecting identities, where each access request is equally treated and verified using strong identity authentication. Microsoft Entra ID (former Azure Active Directory, or Azure AD) is an identity and secure access management solution (IAM) that supports Multi-factor Authentication (MFA), where adding a second factor in authentication process can reduce lost or leaked passwords problems. To make

authentication even stronger, we can use Passwordless authentication using mobile authenticator app or FIDO2 token for even better protection.



Picture 3 Azure AD Conditional Access

Microsoft Entra ID not only enables strong authentication, but it provides Conditional Access to analyse signals, or conditions of users, devices, locations and automate and enforce resource access policies across entire organisation.

New
Conditional access policy

Control user access based on conditional access policy to bring signals together, to make decisions, and enforce organizational policies. [Learn more](#)

Control user access based on signals from conditions like risk, device platform, location, client apps, or device state. [Learn more](#)

Name *
Example: 'Device compliance app policy'

Assignments

Users and groups: 0 users and groups selected

Cloud apps or actions: No cloud apps or actions selected

Conditions: 0 conditions selected

Access controls

Grant: 0 controls selected

Session: 0 controls selected

User risk: Not configured

Sign-in risk: Not configured

Device platforms: Not configured

Locations: Not configured

Client apps: Not configured

Device state (Preview): Not configured

Picture 4 Azure AD Conditional Access Policy "wizard"

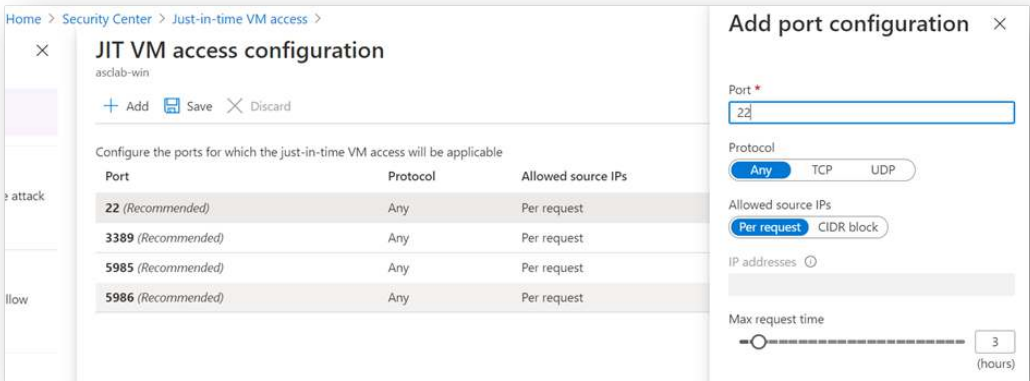
User access control is based on evaluations of several factors, such as user risk, sign-in risk, type of devices and platforms used, locations from where access is requested or device condition. After evaluating all signals, Microsoft Entra Conditional Access can grant access, allow access on specific conditions or block access to requested resources.

In case a legitimate user is requesting access to corporate resources, but the request originates from a compromised device or user sign-in risk signals impossible travel activity – a multi-factor authentication challenge can be requested, or a user access can be denied.

The Microsoft Entra Conditional Access is perhaps the most direct and picturesque example of “do not trust anyone”, “verify explicitly” or Zero Trust approach to security.

Just-in-Time and Just-Enough Administration (JIT, JEA) are not reserved solely for infrastructure Zero Trust approach but in protecting Identities as well.

Microsoft Defender for Cloud is a unified infrastructure security management system that provides security posture insight and actionable advice across Azure, on-premises, and 3rd party cloud workloads. Microsoft Defender for Cloud’s Cloud Workload Protection (CWP) enables intelligent workload protection across hybrid workloads and offers just-in-time virtual machine access. Customisable options include port, protocol, source IP addresses or CIDR blocks and maximum request time allowed to access a virtual machine.



Picture 5 ASC Just-in-time VM Access

Microsoft Entra Privileged Identity Management in contrast, controls JIT and JEA settings for Microsoft Entra identities. It enables security administrators to limit users' access to privileged roles, discovers users with privileged roles assignments and to perform privileged access reviews. Microsoft Entra PIM provides just-in-time privileged access to Microsoft Entra ID and Azure resources, enforces MFA to activate a role, use justification to follow and understand user activations, provides audit history and more.

To protect endpoints and devices, to detect and remediate advanced attacks on endpoints, Microsoft Defender for Endpoint supports Microsoft Windows operating systems from Windows 7 to Windows Server 2019, Android, MacOS and Linux.

Many organisations struggle with Shadow IT, with unknown and unsanctioned cloud applications. To discover applications used in corporate networks and to fight "the invisible and unknown enemy" – that is, attempt to control access to 3rd party applications or "shadow" cloud applications - IT departments need a

cloud app security broker (CASB). Fortunately, Microsoft Defender for Cloud Apps is a CASB solution that tightly integrates with other products in Microsoft portfolio. Its risk catalogue contains more than 16,000 applications that are assessed over 80 risk factors that help IT departments and security professionals make right decisions when assessing applications risk.

Products and features mentioned here are just a tip of the iceberg in a massive, possible Zero Trust security scenarios. You might be using some or all products cited and, of course, you might use some others too, like Microsoft Intune, Microsoft Defender for Identity, Microsoft Defender for Office 365, Data Loss Prevention (DLP) policies, BitLocker, Azure Policy, Microsoft Sentinel, Azure Blueprints, ARM templates, Azure DDoS Protection, Azure Web Application Firewall, Azure Firewall, Network Security groups and Application Security Groups.

No matter what you use, ensure you follow three basic Zero Trust principles: 1. Assume breach, 2. verify explicitly and 3. use least privileged access.

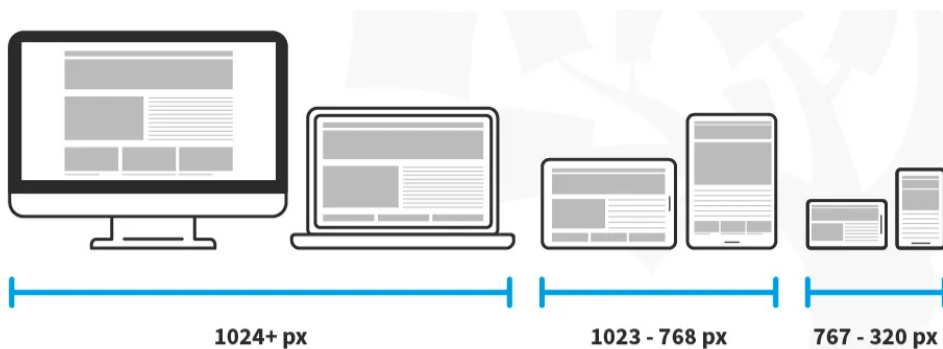


Premzl Urška
urska.premzl@kompas-xnet.si

SPLETNO OBLIKOVANJE ZA RAZLIČNE MOBILNE NAPRAVE

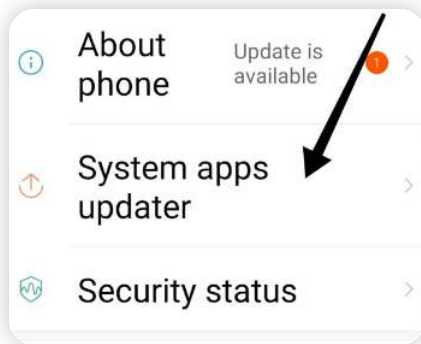
V današnjem digitalnem svetu je prilagoditev spletnega oblikovanja za vse mobilne naprave ključnega pomena. S hitrim napredkom tehnologije in nenehno rastjo uporabe pametnih telefonov in tablic se je povečala potreba po ustvarjanju uporabniško prijaznih in privlačnih spletnih strani ter aplikacij, ki se brezhibno prilagajajo različnim zaslonom in napravam. V tem članku bom pojasnila pomembnosti prilagoditve dizajna za mobilne naprave ter se osredotočila na ključne smernice, ki vodijo k ustvarjanju optimalne uporabniške izkušnje.

Odzivno dizajniranje (Responsive Design): Odzivno dizajniranje je osnovna tehnika za prilagajanje dizajna spletnih strani in aplikacij za mobilne naprave. Gre za pristop, pri katerem se dizajn prilagodi glede na velikost zaslona, na katerem se vsebina prikazuje. To zagotavlja doslednost v izgledu in izkušnji na različnih napravah.



Enostavna Navigacija: Dostopanje do informacij na mobilnih napravah zahteva drugačno razporeditev elementov in navigacije kot na večjih zaslonih. Enostavna in intuitivna navigacija je ključna za uspešno mobilno uporabniško izkušnjo.

Jasna tipografija: Izbrana tipografija mora biti berljiva na manjših zaslonih. Velikosti pisave, razmiki med vrsticami in dolžine odstavkov je treba prilagoditi za najboljšo berljivost.

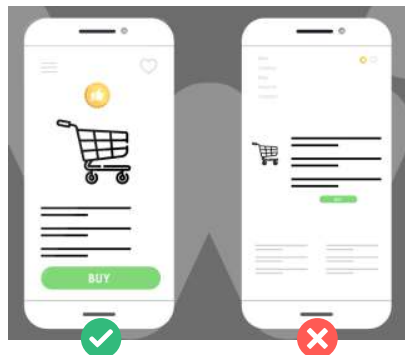


Optimizacija za hitrost nalaganja:

Hitrost nalaganja je ključnega pomena za mobilne uporabnike, ki se pogosto povezujejo prek mobilnih omrežij. Dizajn mora podpirati hitro nalaganje s strateškim stiskanjem slik, minimalno uporabo kode in predpomnilnikom.

Interakcije primerne za mobilne naprave: Dizajn interakcij, kot so gumbi, preklapanje in drsenje, mora upoštevati načine, kako uporabniki delujejo na mobilnih napravah.

Elementi za interakcijo morajo biti dovolj veliki in enostavni za uporabo s prstom.



Preizkušanje na različnih napravah:

Preizkus dizajna na širokem spektru mobilnih naprav je nujen. To zagotavlja dosledno izkušnjo za uporabnike, ki uporabljajo različne naprave.

Barvna paleta in kontrasti: Izberite barvno paletu, ki je primerna za manjše zaslone in zagotavlja dovolj kontrasta za jasno berljivost. Preveč intenzivne barve lahko povzročijo naprežanje oči.

Slike in grafike: Slike in grafike morajo biti prilagojene za mobilne naprave. To vključuje primerno velikost, format in kakovost, da ne obremenjujejo prenosa podatkov in hitrosti nalaganja.

Izbira ikon: Ikonografija je pomemben del dizajna za mobilne naprave. Uporabite preproste in dovolj velike ikone, ki bodo za uporabnika razumljive.

TIPS & TRICKS



Snemanje zaslona PowerPoint

Spoznajte, kako lahko izkoristite možnosti PowerPointa kot programa za snemanje – zajemite vaše predstavitve in celoten zaslon skupaj z dogajanjem.

PREDVAJAJ →



Več-ravenski seznam

Spoznajmo kako lahko hitro in učinkovito številčimo naše naslove in podnaslove, hkrati pa ohranimo povezavo med vsemi ravni dodeljenih slogov naslovov.

PREDVAJAJ →



3D računanje

Kako določiti več različnih virov podatkov pri naših izračunih? 3-D računanje nam omogoča zapisovanje med-listnih in med-dokumentnih kalkulacij pri našem delu v Excelu.

PREDVAJAJ →

Več vsebin lahko najdete tudi na našem Youtube kanalu:



Kompas Xnet